/* Sybase-Script made in December 1990 by Lutz-Peter Kurdelski
** AWI-Rechnergruppe
** Modified: 1991-09-04 by lpk
update_ship trigger geaendert: Beim
update wird das Schiff in jedem Fall
in die Tabelle Ship_Updated eingetragen.

** FILE: cruise.script
** Die Tabellen Ship und Cruise werden erzeugt.
** Es werden Triggers auf delete, insert, update gelegt,
** so dass nachvollziehbar ist, wer was wann geendert hat.
** Dieses Script geht davon aus, dass die Datenbank SouthernOceanDB
** bereits existiert.
*/

use master
/* Erzeugen der Datenbank
** Die Datenbank muss nicht mehr erzeugt werden.
*/
if not exists (select * from master.dbo.sysdatabases
where name = "SouthernOceanDB")
begin
print 'SouthernOceanDB does not exist !!!'
return
end

use SouthernOceanDB
(0 rows affected)
/* Erzeugen der Tabellen */
if exists (select * from sysobjects where name="Cruise")
drop table Cruise

create table Cruise
(Cruise_Number# int, /* internal identification */
Ship_Id# int, /* Cruise number */
)
if exists (select * from sysobjects where name="Cruise_Deleted")
drop table Cruise_Deleted

create table Cruise_Deleted
(Cruise_Number# int, /* internal identification */
Ship_Id# int, /* Cruise number */
user_id# int, /* who deleted data */
date datetime /* date of deletion */
)
if exists (select * from sysobjects where name="Cruise_Updated")
drop table Cruise_Updated

create table Cruise_Updated
(Cruise_Number# int, /* internal identification */
Ship_Id# int, /* Cruise number */
Old_Cruise_Number# int, /* internal identification */
Old_Ship_Id# int, /* Cruise number */
user_id# int, /* who updated data */
date datetime /* date of update */
)
if exists (select * from sysobjects where name='Ship')
drop table Ship

create table Ship
    (Ship_Id# int, /* internal identification */
     Ship_Name varchar(80), /* the name of the ship */
     Country varchar(80), /* the country */
     Ship_Code varchar(8) null /* the ship code */)

if exists (select * from sysobjects where name='Ship_Deleted')
drop table Ship_Deleted

create table Ship_Deleted
    (Ship_Id# int, /* internal identification */
     Ship_Name varchar(80), /* the name of the ship */
     Country varchar(80), /* the country */
     Ship_Code varchar(8) null /* the ship code */
     user_id# int,
     date datetime)

if exists (select * from sysobjects where name='Ship_Updated')
drop table Ship_Updated

create table Ship_Updated
    (Ship_Id# int, /* internal identification */
     Ship_Name varchar(80), /* the name of the ship */
     Country varchar(80), /* the country */
     Ship_Code varchar(8) null, /* the ship code */
     Old_Ship_Name varchar(80), /* the name of the ship */
     Old_Country varchar(80), /* the country */
     Old_Ship_Code varchar(8) null, /* the ship code */
     user_id# int,
     date datetime)

/* Definition der Primaerschluessel in den Tabellen */
execute sp_primarykey Cruise, Cruise_Number#
execute sp_primarykey Ship, Ship_Id# New primary key added.
   (return status = 0)
   New primary key added.
   (return status = 0)

/* Definition der Sekundaerschluessel in den Tabellen */
execute sp_foreignkey Cruise, Ship, Ship_Id# New foreign key added.
   (return status = 0)

/* Generieren der Indizes */
/*
create unique index Cruise_Index on Cruise (Cruise_Number#)
Msg 113, Level 15, State 1:
Server 'SYBASE401', Line 7:
Missing end comment mark '*/'.
*/

create clustered index Cruise_Ship_Index on Cruise (Ship_Id#)
Msg 102, Level 15, State 1:
Server 'SYBASE401', Line 1:
Incorrect syntax near '*'.
1> create unique index Ship_Index on Ship (Ship_Id#)
2> /* Erzeugen der Views */
3> /* Erzeugen der Prozeduren */
4> /* Erzeugen von Trigger */
5> create trigger Insert_Cruise on Cruise for insert as begin
6> declare @msg varchar(80),
7> @sid int
8> if not exists (select *
9> from Ship, inserted
10> where Ship.Ship_Id# = inserted.Ship_Id#)
11> begin
12> select @sid = Ship_Id# from inserted
13> select @msg = "INSERT Cruise: Please insert the ship " +
14> convert(varchar(20),@sid) + " first into Ship"
15> print @msg
16> rollback transaction
17> end
18> create trigger Delete_Cruise on Cruise for delete as begin
19> if exists ( select *
20> from Station, deleted
21> where Station.Cruise_Number = deleted.Cruise_Number#)
22> begin
23> print "DELETE Cruise: There exists a station corresponding to this C"
24> rollback transaction
25> end
26> else
27> begin
28> insert into Cruise_Deleted
29> select deleted.Cruise_Number#, deleted.Ship_Id#,
30> suser_id(), getdate()
31> from deleted
32> end
33> end
34> create trigger Update_Cruise on Cruise for update as begin
35> declare @msg varchar(80),
36> @sid int
37> if not exists ( select *
38> from Ship, inserted
39> where Ship.Ship_Id# = inserted.Ship_Id#)
40> begin
```sql
select @sid = Ship_Id# from inserted
select @msg = "UPDATE Cruise: Please insert the ship #" +
    convert(varchar(20),@sid) + " first into Ship"
print @msg
rollback transaction
end
else
    begin
        if update(Cruise_Number#)
            begin
                if exists (select *
                        from Station, deleted
                        where Station.Cruise_Number = deleted.Cruise_Number#)
                    begin
                        select @sid = Cruise_Number from deleted
                        select @msg = "UPDATE Cruise: There exists at least one station"
                        + "+" + convert(varchar(20),@sid) + ")" +
                        + " corresponding to this Cruise_Number"
                    print @msg
                    rollback transaction
                end
            end
        else
            begin
                insert into Cruise Updated
                select inserted.Cruise_Number#,
                    inserted.Ship_Id#,­
                deleted.Cruise_Number#,
                    deleted.Ship_Id#,­
                suser_id(),
                getdate()
                from inserted, deleted
                print "UPDATE Cruise successful"
            end
    end
end
Msg 102, Level 15, State 1:
Server 'SYBASE401', Procedure 'Update_Cruise', Line 50:
Incorrect syntax near 'end'.
1>
2> create trigger Delete_Ship
3> on Ship
4> for delete
5> as
6> begin
7> declare @msg varchar(80),
8>     @sid int
9> if exists (select *
10>     from Cruise, deleted
11>     where Cruise.Ship_Id# = deleted.Ship_Id#)
12>     begin
13>         select @sid = Ship_Id# from deleted
14>         select @msg = "DELETE Ship: Please delete the cruises for ship #" +
15>             convert(varchar(20),@sid) + " first"
16>     print @msg
17>     rollback transaction
18> end
19>
2> create trigger Update_Ship
3> on Ship
4> for update
5> as
6> begin
7> insert into Ship Updated
8> select deleted.Ship_Id#,
```
null

/* Eintragen der Nutzer */
grant select, insert, delete on Cruise to PHYSIK1

grant select, insert, delete on Ship to PHYSIK1
```sql
/* Sybase-Script made in by Lutz-Peter Kurdelski
** AWI-Rechnergruppe
** Creation-Date: 1991-02-07
** Changing of cruise ids in table Cruise which has been
** erroneous in table Aari_Cruise_Bck.
** Aendern der Cruise Identifikation in der Tabelle Cruise,
** da diese Daten aus der an dieser Stelle fehlerhaften
** Tabelle Aari_Cruise_Bck entnommen wurden.
** FILE: CRUISE_UPD070291.SCRIPT
*/

-- Dieses Script geht davon aus, dass die Datenbank SouthernOceanDB
** bereits existiert.

use master
if not exists ( select *
from master.dbo.sysdatabases
where name = "SouthernOceanDB"
)
begin
print 'SouthernOceanDB does not exist !!!'
return
end

use SouthernOceanDB
(0 rows affected)

/* Erzeugen der Tabellen */
/* Definition der Primaeurschlussel in den Tabellen */
/* Definition der Sekundaerschlussel in den Tabellen */
/* Generieren der Indizes */
/* Erzeugen der Views */
/* Erzeugen der Prozeduren */
/* Erzeugen von Trigger */
/* SQL Statements */

create procedure UpdateShipCruise @cid int,
    @sn varchar (80) = null,
    @m int = null
as
    print 'enter UpdateShipCruise'
    insert Ship
        values(@m,@sn,'unknown',NULL)
    update Cruise
    set Ship_Id# = @m
    where Cruise_Number# = @cid
    print 'leave UpdateShipCruise'

create table #temp (id# int not null,
    number int null,
    name varchar(80) null)

begin transaction

declare @counter int,
        @max int,
        @m int,
        @number int,
        @s varchar (80),
        @msg varchar(80)
```
insert #temp values (1, -25000, 'unknown')
insert #temp values (2, -25001, 'unknown')
insert #temp values (3, -25002, 'unknown')
insert #temp values (4, -25003, 'unknown')
insert #temp values (5, -25004, 'unknown')
insert #temp values (6, -25005, 'unknown')
insert #temp values (7, -25006, 'unknown')
insert #temp values (8, -25007, 'unknown')
insert #temp values (9, -25016, 'unknown')
insert #temp values (10, -25017, 'unknown')
insert #temp values (11, -25018, 'unknown')
insert #temp values (12, -25019, 'unknown')
insert #temp values (13, -25020, 'unknown')
insert #temp values (14, -25021, 'unknown')
insert #temp values (15, -25022, 'unknown')
insert #temp values (16, -25023, 'unknown')
insert #temp values (17, -25024, 'unknown')

select @counter = 0
select @max = (select count(*) from #temp)
while @counter < @max
begin
    select @counter = @counter + 1
    select @msg = 'loop value = ' + convert(varchar(4), @counter)
    print @msg
    if exists (select * from Cruise
        where Cruise_Number# = (select number from #temp
            where id# = @counter))
    begin
        select @m = @m + 1
        select @msg = ' max ship = ' + convert(varchar(4), @m)
        print @msg
        select @number = (select number from #temp
            where id# = @counter)
        select @msg = ' number = ' + convert(varchar(7), @number)
        print @msg
        select @s = (select name from #temp
            where id# = @counter)
        select @msg = ' name = ' + @s
        print @msg
        exec UpdateShipCruise @cid = @number,
            @sn = @s,
            @m = @m
    end
end

select @s =
(select Ship from Aari_Cruise where Cruise_Number = -25001)
print @s
select * from Aari_Cruise
    where Cruise_Number between -25024 and -25016
    or Cruise_Number between -25007 and -25000
select * from Aari_Cruise where Ship = 'Ob'
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
loop value = 1
(0 rows affected)
(1 row affected)
max ship = 253
(1 row affected)
number = -25000
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 2
(0 rows affected)
(1 row affected)
max ship = 254
(1 row affected)
number = -25001
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 3
(0 rows affected)
(1 row affected)
max ship = 255
(1 row affected)
number = -25002
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 4
(0 rows affected)
(1 row affected)
(1 row affected)
max ship = 256
(1 row affected)
(1 row affected)
number = -25003
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 5
(0 rows affected)
(1 row affected)
(1 row affected)
max ship = 257
(1 row affected)
(1 row affected)
number = -25004
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 6
(0 rows affected)
(1 row affected)
(1 row affected)
max ship = 258
(1 row affected)
(1 row affected)
number = -25005
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 7
(0 rows affected)
(1 row affected)
(1 row affected)
max ship = 259
(1 row affected)
(1 row affected)
number = -25006
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 8
(0 rows affected)
(1 row affected)
(1 row affected)
max ship = 260
(1 row affected)
(1 row affected)
number = -25007
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 9
(0 rows affected)
(1 row affected)
(1 row affected)
max ship = 261
(1 row affected)
(1 row affected)
number = -25016
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 10
(0 rows affected)
(1 row affected)
(1 row affected)
max ship = 262
(1 row affected)
(1 row affected)
number = -25017
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 11
(0 rows affected)
(1 row affected)
(1 row affected)
max ship = 263
(1 row affected)
(1 row affected)
number = -25018
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 12
(0 rows affected)
(1 row affected)
(1 row affected)
max ship = 264
(1 row affected)
(1 row affected)
number = -25019
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 13
(0 rows affected)
(1 row affected)
max ship = 265
(1 row affected)
(1 row affected)
number = -25020
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 14
(0 rows affected)
(1 row affected)
max ship = 266
(1 row affected)
(1 row affected)
number = -25021
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 15
(0 rows affected)
(1 row affected)
max ship = 267
(1 row affected)
(1 row affected)
number = -25022
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise
(return status = 0)
(1 row affected)
(1 row affected)
loop value = 16
(0 rows affected)
(1 row affected)
(max ship = 268
(1 row affected)
(1 row affected)
number = -25023
(1 row affected)
(1 row affected)
name = unknown
enter UpdateShipCruise
leave UpdateShipCruise  
(return status = 0)  
(1 row affected)  
(1 row affected)  
loop value = 17  
(0 rows affected)  
(1 row affected)  
max ship = 269  
(1 row affected)  
number = -25024  
(1 row affected)  
name = unknown  
enter UpdateShipCruise  
leave UpdateShipCruise  
(return status = 0)  
Msg 512, Level 16, State 1:  
Server 'SYBASE401', Line 57:  
Subquery returned more than 1 value. This is illegal when the  
subquery follows =, ! =, <, <= , >, >=, or when the subquery is used as  
an expression.  
(0 rows affected)  
unknown  
Cruise_Number  
Ship  
Country  
Code  

<table>
<thead>
<tr>
<th>Cruise_Number</th>
<th>Ship</th>
<th>Country</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>-25000</td>
<td>Discovery II</td>
<td>U.K.</td>
<td>NULL</td>
</tr>
<tr>
<td>-25000</td>
<td>Discovery II</td>
<td>U.K.</td>
<td>NULL</td>
</tr>
<tr>
<td>-25000</td>
<td>Discovery II</td>
<td>U.K.</td>
<td>NULL</td>
</tr>
<tr>
<td>-25000</td>
<td>Discovery II</td>
<td>U.K.</td>
<td>NULL</td>
</tr>
<tr>
<td>-25001</td>
<td>Ob</td>
<td>USSR</td>
<td>NULL</td>
</tr>
<tr>
<td>-25001</td>
<td>Ob</td>
<td>USSR</td>
<td>NULL</td>
</tr>
<tr>
<td>-25001</td>
<td>Ob</td>
<td>USSR</td>
<td>NULL</td>
</tr>
<tr>
<td>-25001</td>
<td>Ob</td>
<td>USSR</td>
<td>NULL</td>
</tr>
<tr>
<td>-25002</td>
<td>Ob</td>
<td>USSR</td>
<td>NULL</td>
</tr>
<tr>
<td>ID</td>
<td>Name</td>
<td>Country</td>
<td>City</td>
</tr>
<tr>
<td>-----</td>
<td>----------------</td>
<td>---------</td>
<td>----------</td>
</tr>
<tr>
<td>-25006</td>
<td>Gen. San-Martin</td>
<td>Argentina</td>
<td>NULL</td>
</tr>
<tr>
<td>-25006</td>
<td>William Scoresby</td>
<td>U.K.</td>
<td>NULL</td>
</tr>
<tr>
<td>-25007</td>
<td>Carnegie</td>
<td>USA</td>
<td>NULL</td>
</tr>
<tr>
<td>-25016</td>
<td>S.E. Baird</td>
<td>USA</td>
<td>NULL</td>
</tr>
<tr>
<td>-25017</td>
<td>unknown</td>
<td>Denmark</td>
<td>NULL</td>
</tr>
</tbody>
</table>
-25022
Galathea
Denmark
NULL
-25022
Galathea
Denmark
NULL
-25023
Tui
New Zealand
NULL
-25023
Tui
New Zealand
NULL
-25023
Tui
New Zealand
NULL
-25024
Argo
USA
NULL
-25003
unknown
unknown
NULL
-25004
unknown
unknown
NULL
-25005
unknown
unknown
NULL
-25006
unknown
unknown
NULL
-25016
unknown
unknown
NULL
-25017
unknown
unknown
NULL
-25018
<table>
<thead>
<tr>
<th>Cruise Number</th>
<th>Ship</th>
<th>Country</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>-21482</td>
<td>Ob</td>
<td>USSR</td>
<td>NULL</td>
</tr>
<tr>
<td>-318</td>
<td>Ob</td>
<td>USSR</td>
<td>NULL</td>
</tr>
<tr>
<td>4298</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Ob
USSR
NULL
55001
Ob
USSR
NULL
186
Ob
USSR
NULL
187
Ob
USSR
NULL
288
Ob
USSR
NULL
369
Ob
USSR
NULL
20
Ob
USSR
NULL
17
Ob
USSR
NULL
18
Ob
USSR
NULL
-1318
Ob
USSR
NULL
-21482
Ob
USSR
NULL
10063
Ob
USSR
NULL
-25001
Ob
USSR
NULL
-318
Ob
USSR
NULL
4298
Ob
USSR
NULL
55001
Ob
USSR
NULL
186
Ob
USSR
| NULL | 10063 | Ob | USSR | NULL | -25001 | Ob | USSR | NULL | -318 | Ob | USSR | NULL | 4298 | Ob | USSR | NULL | 55001 | Ob | USSR | NULL | 186 | Ob | USSR | NULL | 187 | Ob | USSR | NULL | 288 | Ob | USSR | NULL | 369 | Ob | USSR | NULL | 20 | Ob | USSR | NULL | 17 | Ob | USSR | NULL | 18 | Ob | USSR | NULL |

(65 rows affected)

```sql
if @@error != 0
  rollback transaction
else
  commit transaction

drop table #temp
drop procedure UpdateShipCruise
```
```sql
/* SYBASE script by L.-P. Kurdelski */
** Inserting new Cruises to Cruise and Ship from Tokyo Fisheries Data
** created 1991-02-13
**
** FILE: ADD_TOKYO_SHIPS.SCRIPT
*/

use master
if not exists ( select * from master.dbo.sysdatabases
    where name = "SouthernOceanDB"
) begin
    print 'SouthernOceanDB does not exist !!!'
    return
end

use SouthernOceanDB
(0 rows affected)
begin transaction

declare @shipid int,
    @ship varchar(80),
    @msg varchar(80)

select @ship = 'Umitaka-Maru'
select @shipid = (select Ship_Id#
    from Ship
    where Ship_Name like @ship
    and Ship_Id# < 100000 )

insert Cruise values (-26001,@shipid)
insert Cruise values (-26002,@shipid)
insert Cruise values (-26003,@shipid)
insert Cruise values (-26004,@shipid)
insert Cruise values (-26005,@shipid)

select Cruise_Number, Ship from Aari_Cruise where Ship = @ship
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)

---
-21692
Umitaka-Maru
602
Umitaka-Maru
55004
Umitaka-Maru
-26001
Umitaka-Maru
-26002
Umitaka-Maru
-26003
Umitaka-Maru
-26004
Umitaka-Maru
-26005
```
Umitaka-Maru
-1934
Umitaka-Maru
-21692
Umitaka-Maru
-25004
Umitaka-Maru
602
Umitaka-Maru
55004
Umitaka-Maru
-21692
Umitaka-Maru
-25004
Umitaka-Maru
602
Umitaka-Maru
55004
Umitaka-Maru
-21692
Umitaka-Maru
-25004
Umitaka-Maru
602
Umitaka-Maru
55004
Umitaka-Maru
-21692
Umitaka-Maru
-25004
Umitaka-Maru
602
Umitaka-Maru
55004
Umitaka-Maru
(25 rows affected)
1> if @@error != 0
2> rollback transaction
3> else
4> commit transaction
5>
/* SYBASE Script vom 1991-03-08 */

use SouthernOceanDB
begin transaction
exec updateCruise -25000, 'Fuji', 'Japan'
Msg 515, Level 16, State 3: Server 'SYBASE401', Procedure 'updateCruise', Line 23:
Attempt to insert the value NULL into column 'Ship_Id#', table 'SouthernOceanDB.dbo.Cruise'; column does not allow nulls. Update fails.
(return status = -6)
exec updateCruise -25001, 'Fuji', 'Japan'
Msg 515, Level 16, State 3: Server 'SYBASE401', Procedure 'updateCruise', Line 23:
Attempt to insert the value NULL into column 'Ship_Id#', table 'SouthernOceanDB.dbo.Cruise'; column does not allow nulls. Update fails.
(return status = -6)
exec updateCruise -25002, 'Fuji', 'Japan'
Msg 515, Level 16, State 3: Server 'SYBASE401', Procedure 'updateCruise', Line 23:
Attempt to insert the value NULL into column 'Ship_Id#', table 'SouthernOceanDB.dbo.Cruise'; column does not allow nulls. Update fails.
(return status = -6)
exec updateCruise -22012, 'Professor Zubov', 'USSR'
Msg 515, Level 16, State 3: Server 'SYBASE401', Procedure 'updateCruise', Line 23:
Attempt to insert the value NULL into column 'Ship_Id#', table 'SouthernOceanDB.dbo.Cruise'; column does not allow nulls. Update fails.
(return status = -6)
exec updateCruise -25007, 'Novoukrainka', 'USSR'
(return status = 0)
exec updateCruise -22013, 'Prof. Viese', 'USSR'
Msg 515, Level 16, State 3: Server 'SYBASE401', Procedure 'updateCruise', Line 23:
Attempt to insert the value NULL into column 'Ship_Id#', table 'SouthernOceanDB.dbo.Cruise'; column does not allow nulls. Update fails.
(return status = -6)
if @@error != 0
    rollback transaction
else
    commit transaction
use SouthernOceanDB
truncate table Ship
truncate table Cruise
1> /*
2> ** SYBASE Script by L.-P. Kurdelski
3> **
4> ** FILE: AL111_SHIPS.SCRIP
5> **
6> ** CREATED: 1991-03-27
7> **
8> ** MODIFIED:
9> ** 1991-08-19 lpk Procedures insert_into_cruise and
10> ** insert_new_ships are expected.
11> ** If they are not available they will
12> ** be created.
13> ** 1991-08-29 lpk The procedures will not be created.
14> **
15> ** Insert new ships
16> ** The ships are taken from the AL111 dataset.
17> */
18> if not exists (select * from sysobjects where name like "SouthernOceanDB")
19> begin
20> print "SouthernOceanDB not found"
21> return
22> end
23> begin
24> if not exists (select * from sysobjects where name like "insert_into_cruise")
25> begin
26> print "Procedure insert_into_cruise missing."
27> return
28> end
29> begin
30> if not exists (select * from sysobjects where name like "insert_new_ships")
31> begin
32> print "Procedure insert_new_ships missing."
33> return
34> end
35> begin
36> if exists (select * from sysobjects where name like "tempships")
37> begin
38> truncate table tempships
39> end
40> else
41> begin
42> print "No table tempships"
43> return
44> end
45> declare @zaehler int
46> select @zaehler = 0
47> select @zaehler = @zaehler + 1
48> insert tempships values(@zaehler, -21017, "Krusenstern", "USSR")
49> select @zaehler = @zaehler + 1
50> insert tempships values(@zaehler, -21018, "Krusenstern", "USSR")
51> select @zaehler = @zaehler + 1
52> insert tempships values(@zaehler, -22015, "Prof. Viese", "USSR")
53> select @zaehler = @zaehler + 1
54> insert tempships values(@zaehler, -22016, "Prof. Zubov", "USSR")
55> select @zaehler = @zaehler + 1
56> insert tempships values(@zaehler, -22019, "Pioner Latvii", "USSR")
57> select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22020, "Titanit", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22021, "Atlantniro", "USSR")
insert tempships values(@zaehler, -22022, "Maltsevo", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22023, "Volniy Veter", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22024, "Pioner Latvii", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22025, "Ocher", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22026, "Patriot", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22027, "Ekliptika", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22028, "Salehard", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22029, "Anchar", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22030, "Novocheboksarsk", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22031, "Afeliy", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22032, "Ekliptika", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22033, "Atlantniro", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22034, "Monokristall", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22035, "Saulkrasty", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22036, "Andrus Iohann", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22037, "Torok", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22038, "Salehard", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22039, "Tava", "USSR")
select @zaehler = @zaehler + 1
insert tempships values(@zaehler, -22040, "Anchar", "USSR")

(12 rows affected)
begin transaction
exec insert new ships
(return status = 0)
if @@error !$= 0
    begin
        print 'transaction aborted'
        rollback transaction
    end
else
    begin
        print 'transaction successful'
        commit transaction
    end
transaction successful
truncate table tempships
print "table tempships truncated"

table tempships truncated
/*
** SYBASE Script by L.-P. Kurdelski
** CREATED: 1991-03-27
**
** Modification of the relation between a ship and a cruise.
**
** FILE: Allll SHIPS_UPD.SCRIPT
**
*/
use SouthernOceanDB

if not exists (select * from sysobjects where name like "changeShipNameInCruise")
begin
print "Procedure changeShipNameInCruise not found."
return
end
(rows affected)
exec changeShipNameInCruise "Prof. Vieze", "Prof. Viese", 0, 0
Update of Prof. Vieze to Prof. Viese successful.
(return status = 0)
begin transaction
delete from Ship where Ship_Name like "Prof. Vieze"
(1 row affected)
if @@error != 0
begin
print 'transaction aborted'
rollback transaction
end
else
begin
print 'transaction successful'
commit transaction
end
transaction successful!
source. onf$:daten: [s0cean.text]unknownchanges.text

Modifications:
** 1991-10-29  "Prof. Viese" should be spelled as "Prof. Vieze"
**  Insert new ship "Investigator", "Australia"
**  "Discovery" is from "U.K."
**  Three new faulty relations corrected for
**  Cruise_Numbers 9366, 9368, - 629.

Declare @smax int
select @smax = max(Ship_Id#) from Ship where Ship_Id# < 100000
select @smax = @smax + 1
insert into Ship (Ship_Id#, Ship_Name, Country)
values (@smax, "Investigator", "Australia")
(1 row affected)
(1 row affected)
(1 row affected)

exec changeShipNameAndCountryInCruise
"unknown", "unknown", "Prof. Vieze", "USSR", 7660, 0
Update of unknown to Prof. Vieze successful.
(return status = 0)

exec changeShipNameAndCountryInCruise
"unknown", "unknown", "Prof. Zubov", "USSR", 10186, 0
Update of unknown to Prof. Zubov successful.
(return status = 0)

exec changeShipNameAndCountryInCruise
"unknown", "unknown", "Prof. Zubov", "USSR", 3494, 0
Update of unknown to Prof. Zubov successful.
(return status = 0)

exec changeShipNameAndCountryInCruise
"unknown", "unknown", "Discovery", "U.K.", 55071, 0
Update of unknown to Discovery successful.
(return status = 0)

exec changeShipNameAndCountryInCruise
"unknown", "unknown", "Yelcho", "Chile", 5367, 0
Update of unknown to Yelcho successful.
(return status = 0)

exec changeShipNameAndCountryInCruise
"unknown", "unknown", "Edisto", "USA", 3482, 0
Update of unknown to Edisto successful.
(return status = 0)

exec changeShipNameAndCountryInCruise
"unknown", "unknown", "Investigator", "Australia", 426, 0
Update of unknown to Investigator successful.
(return status = 0)

exec changeShipNameAndCountryInCruise
"unknown", "unknown", "Investigator", "Australia", 634, 0
Update of unknown to Investigator successful.
(return status = 0)

exec changeShipNameAndCountryInCruise
"unknown", "unknown", "Investigator", "Australia", 635, 0
Update of unknown to Investigator successful.
Update of unknown to Investigator successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Investigator", "Australia", 636, 0
Update of unknown to Investigator successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Investigator", "Australia", 637, 0
Update of unknown to Investigator successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Investigator", "Australia", 638, 0
Update of unknown to Investigator successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Investigator", "Australia", 639, 0
Update of unknown to Investigator successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Investigator", "Australia", 643, 0
Update of unknown to Investigator successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Investigator", "Australia", 678, 0
Update of unknown to Investigator successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Gascoyne", "Australia", 410, 0
Update of unknown to Gascoyne successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Yelcho", "Chile", 5371, 0
Update of unknown to Yelcho successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Chipana", "Chile", 5369, 0
Update of unknown to Chipana successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Prof. Zubov", "USSR", 14539, 0
Update of unknown to Prof. Zubov successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Myslitel", "USSR", 10341, 0
Update of unknown to Myslitel successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Prof. Vieze", "USSR", 4244, 0
Update of unknown to Prof. Vieze successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Melville", "USA", 11669, 0
Update of unknown to Melville successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Myslitel", "USSR", -1192, 0
Update of unknown to Myslitel successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "Investigator", "India", "Investigator", "Australia", 9366, 0
Update of Investigator to Investigator successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "Investigator", "India", "Investigator", "Australia", 9368, 0
Update of Investigator to Investigator successful.
(return status = 0)
Update of Investigator to Investigator successful.
(return status = 0)
1> /* Sybase script
2> **
3> ** Filename : Biomass.script
4> **
5> ** Author: L.-P. Kurdelski
6> **
7> ** Creation date : 1991-07-09
8> **
9> ** Modification History:
10> **
11> */
12> use SouthernOceanDB
13>
14> if exists (select * from sysobjects where name like "Biomass_Station_BCK")
15> drop table Biomass_Station_BCK
16> if exists (select * from sysobjects where name like "Biomass Standard Data_BCK")
17> drop table Biomass Standard Data_BCK
18> if exists (select * from sysobjects where name like "Biomass Station")
19> exec sp_rename Biomass Station, Biomass Station_BCK
20> Object name has been changed.
21> (return status = 0)
22>
23> if exists (select * from sysobjects where name like "Biomass Standard Data")
24> exec sp_rename Biomass Standard Data, Biomass Standard Data_BCK
25> Object name has been changed.
26> (return status = 0)
27>
28> create table Biomass Station
29> (Biomass Station_IId# int not null, /* internal identification */
30> Cruise Number int not null, /* for compatibility */
31> Station Number varchar(8) null,/* Biomass STATION */
32> Longitude float not null, /* Biomass LONGITUDE */
33> Latitude float not null, /* Biomass LATITUDE */
34> Date Time datetime null, /* Biomass DATE */
35> Marsden Square# varchar(8) null)/* Biomass CRUISE */
36>
37> create table Biomass Standard Data
38> (Biomass Standard Data_Id# int not null,/* internal identification */
39> Biomass Station_IId# int not null,/* Biomass DEPTH */
40> Depth int not null,/* Biomass DEPTH */
41> PrDepth int not null,/* Biomass PRDEPTH */
42> ObsType char (1) not null,/* Biomass OBSTYPE */
43> DepthType char (1) not null,/* Biomass DEPTHTYPE */
44> Temperature float null,/* Biomass TEMP */
45> Salinity float null,/* Biomass SALINITY */
46> Oxygen float null,/* Biomass OXYGEN */
47> InPhos float null,/* Biomass INPHOS */
48> Nitrite float null,/* Biomass NITRITE */
49> Nitrate float null,/* Biomass NITRATE */
50> Ammonium float null,/* Biomass AMMONIUM */
51> Silicate float null,/* Biomass SILICATE */
52> Ph float null,/* Biomass PH */
53> Chloride float null,/* Biomass CHLORIDE */
54> Chlorophyll A float null,/* Biomass CHLOROPHYLL A */
55> Sigma_T float null,/* Biomass SIGMA_T */
56> Sigma_T_Obs float null,/* Biomass SIGMA_T_OBS */
57> Sound_Velocity float null,/* Biomass SOUND_VELOCITY */
58> Dynamic Height float null,/* Biomass DYNAMIC HEIGHT */
59> Sigma_Theta float null,/* Biomass SIGMA_THETA */
60> Pot_Temp float null,/* Biomass */
61> Spec_Vol_Anomaly float null,/* SPEC_VOL_ANOMALY */
Delta_D float null /* Biomass DELTA_D */

1> /*
2> ** Definition of primary keys
3> */
4> exec sp_primarykey Biomass_Station, Biomass_Station_Id#
5> exec sp_primarykey Biomass_Standard_Data, Biomass_Standard_Data_Id#
New primary key added.
(return status = 0)
New primary key added.
(return status = 0)
1>
2> /*
3> ** Definition of foreign keys
4> */
5> exec sp_foreignkey Biomass_Standard_Data, Biomass_Station, Biomass_Station_Id#
New foreign key added.
(return status = 0)
1>
2> /*
3> ** Enter user
4> */
5> grant select on Biomass_Station to PHYSIK1
6> grant select on Biomass_Standard_Data to PHYSIK1
This script changes ship names in the table Ship. Some ships were only defined by their NOCD code, now their correct names are received. One name has to be changed.

FILE: SHIPS_UPD050291.SCRIPT

Modified 1991-08-29 by D. Hinz

BEGIN:

use master

if not exists ( select * from master.dbo.sysdatabases
where name = "SouthernOceanDB")
begin
print 'SouthernOceanDB does not exist !!!'
return
end
use SouthernOceanDB

BEGIN:

update Ship
set Ship_Name = 'Goyena'
where Ship_Name = 'Cayena'
(1 row affected)

update Ship
set Ship_Name = 'Billie',
Ship_Code = 'A5N31'
where Ship_Name = 'A5N31'
(1 row affected)

update Ship
set Ship_Name = 'Eldorado',
Ship_Code = 'AEL31'
where Ship_Name = 'AEL31'
(1 row affected)

update Ship
set Ship_Name = 'Goyena',
Ship_Code = 'AGO 8'
where Ship_Name = 'AGO 8'
(1 row affected)

update Ship
set Ship_Name = 'Idaho Standard',
Ship_Code = 'AIS31'
where Ship_Name = 'AIS31'
(1 row affected)
update Ship
   set Ship_Name = 'Mihail Somov',
       Ship_Code = 'ASB90'
where Ship_Name = 'ASB90'
(1 row affected)
if @@error != 0
   rollback transaction
else
   commit transaction
use master

if not exists (select * from master.dbo.sysdatabases
    where name = "SouthernOceanDB")
begin
    print 'SouthernOceanDB does not exist !!!'
    return
end

use SouthernOceanDB

if exists (select * from sysobjects where name like "insertShip")
    drop proc insertShip

create proc insertShip ( @ship varchar(80),
    @country varchar(80),
    @cru int )

    declare @shipid int ,
        @msg varchar(80)

    begin transaction

    if exists ( select Ship_Id#
        from Ship
        where Ship_Name like @ship )
        begin
            select @shipid = (select Ship_Id#
                from Ship
                where Ship_Name like @ship )
        end
    else
        begin
            select @shipid = (select max(Ship_Id#)
                from Ship
                where Ship_Id# < 100000)
            if @shipid is null
                select @shipid = 1
            else
                select @shipid = @shipid + 1
        end
    insert Ship
    values (@shipid, @ship, @country, NULL)
insert Cruise values (@cru, @shipid)
select Cruise_Number, Ship from Aari_Cruise where Ship like @ship
print @msg
if @@error != 0
  rollback transaction
  select @msg = "ERROR: Nothing inserted"
else
  begin
    commit transaction
    select @msg = "Ship " + @ship + " and Cruise " +
    convert(varchar(20), @cru) + " inserted"
  end
print @msg
end
if not exists (select * from sysobjects where name like "SouthernOceanDB")
begin
print "SouthernOceanDB not found"
return
end
(0 rows affected)
SouthernOceanDB not found

use SouthernOceanDB

if exists (select * from sysobjects where name like "tempships")
drop table tempships

create table tempships ( Id# int,
 cn int,
 ship varchar(80),
 country varchar(80))

if exists (select * from sysobjects where name like "insert_into_cruise")
drop procedure insert_into_cruise

create proc insert_into_cruise
( @cruisen int, @ship varchar(20), @scountry varchar(20))

declare @sid int

select @sid = Ship_Id#
from Ship
where Ship_Name like @ship and
 Country like @scountry

if @sid is null
begin
select @sid = max(Ship_Id#) + 1
from Ship
where Ship_Id# < 10000
insert into Ship values(@sid, @ship, @scountry, null)
end
insert into Cruise values (@cruisen, @sid)

if exists (select * from sysobjects where name like "insert_new_ships")
drop procedure insert_new_ships

create proc insert_new_ships
as
/*
** insert ship from a table tempships into table Ship and
** the pair Ship_id# and Cruise_Number# into Cruise.
** This procedure requires the procedure insert_into_cruise
*/
declare @zaehler int,
    @cruisen  int,
    @ship    varchar(20),
    @scountry varchar(20),
    @m       int
select @zaehler = 1
select @m = max(Id#) from tempships
while @zaehler <= @m
begin
    select @cruisen = cn,
        @ship = ship,
        @scountry = country
    from tempships
    where Id# = @zaehler
    exec insert_into_cruise @cruisen, @ship, @scountry
    select @zaehler = @zaehler+1
end
/* SYBASE script

** Created: 1991-08-28
**
** Creator: L.-P. Kurdelski
**
*/

use SouthernOceanDB

if exists (select *
from sysobjects
where name like "updateCruise")

drop proc updateCruise

create procedure updateCruise

@cru int,
@ship varchar(80),
@country varchar(80)

as

begin

select @s = Ship_Id#
from Ship
where Ship_Name like @ship and
Country like @country

update Cruise
set Ship_Id# = @s
where Cruise_Number# = @cru
end
Changes in table Cruise

Ship_Id# of selected Cruise_Number# is changed according to Country

Ship_Id# of the table Ship where Ship_Name and Country are unknown

begin transaction

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 202

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 266

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 410

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 424

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 426

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 634

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 635

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 636

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 637

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 638

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 639

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 643

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 678

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 891

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 895

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 896

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 910
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 1078

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 1393

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 2102

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 2157

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 2162

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 2326

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 2968

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 2970

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 3020

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 3070

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 3359

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 3482

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 3494

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 3927

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 3958

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 3995

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 3996

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 3997

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 4031

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 4063

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 4231

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 4244
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 4375
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 4395
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 4564
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 4576
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 4597
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 4813
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 4835
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 4859
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 5027
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 5223
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 5367
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 5369
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 5371
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 5373
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 5455
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 5547
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 5552
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 5581
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 5596
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 5610
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 5610
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 6424
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 6468
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 6525

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 7660

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 8683

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 8755

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 9230

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 9443

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 9689

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 10109

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 10167

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 10186

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 10279

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 10282

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 10291

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 10341

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 10350

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 10422

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 10522

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 11319

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 12031

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 12220

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 12864

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 13968
update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 14539

update Cruise
set Ship_Id# = 1000000 where Cruise_Number# = 15000
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)

1> if @@error != 0
2> begin
3> rollback transaction
4> print "rollback"
5> end
6> else
7> begin
8> commit transaction
9> print "commit"
10> end
11> commit

commit
SYBASE script by D. Hinz, L.-P. Kurdelski

Creation date: 1991-07-22

Abstract: Inserting chlorophyll values for BIOMASS from a source given by F. Brandini.

Preparing the insertion procedure.

use SouthernOceanDB

if exists (select * from sysobjects where name like "insertChlorophyllA")
drop proc insertChlorophyllA

create proc insertChlorophyllA

(@sn varchar(8),  /*Station Number*/
@d int,          /*Depth*/
@c_a float      /*Chlorophyll_A*/
)

as

begin

declare

/*Tabelle Biomass_Station*/

@bsid int,       /*Biomass_Station_Id*/
@bsidd int,      /*Biomass_Station_Id*/
@cnu int,        /*Cruise_Number*/
@lat float,      /*Latitude*/
@long float,     /*Longitude*/
@d_t datetime,   /*DateTime*/
@m_s int,        /*Marsden_Square*/
@cNa varchar(8), /*Cruise_Name*/

/*Tabelle Biomass_Standard_Data*/

@bsdid int,      /*Biomass_Standard_Data_Id*/
@d int,          /*Depth*/
@d_p int,        /*PrDepth*/
@ot char(1),     /*ObsType*/
@dt char(1),     /*DepthType*/
@temp float,     /*Temperature*/
@sal float,      /*Salinity*/
@ox float,       /*Oxygen*/
@ip float,       /*InPhos*/
@ni float,       /*Nitrite*/
@na float,       /*Nitrate*/
@am float,       /*Ammonium*/
@si float,       /*Silicate*/
@ph float,       /*PH*/
@chl float,      /*Chloride*/
@c_a float,      /*Chlorophyll_A*/
@st float,       /*Sigma_T*/
@sto float,      /*Sigma_T_Obs*/
@sv float,       /*Sound_Velocity*/
@dh float,       /*Dynamic_Height*/
@sth float,      /*Sigma_Theta*/
@p_t float,      /*Pot_Thick*/
@sca float,      /*Spec_Col_Anomaly*/
@d_d float       /*Delta_D*/
select @bsid = Biomass_Station_Id#, @cnu = Cruise_Number, @sn = Station_Number, @lat = Latitude, @long = Longitude, @dt = Date_Time, @m_s = Marsden_Square#, @cna = Cruise_Name from Biomass_Station where Cruise_Name like "BES%" and Station_Number-like @sn and Biomass_Station_Id# < 880000

select @bsdid = Biomass_Standard_Data_Id#, @bsidd = Biomass_Station_Id#, @dd = Depth, /*wird ersetzt!!!*/ @pd = PrDepth, @ot = ObsType, @dt = DepthType, @temp = Temperature, @sal = Salinity, @ox = Oxygen, @ip = InPhos, @ni = Nitrite, @am = Ammonium, @na = Nitrate, @am = Ammonium, @si = Silicate, @ph = PH, @chl = Chloride, @c_aa = Chlorophyll_A, /*wird ersetzt!!!*/ @st = Sigma_T, @sto = Sigma_T_Obs, @sv = Sound_Velocity, @dh = Dynamic_Height, @st = Sigma_Theta, @pt = Pot_Temp, @sca = Spec_Col_Anomaly, @d_d = Delta_D from Biomass_Station where Biomass_Station_Id# = @bsid and Depth = @d and Biomass_Standard_Data_Id# < 880000

select @bsid = @bsid + 80000

select @bsdid = @bsdid + 80000

/*
**Die Stationen werden dupliziert und, mit einer neuen Kennung versehen,
**als neue Station eingefügt. Dies geschieht aus Gründen der
**Vollständigkeit. Diese Stationen können zu einem späteren Zeitpunkt
**wieder gelöscht werden, wenn geklärt ist, dass die Stationsdaten so
**belassen werden können.
*/

if not exists (select * from Biomass_Station where Biomass_Station_Id# = @bsid)

begin

insert Biomass_Station (Biomass_Station_Id#, Cruise_Number, Station_Number, Latitude, Longitude, Date_Time,
Marsden_Square#, Cruise_Name)
values (@bsid, @cnu, @sn, @lat, @long, @d_t, @m_s, @cna)
end
/* **Die Daten werden dupliziert und, mit einer neuen Kennung versehen, als **neuer datensatz in die Tabelle eingefügt. Dies geschieht, da die **ursprünglichen Stationsdaten KEINE Chlorophyll A-Werte enthielten, **aber die neuen Werte als zu den alten Station gehören betrachtet werden. **zu einem späteren Zeitpunkt können diese Stationsdaten wieder gelöscht **werden, wenn die BIOMASS-Daten in der nun gültigen Form gehalten **werden sollen. */
insert Biomass Standard Data
(Biomass Standard Data_Id#, Biomass_Station_Id#, Depth, PrDepth, Obstype, DepthType, Temperature, Salinity, Oxygen, InPhos, Nitrate, Nitrite, Ammonium, Silicate, PH, Chloride, Chlorophyll_A, Sigma_T, Sigma_T_Obs, Sound_Velocity, Dynamic_Height, Sigma_Theta, Pot_Temp, Spec_Col_Anomaly, Delta_D)
values (@bsdid, @bsid, @d, @pd, @ot, @dt, @temp, @sal, @ox, @ip, @ni, @na, @am, @si, @ph, @chl, @c_a, @st, @sto, @s_v, @d_h, @sth, @p_t, @sca, @d_d)
end
/* File: Muench_copy_all.script - LOG */

use SouthernOceanDB
create procedure Muench_copy_all
as
/* Fill Station and Standard Data with Muench Data */
/* L.-P. Kurdelski 1991-08-12 */
begin
insert Station (Station_Id#, Cruise_Number, Station_Number,
Date_Time, Longitude, Latitude, Bottom_Depth, Max_Obse_Depth,
Number_Obse, Marsden_Square#)
select Muench_Station_Id#, Cruise_Number, Station_Number,
Date_Time, Longitude, Latitude, Bottom_Depth, Max_Obse_Depth,
Number_Obse, Marsden_Square#
from Muench_Station
insert Standard_Data (Standard_Data_Id#, Station_Id#, Depth, Temperature, Salinity)
select Muench_Standard_Data_Id#, Muench_Station_Id#, Depth, Temperature, Salinity
from Muench_Standard_Data
end
use SouthernOceanDB
update Muench_Station
set Date_Time = NULL
where Date_Time = "Jan 01 1900 00:00:00.0"
(0 rows affected)
update Muench_Station
set Bottom_Depth = NULL
where Bottom_Depth = -9999
(208 rows affected)
update Muench_Station
set Temperature = NULL
where Temperature < -10.0
(0 rows affected)
update Muench_Station
set Salinity = NULL
where Salinity < 0
(0 rows affected)
update Muench_Station
set Marsden_Square# = NULL
where Marsden_Square# = -9999
(208 rows affected)
/** File: Biomass_copy_all.script */
use SouthernOceanDB
if exists (select * from sysobjects where name like "Biomass_copy_all")
drop procedure Biomass_Copy_all
create procedure Biomass_copy_all
as
/* L.-P. Kurdelski 1991-08-12 */
begin
insert Station (Station Id#, Cruise Number, Date Time, Longitude, Latitude, Marsden Square#)
select Biomass Station Id#, Cruise Number, Date Time, Longitude, Latitude, Marsden Square#
from Biomass Station
insert Standard Data (Standard Data Id#, Station Id#, Depth, Temperature, Salinity, Oxygen)
select Biomass Standard Data Id#, Biomass Station Id#, Depth, Temperature, Salinity, Oxygen
from Biomass Standard Data
end
begin transaction
declare @sid int

select @sid = Ship_Id# from Ship
where Ship_Name like "Evrika"

update Cruise
set Ship_Id# = @sid where Cruise_Number# = 42
(rows affected)

Msg 515, Level 16, State 3:
Server 'SYBASE401', Line 7:
Attempt to insert the value NULL into column 'Ship_Id#', table 'SouthernOceanDB.dbo.Cruise'; column does not allow nulls. Update fails.
(0 rows affected)

if @@error != 0
begin
rollback transaction
print "rollback"
end
else
begin
commit transaction
print "commit"
end
rollback

begin transaction
declare @sid int

select @sid = Ship_Id# from Ship
where Ship_Name like "Babchisarai"

update Cruise
set Ship_Id# = @sid where Cruise_Number# = 2501
(0 rows affected)

Msg 515, Level 16, State 3:
Server 'SYBASE401', Line 7:
Attempt to insert the value NULL into column 'Ship_Id#', table 'SouthernOceanDB.dbo.Cruise'; column does not allow nulls. Update fails.
(0 rows affected)

if @@error != 0
begin
rollback transaction
print "rollback"
end

end
else
begin transaction
begin
declare @sid int

select @sid = Ship_Id# from Ship
where Ship_Name like "Africana II"

update Cruise
set Ship_Id# = @sid where Cruise_Number# = 5406
(1 row affected)
(1 row affected)

if @@error != 0	hen
rollback transaction
print "rollback"
end
end
else
begin
commit transaction
print "commit"
end
commit
Use SouthernOceanDB

Update of illegal values in the database SouthernOceanDB tables Biomass Station and Biomass Standard Data. Illegal values are stated by the number -9999.0. This is updated to NULL.

Chlorophyll A for values less than or equal to 0.0 set to NULL
Chlorophyll A values for cruise ACS1 divided by 10
(Hint by F. Brandini)

illegal values for Temperature, Salinity etc. i.e. values which are not possible in nature in this cases

Update of illegal values in the database SouthernOceanDB tables Biomass Station and Biomass Standard Data. Illegal values are stated by the number -9999.0. This is updated to NULL.

Chlorophyll A for values less than or equal to 0.0 set to NULL
Chlorophyll A values for cruise ACS1 divided by 10
(Hint by F. Brandini)

illegal values for Temperature, Salinity etc. i.e. values which are not possible in nature in this cases

Update of illegal values in the database SouthernOceanDB tables Biomass Station and Biomass Standard Data. Illegal values are stated by the number -9999.0. This is updated to NULL.

Chlorophyll A for values less than or equal to 0.0 set to NULL
Chlorophyll A values for cruise ACS1 divided by 10
(Hint by F. Brandini)

illegal values for Temperature, Salinity etc. i.e. values which are not possible in nature in this cases

Update of illegal values in the database SouthernOceanDB tables Biomass Station and Biomass Standard Data. Illegal values are stated by the number -9999.0. This is updated to NULL.

Chlorophyll A for values less than or equal to 0.0 set to NULL
Chlorophyll A values for cruise ACS1 divided by 10
(Hint by F. Brandini)

illegal values for Temperature, Salinity etc. i.e. values which are not possible in nature in this cases

Update of illegal values in the database SouthernOceanDB tables Biomass Station and Biomass Standard Data. Illegal values are stated by the number -9999.0. This is updated to NULL.

Chlorophyll A for values less than or equal to 0.0 set to NULL
Chlorophyll A values for cruise ACS1 divided by 10
(Hint by F. Brandini)

illegal values for Temperature, Salinity etc. i.e. values which are not possible in nature in this cases
If there are Salinity values below 32 and above 0 and it seems that they are divided by factor 10 multiply them by 10 and it is ok.

update Biomass Standard Data
set Salinity = Salinity * 10
where Salinity < 30
(997 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Oxygen = NULL
where Oxygen = -9999
(16490 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Oxygen = NULL
where Oxygen < 0
(0 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Nitrite = NULL
where Nitrite = -9999
(15700 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Nitrite = NULL
where Nitrite < 0
(11 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Nitrate = NULL
where Nitrate = -9999
(14607 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Nitrate = NULL
where Nitrate < 0
(0 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Ammonium = NULL
where Ammonium = -9999
(19286 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Ammonium = NULL
where Ammonium < 0
(0 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Silicate = NULL
where Silicate = -9999
(14598 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Silicate = NULL
4> where Silicate < 0  
   (0 rows affected)
1> dump tran SouthernOceanDB with truncate_only
1>
2> update Biomass Standard Data
3> set Ph = NULL
4> where Ph = -9999  
   (19723 rows affected)
1> dump tran SouthernOceanDB with truncate_only
1>
2> update Biomass Standard Data
3> set Ph = NULL
4> where Ph < 0  
   (0 rows affected)
1> dump tran SouthernOceanDB with truncate_only
1>
2> update Biomass Standard Data
3> set Chloride = NULL
4> where Chloride < 0  
   (20052 rows affected)
1> dump tran SouthernOceanDB with truncate_only
1>
2> update Biomass Standard Data
3> set ChlorophyllA = ChlorophyllA / 10.0
4> where Biomass Station Id# in  
   (select Biomass Station Id#  
    from Biomass Station  
    where Cruise_Name like "ACS1")  
   (906 rows affected)
1> dump tran SouthernOceanDB with truncate_only
1>
2> update Biomass Standard Data
3> set ChlorophyllA = ChlorophyllA / 10.0
4> where ChlorophyllA < 0  
   (6 rows affected)
1> dump tran SouthernOceanDB with truncate_only
1>
2> update Biomass Standard Data
3> set Sigma_T = NULL
4> where Sigma_T = -9999  
   (18470 rows affected)
1> dump tran SouthernOceanDB with truncate_only
1>
2> update Biomass Standard Data
3> set Sigma_T = NULL
4> where Sigma_T < 0  
   (1617 rows affected)
1> dump tran SouthernOceanDB with truncate_only
1>
2> update Biomass Standard Data
3> set Sigma_T_Obs = NULL
4> where Sigma_T_Obs = -9999  
   (15441 rows affected)
1> dump tran SouthernOceanDB with truncate_only
1>
2> update Biomass Standard Data
3> set Sigma_T_Obs = NULL
4> where Sigma_T_Obs < 0  
   (0 rows affected)
1> dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Sound_Velocity = NULL
where Sound_Velocity = -9999
(16876 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Sound_Velocity = NULL
where Sound_Velocity < 0
(0 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Dynamic_Height = NULL
where Dynamic_Height = -9999
(9557 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Sigma_Theta = NULL
where Sigma_Theta = -9999
(4748 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Sigma_Theta = NULL
where Sigma_Theta < 0
(1 row affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Pot_Temp = NULL
where Pot_Temp = -9999
(4162 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Pot_Temp = NULL
where Pot_Temp < -9 /* !!!!! see log from 1991-07-15 */
(0 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Spec_Vol_Anomaly = NULL
where Spec_Vol_Anomaly = -9999
(13505 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set Delta_D = NULL
where Delta_D = -9999
(19982 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set InPhos = NULL
where InPhos = -9999
(14608 rows affected)
dump tran SouthernOceanDB with truncate_only
update Biomass Standard Data
set InPhos = NULL
where InPhos < 0
(0 rows affected)
dump tran SouthernOceanDB with truncate_only
Two ships and tables. First temptab. Last cruise table.

lpk table tempships is only truncated

if not exists (select * from sysobjects where name like "SouthernOceanDB")
begin
print "SouthernOceanDB not found"
return
end
(0 rows affected)
SouthernOceanDB not found
use SouthernOceanDB
if not exists (select * from sysobjects where name like "insert_into_cruise")
begin
print "procedure insert_into_cruise expected"
return
end
(0 rows affected)
if not exists (select * from sysobjects where name like "insert_new_ships")
begin
print "procedure insert_new_ships expected"
return
end
(0 rows affected)
if not exists (select * from sysobjects where name like "temptab")
begin
print "Table tempships does not exist."
return
end
(0 rows affected)

/* clean up tempships */
truncate table tempships
begin transaction

declare @zaehler int
select @zaehler = 1
insert tempships values (@zaehler, -25026, "Fuji", "Japan")
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, -25027, "Fuji", "Japan")
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, -25028, "Shirase", "Japan")
14> select @zaehler = @zaehler + 1

15> insert tempships values(@zaehler, -25029, "Shirase", "Japan")

16> select @zaehler = @zaehler + 1

17> insert tempships values(@zaehler, -25030, "Shirase", "Japan")

18> select @zaehler = @zaehler + 1

19> insert tempships values(@zaehler, -25031, "Shirase", "Japan")

20> select @zaehler = @zaehler + 1

21> insert tempships values(@zaehler, -25032, "Shirase", "Japan")

22> select @zaehler = @zaehler + 1

23> insert tempships values(@zaehler, -25033, "Shirase", "Japan")

24> exec insert_new_ships

25> if @@error != 0

26> begin

27> print "transaction aborted"

28> rollback transaction

29> end

30> else

31> begin

32> print "transaction successful"

33> commit transaction

34> end

35> truncate table tempships
Two ships and tables. First temptab. Last cruise table.
lpk table tempships is only truncated
eight cruises should be inserted into ship and cruise
tables. First truncate table temptab then insert the ships into
temptab. Last use procedure insert_new_ships to insert ships into

if not exists (select* from sysobjects where name like "SouthernOceanDB")
begin
print "SouthernOceanDB not found"
return
end

use SouthernOceanDB

if not exists (select* from sysobjects where name like "insert_into_cruise")
begin
print "procedure insert_into_cruise expected"
return
end

if not exists (select* from sysobjects where name like "insert_new_ships")
begin
print "procedure insert_new_ships expected"
return
end

if not exists (select* from sysobjects where name like "temptab")
begin
print "Table tempships does not exist."
return
end

declare @zaehler int

insert tempships values (@zaehler, -25026, "Fuji", "Japan")
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, -25027, "Fuji", "Japan")
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, -25028, "Shirase", "Japan")

*/

clean up tempships
*/

truncatet table tempships
begin transaction
*/

filling tempships with new ships
*/

-- clean up tempships
*/

declare @zaehler int

insert tempships values (@zaehler, -25026, "Fuji", "Japan")
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, -25027, "Fuji", "Japan")
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, -25028, "Shirase", "Japan")
14> select @zaehler = @zaehler + 1
15> insert tempships values(@zaehler, -25029, "Shirase", "Japan")
16> select @zaehler = @zaehler + 1
17> insert tempships values(@zaehler, -25030, "Shirase", "Japan")
18> select @zaehler = @zaehler + 1
19> insert tempships values(@zaehler, -25031, "Shirase", "Japan")
20> select @zaehler = @zaehler + 1
21> insert tempships values(@zaehler, -25032, "Shirase", "Japan")
22> select @zaehler = @zaehler + 1
23> insert tempships values(@zaehler, -25033, "Shirase", "Japan")
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
   (1 row affected)
24> exec insert_new_ships
   (return status = 0)
25> if @@error != 0
26> begin
27> print "transaction aborted"
28> rollback transaction
29> end
30> else
31> begin
32> print "transaction successfull"
33> commit transaction
34> end
35> transaction successfull
36> truncate table tempships
Dieses Script laedt die Cruise Number aus der Tabelle Aari Cruise BCK sowie die entsprechende Ship_Id# aus der Tabelle Ship in die Tabelle Cruise.

FILE: aari_and_ship_to_cruise.script

creation date: 1991-08-21

author: D. Hinz

but it really works

use SouthernOceanDB

if exists (select * from sysobjects where name like "aari_and_ship_to_cruise"

drop proc aari_and_ship_to_cruise

create proc aari_and_ship_to_cruise

as

insert into Cruise (Cruise Number#, Ship_Id#)

select Cruise Number, Ship_Id#

from Ship s, Aari_Cruise Bck a

where s.Ship_Name-like a.Ship

and s.Country like a.Country
if exists (select * from sysobjects where name like "Aari_Cruise2")
  drop view Aari_Cruise2
create view Aari_Cruise2 (Cruise_Number, Ship, Country)
as
  select c.Cruise_Number#, s.Ship_Name, s.Country
  from Ship s, Cruise c
  where s.Ship_Id# = c.Ship_Id#
grant select on Aari_Cruise2 to SOCEAN
SYBASE script by L.-P. Kurdelski
** Inserting new Cruises to Cruise and Ship from Kuropatkin Data
** created 1991-08-28
** FILE: ADD_KUROPATKIN_SHIPS.SCRIPT
*/

use master
if not exists (select *
from master.dbo.sysdatabases
where name = "SouthernOceanDB")
begin
print 'SouthernOceanDB does not exist !!!'
return
end

use SouthernOceanDB
(0 rows affected)

exec insertShip 'N.Kuropatkin', 'USSR', -23011
Ship N.Kuropatkin and Cruise -23011 inserted

(0 rows affected, return status = 0)
SYBASE script

Created: 1991-08-29

Creator: L.-P. Kurdelski

FILE: changeShipNameInCruise.script

** Change the ship ids in table Cruise due to the change of name from old ship to new ship.

* An old ship name, a new ship name and two cruise number are used as parameters. If the second number is 0 then only the first number is used (just as you expect). Otherwise both numbers are used in a junction.

** If both cruise numbers are 0 it is ok.

** if the first cruise number is 0 and the second is not 0 this is an error.

** Summary

<table>
<thead>
<tr>
<th>cru1</th>
<th>cru2</th>
<th>comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>change all ship id for the old ship</td>
</tr>
<tr>
<td>1=0</td>
<td>0</td>
<td>... only for cruise cru1</td>
</tr>
<tr>
<td>1=0</td>
<td>1=0</td>
<td>... only for cruise cru1 and cru2</td>
</tr>
</tbody>
</table>

**

Summary

<table>
<thead>
<tr>
<th>cru1</th>
<th>cru2</th>
<th>comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>change all ship id for the old ship</td>
</tr>
<tr>
<td>1=0</td>
<td>0</td>
<td>... only for cruise cru1</td>
</tr>
<tr>
<td>1=0</td>
<td>1=0</td>
<td>... only for cruise cru1 and cru2</td>
</tr>
</tbody>
</table>

illegal

declare @sid int,
@sidalt int,
@msg varchar(80)

begin transaction
select @sidalt = Ship_Id#
from Ship
where Ship_Name like @oldShip

select @sid = Ship_Id#
from Ship
where Ship_Name like @newShip

if @sid is null
begin
  select @sid = Ship_Id#
  from Ship
  where Ship_Name like "unknown"

select @msg = "Unknown ship inserted for " + @newShip
print @msg
end

if @cru2 = 0
begin
  if @cru1 = 0
  begin
    
  end
  else
  begin
    update Cruise
    set Ship_Id# = @sid
    where Ship_Id# = @sidalt
  end
end
else
begin
  update Cruise
  set Ship_Id# = @sid
  where Ship_Id# = @sidalt and
  Cruise_Number# = @cru1
end
end
if @@error = 0
begin
  rollback transaction
  return
end
else
begin
  update Cruise
  set Ship_Id# = @sid
  where Ship_Id# = @sidalt and
  (Cruise_Number# = @cru1 or Cruise_Number# = @cru2)
end
end
if @error != 0
begin
  rollback transaction
  select @msg = "ERROR occured during update of " + @oldShip
end
else
begin
  commit transaction
  select @msg = "Update of " + @oldShip + " to " + @newShip + " successful."
end
print @msg
end
use SouthernOceanDB

exec copy_gordon_ship_to_ship
(return status = 0)

exec copy_gordon_cruise_to_cruise
(return status = 0)
create procedure copy_gordon_ship_to_ship
as
/* Copy the Gordon_Ships and relations to the table Ship */
begin
insert into Ship (Ship Id#, Ship Name, Country, Ship Code)
select s.Ship Id#, s.Ship Name, s.Country, s.Ship Code
from Gordon_Ship s
end

create procedure copy_gordon_cruise_to_cruise
as
/* Copy the Gordon_Ships and relations to the table Ship */
begin
insert into Cruise (Ship Id#, Cruise Number#)
select s.Ship Id#, c.Gordon Cruise Id#
from Gordon_Ship s, Gordon Cruise c
where s.Ship Code like c.Ship Code
end
1> /*
2> **
3> **
4> **
5> **
6> **
7> **
8> */
9> /*
10> Adding ships and cruises due to Haines data to the
11> tables Ship and Cruise. The data are read from DBTEXT1.DAT
12> from V. Guretzky. Some checks marked that the cruise
13> numbers must be modified by adding 1099 to the cruise number.
14> */
15> use SouthernOceanDB
16>
17> if not exists (select * from sysobjects where name like "Ship")
18> begin
19>   print "Table Ship not found"
20>   return
21> end
22> (0 rows affected)
23>
24> if not exists (select * from sysobjects where name like "Cruise")
25> begin
26>   print "Table Cruise not found"
27>   return
28> end
29> (0 rows affected)
30>
31> /*
32> ** clean up tempships
33> **/
34> truncate table tempships
35> begin transaction
36> declare @zaehler int
37> select @zaehler = 0
38> /*
39> **
40> select @zaehler = @zaehler + 1
41> insert tempships values (@zaehler, 54001, "Westwind", "USA")
42> /*
43> **
44> select @zaehler = @zaehler + 1
45> insert tempships values (@zaehler, 54002, "Melville", "USA")
46> /*
47> **
48> select @zaehler = @zaehler + 1
49> insert tempships values (@zaehler, 54003, "Melville", "USA")
50> /*
51> **
52> select @zaehler = @zaehler + 1
53> insert tempships values (@zaehler, 54004, "Atlantis II", "USA")
54> /*
55> **
56> select @zaehler = @zaehler + 1
57> insert tempships values (@zaehler, 54005, "Thomas Washington", "USA")
58> /*
59> */
** 6 54006 Meiring Naude RSA

** 7 54007 Discovery II UK

** 8 54008 Knorr USA

exec insert new ships
(return status = 0)

if @@error != 0
begin
print "transaction aborted"
rollback transaction
end
else
begin
print "Ships inserted"
commit transaction
end
Ships inserted

truncate table tempships
/* SYBASE Script by L.-P. Kurdelski
** FILE: NOWLIN_SHIPS.SCRIPT
** CREATED: 1991-08-30
** MODIFIED:
** Insert new ships
** The ships are taken from the Nowlin dataset.
*/

if not exists ( select * from sysobjects where name like "SouthernOceanDB"
begin
print "SouthernOceanDB not found"
return
end
0 rows affected)
SouthernOceanDB not found

use SouthernOceanDB

if not exists ( select * from sysobjects where name like "insert_into_cruise"
begin
print "Procedure inser_into_cruise missing."
return
end
0 rows affected)

if not exists ( select * from sysobjects where name like "insert_new_ships"
begin
print "Procedure insert_new_ships missing."
return
end
0 rows affected)

if exists ( select * from sysobjects where name like "tempships"
begin
truncate table tempships
end
else
begin
print "No table tempships"
return
end

declare @zaehler int

select @zaehler = 0
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, 54000, "Atlantis II", "USA")
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, 54001, "Knorr", "USA")
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, 54002, "AGO 8", "Argentina")
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, 54003, "Atka", "USA")
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, 54004, "Discovery II", "U.K.")
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, 54005, "AHZ49", "Japan")
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, 54006, "Melville", "USA")
select @zaehler = @zaehler + 1
insert tempships values (@zaehler, 54007, "A5N31", "USA")
select @zaehler = @zaehler + 1
22> insert tempships values (@zaehler, 54008, "Hudson", "Canada")
23> select @zaehler = @zaehler + 1
24> insert tempships values (@zaehler, 54009, "Islas Orcadas", "Argentina")
25> select @zaehler = @zaehler + 1
26> insert tempships values (@zaehler, 54010, "Eltanin", "USA")
27> select @zaehler = @zaehler + 1
28> insert tempships values (@zaehler, 54011, "Thomas Washington", "USA")
29> select @zaehler = @zaehler + 1
30> insert tempships values (@zaehler, 54012, "Glacier", "USA")
31> select @zaehler = @zaehler + 1
32> insert tempships values (@zaehler, 54013, "Discovery II", "U.K.")
33> select @zaehler = @zaehler + 1
34> insert tempships values (@zaehler, 54014, "AEL31", "USA")
35> select @zaehler = @zaehler + 1
36> insert tempships values (@zaehler, 54015, "R.D.Conrad", "USA")
37> select @zaehler = @zaehler + 1
38> insert tempships values (@zaehler, 54016, "AVZ90", "USSR")
39> select @zaehler = @zaehler + 1
40> insert tempships values (@zaehler, 54017, "ASB90", "USSR")
41> select @zaehler = @zaehler + 1
42> insert tempships values (@zaehler, 54018, "AIS31", "USA")
43> select @zaehler = @zaehler + 1
44> insert tempships values (@zaehler, 54019, "AAQ06", "FRG")
45> select @zaehler = @zaehler + 1
46> insert tempships values (@zaehler, 54020, "AMT06", "FRG")
begin transaction
exec insert new ships
print "transaction successful!"
end

if @@error != 0
begin
print 'transaction aborted'
rollback transaction
end
else
begin
print 'transaction successful'
commit transaction
end

truncate table tempships
print "table tempships truncated"
/*
** SYBASE script
**
** FILE: unknown_ship.script
**
** CREATION: 1991-08-19 lpk
**
** Insert the "unknown" ship into the ship table
*/

if not exists (select * from sysobjects where name like "SouthernOceanDB")
begin
    print "SouthernOceanDB not found"
    return
end

(0 rows affected)
SouthernOceanDB not found

use SouthernOceanDB

insert Ship
values (1000000,"unknown","unknown",NULL)

(1 row affected)
/* Sybase script
** Written 1991-06-11 by Jens Begemann and L.-P. Kurdelski
** Modified 1991-08-29 by D. Hinz and L.-P. Kurdelski
**(uebersichtlicher!)
** old version: FILE UpdateShipCruise100691.script
** new version: FILE UpdateShipCruise290891.script
** Update of Ship and Cruise in SouthernOceanDB
** Data from V. Guretzky
*/
/** assume SouthernOceanDB is available */
use SouthernOceanDB

exec changeShipNameInCruise "Prof. Zubov", "Investigator", 643, 0
Update of Prof. Zubov to Investigator successful.
(return status = 0)

exec changeShipNameInCruise "Prof. Zubov", "Gascoyne", 424, 0
Update of Prof. Zubov to Gascoyne successful.
(return status = 0)

exec changeShipNameInCruise "Prof. Zubov", "Melville", 12031, -21814
Update of Prof. Zubov to Melville successful.
(return status = 0)

exec changeShipNameInCruise "Prof. Zubov", "Edisto", 3482, 0
Update of Prof. Zubov to Edisto successful.
(return status = 0)

exec changeShipNameInCruise "Prof. Zubov", "Hudson", 6468, 0
Update of Prof. Zubov to Hudson successful.
(return status = 0)

exec changeShipNameInCruise "Prof. Zubov", "Carnegie", 1393, 0
Update of Prof. Zubov to Carnegie successful.
(return status = 0)

exec changeShipNameInCruise "Prof. Zubov", "Hakuho Maru", 5547, 0
Update of Prof. Zubov to Hakuho Maru successful.
(return status = 0)

exec changeShipNameInCruise "Kokar", "Ob", 0, 0
Update of Kokar to Ob successful.
(return status = 0)

exec changeShipNameInCruise "Lesnoi", "Eltanin", -667, 0
Update of Lesnoi to Eltanin successful.
(return status = 0)

exec changeShipNameInCruise "Walter Herwig", "Discovery II", -1123, 0
Update of Walter Herwig to Discovery II successful.
(return status = 0)

exec changeShipNameInCruise "Dolphin", "Cap. Canepa", -1809, 0
Update of Dolphin to Cap. Canepa successful.
(return status = 0)

exec changeShipNameInCruise "Atlantis", "Atlantis II", -180, 0
Update of Atlantis to Atlantis II successful.
(return status = 0)
2> exec changeShipNameInCruise "Schwabenland", "Discovery II", -12, 0
Update of Schwabenland to Discovery II successful.
(return status = 0)
1> 2> exec changeShipNameInCruise "Kvant", "Salehard", -1117, 0
Update of Kvant to Salehard successful.
(return status = 0)
1> 2> declare @sid int,
    @sidalt int
4> select @sid = Ship_Id# from Ship where Ship_Name like "Gen. San-Martin"
5> select @sidalt = Ship_Id# from Ship
    where Ship_Name Tike "unknown" and
    Ship_Id# in (select Ship_Id# from Cruise
      where Cruise_Number# = -1929)
10> update Cruise
11> set Ship_Id# = @sid
12> where Ship_Id# = @sidalt and Cruise_Number# = -1929
(1 row affected)
(1 row affected)
(1 row affected)
1> 2> exec changeShipNameInCruise "Kiwi", "Tui", -21649, 0
Update of Kiwi to Tui successful.
(return status = 0)
1> 2> exec changeShipNameInCruise "Burton Island", "Anton Bruun", -816, 0
Update of Burton Island to Anton Bruun successful.
(return status = 0)
1> 2> exec changeShipNameInCruise "Africana II", "Natal", 5406, 0
Update of Africana II to Natal successful.
(return status = 0)
1> 2> exec changeShipNameInCruise "Poceidon", "Umitaka-Maru", -1934, 0
Update of Poceidon to Umitaka-Maru successful.
(return status = 0)
1> 2> exec changeShipNameInCruise "Mitory-Maru", "Eltanin", -1677, 0
Update of Mitory-Maru to Eltanin successful.
(return status = 0)
1> 2> exec changeShipNameInCruise "Atlantis II", "Atka", -180, 0
Update of Atlantis II to Atka successful.
(return status = 0)
1> 2> exec changeShipNameInCruise "Discovery II", "Discovery", -21137, 0
Update of Discovery II to Discovery successful.
(return status = 0)
1> 2> exec changeShipNameInCruise "Glacier", "Burton Island", 619, 0
Update of Glacier to Burton Island successful.
(return status = 0)
1> 2> exec changeShipNameInCruise "Adm. Moucheeg", "Norvegia", -97, 0
Update of Adm. Moucheeg to Norvegia successful.
(return status = 0)
1> 2> exec changeShipNameInCruise "Olonets", "Diamantina", 430, 0
Update of Olonets to Diamantina successful.
(return status = 0)
1> 2> exec changeShipNameInCruise "Evrice", "Vityaz", 42, 0
Update of Evrica to Vityaz successful.
(return status = 0)

exec changeShipNameInCruise "Ushakov", "Slava", 9768, 0
Update of Ushakov to Slava successful.
(return status = 0)

exec changeShipNameInCruise "Billie", "Discovery", 54007, 0
Update of Billie to Discovery successful.
(return status = 0)
exec changeshipnamelnCruise "AHZ49", "Fuji-Maru", 54005, 0
Update of AHZ49 to Fuji-Maru successful.
(return status = 0)
update Ship
set Ship_Code = "AHZ49"
where Ship_Name like "Fuji-Maru" and Country like "Japan"
(1 row affected)
delete Ship
where Ship_Name like "AHZ49" and Country like "Japan"
(1 row affected)
exec changeshipnamelnCruise "Anton", "Anton Bruun", -21047, 0
Update of Anton to Anton Bruun successful.
(return status = 0)
delete Ship
where Ship_Name like "Anton" and Country like "USA"
(1 row affected)
exec changeshipnamelnCruise "Cascoyne", "Gascoyne", 3014, 0
Update of Cascoyne to Gascoyne successful.
(return status = 0)
delete Ship
where Ship_Name like "Cascoyne" and Country like "Australia"
(1 row affected)
update Ship
set Ship_Name = "Charoit"
where Ship_Name like "Charoi" and Country like "USSR"
(1 row affected)
exec changeshipnamelnCruise "Fuji", "Fuji-Maru", -25027, 0
Update of Fuji to Fuji-Maru successful.
(return status = 0)
exec changeshipnamelnCruise "Fuji", "Fuji-Maru", -25026, 0
Update of Fuji to Fuji-Maru successful.
(return status = 0)
exec changeshipnamelnCruise "Fuji", "Fuji-Maru", -25002, 0
Update of Fuji to Fuji-Maru successful.
(return status = 0)
exec changeshipnamelnCruise "Fuji", "Fuji-Maru", -25001, 0
Update of Fuji to Fuji-Maru successful.
(return status = 0)
exec changeshipnamelnCruise "Fuji", "Fuji-Maru", -25000, 0
Update of Fuji to Fuji-Maru successful.
(return status = 0)
delete Ship
where Ship_Name like "Fuji"
and Country like "Japan"
(1 row affected)
update Ship
set Ship_Name = "Prof. Vieze"
where Ship_Name like "Prof. Viese"
(1 row affected)
update Ship
set Ship_Name = "Ranquel"
where Ship_Name like "Rambuel"
and Country like "Argentina"
(1 row affected)
exec changeShipNameInCruise "Ranonel", "Ranquel", 116, 0
Update of Ranonel to Ranquel successful.
(return status = 0)
delete Ship
where Ship_Name like "Ranonel"
and Country like "Argentina"
(0 row affected)
update Ship
set Ship_Name = "Yamana"
where Ship_Name like "Yamaha"
and Country like "Argentina"
(1 row affected)
** Insert triggers for table Station and Standard Data.
** Special solution for updating the validation flags.
**
** created: 1991-09-05
** creator: l.-P. kurdelski

--

drop trigger UpdateStationTgr

drop trigger UpdateStandard_DataTgr

create trigger UpdateStationTgr
on Station for update
as
begin
if update (Station_Id#)
begin
  print "Station_Id is not allowed to be updated!! Update aborted!"
  rollback transaction
return
end

if (select count(*) from deleted)=0
begin
  print "No rows found for update! Update aborted!"
  rollback transaction
return
end

if not update(Validation_Flag)
begin
  print "Validation_Flag has also to be updated for documentation purposes!"
  rollback transaction
return
end

insert into UpdateStation
select inserted.Station_Id#, inserted.Validation_Flag, getdate(), suser_id
from inserted

create trigger UpdateStandard_DataTgr
on Standard_Data for update
as
begin
if update (Standard_Data_Id#)
begin
  print "Standard_Data_Id# is not allowed to be updated!! Update aborted"
  rollback transaction
return
end

if (select count(*) from deleted)=0
begin
  print "No rows selected!! Update aborted"
  rollback transaction
return
end

if not exists (select * from Station s, inserted
  where s.Station_Id#=inserted.Station_Id#)
begin
  print "No Station with the new Station_Id# exists! Update aborted!"
  rollback transaction
return
end

if not update(Validation_Flag)
begin
  print "Validation_Flag has also to be updated for documentation purposes!"
rollback transaction
return
end
insert into UpdateData
select deleted.Standard_Data_Id#
    deleted.Station_Id#
    inserted.Station_Id#
    inserted.Validation_Flag,
    getdate(),
    suser_id()
from inserted, deleted
end
Zum Einfügen von Chlorophyll A-Werten in die Biomass Standard Data der SouthernOceanDB

Autoren: D. Hinz, J. Begemann
Datum: 22-Juli-1991

use SouthernOceanDB

exec insertChlorophyllA "4458",0,0.17
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4458",23,0.22
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4458",50,0.16
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4458",100,0.26
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4458",150,0.21
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4459",0,0.13
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4459",20,0.16
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4459",50,0.15
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4460",0,0.06
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4460",25,0.05
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4460",50,0.07
data update successful
transaction committed
(return status = 0)
exec insertChlorophyllA "4460",100,0.26
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4460",150,0.18
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4461",0,0.14
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4461",20,0.18
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4461",30,0.25
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4461",47,0.19
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4461",75,0.19
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4461",100,0.12
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4461",150,0.10
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4462",0,0.13
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4462",50,0.15
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4462",100,0.08
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4462",150,0.04
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4463",30,0.14
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4463",39,0.21
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4463",75,0.29
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4463",100,0.13
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4463",150,0.09
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4464",0,0.49
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4464",10,0.45
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4464",50,0.25
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4464",100,0.16
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4465",0,0.45
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4465",15,0.50
data insert successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4465",30,0.45
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4465",50,0.34
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4465",100,0.20
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4465",150,0.10
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4466",0,0.75
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4466",25,0.46
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4466",50,0.17
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4466",75,0.28
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4466",100,0.26
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4466",150,0.24
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4467",0,0.24
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4467",25,0.24
data insert successful!
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4467",50,0.22
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4467",75,0.25
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4467",100,0.21
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4467",125,0.06
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4467",150,0.12
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4468",0,0.50
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4468",25,0.25
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4468",50,0.29
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4468",75,0.36
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4468",100,0.44
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4468",125,0.14
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4468",150,0.38
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4469",0,0.35
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4469",30,0.29
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4469",50,0.15
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4469",100,0.35
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4470",0,0.02
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4470",30,0.01
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4470",75,0.02
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4470",100,0.02
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4470",150,0.02
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4471",0,0.01
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4471",25,0.02
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4471",50,0.02
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4471",75,0.01
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4471",100,0.07
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4471",125,0.00
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4471",150,0.00
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4472",0,0.02
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4472",10,0.01
data update successful
transaction committed
(return status = 0)
1> exec insertChlorophyllA "4472",25,0.01
  data update successful
  transaction committed
  (return status = 0)
1> exec insertChlorophyllA "4472",30,0.02
  data update successful
  transaction committed
  (return status = 0)
1> exec insertChlorophyllA "4472",50,0.10
  data update successful
  transaction committed
  (return status = 0)
1> exec insertChlorophyllA "4472",100,0.07
  data update successful
  transaction committed
  (return status = 0)
1> exec insertChlorophyllA "4473",0,0.02
  data update successful
  transaction committed
  (return status = 0)
1> exec insertChlorophyllA "4473",20,0.01
  data update successful
  transaction committed
  (return status = 0)
1> exec insertChlorophyllA "4473",24,0.03
  data update successful
  transaction committed
  (return status = 0)
1> exec insertChlorophyllA "4473",30,0.03
  data update successful
  transaction committed
  (return status = 0)
1> exec insertChlorophyllA "4473",50,0.05
  data update successful
  transaction committed
  (return status = 0)
1> exec insertChlorophyllA "4473",100,0.03
  data update successful
  transaction committed
  (return status = 0)
1> exec insertChlorophyllA "4473",150,0.00
  data update successful
  transaction committed
  (return status = 0)
1> exec insertChlorophyllA "4474",0,0.10
  data update successful
  transaction committed
  (return status = 0)
exec insertChlorophyllA "4474",28,0.13
data insert successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4474",50,0.17
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4474",100,0.32
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4474",150,0.31
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4475",0,0.20
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4475",23,0.91
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4475",50,0.13
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4475",100,0.32
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4475",150,0.06
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4476",0,0.16
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4476",10,0.29
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4476",20,0.26
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4476",30,0.31
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4476",50,0.21
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4476",100,0.29
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4476",150,0.26
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4477",0,1.27
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4477",25,0.42
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4477",50,0.20
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4477",100,0.20
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4477",150,0.63
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4478",0,0.56
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4478",20,0.31
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4478",25,0.39
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4478",50,0.19
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4478",100,0.15
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4480",0,0.45
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4480",21,0.54
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4480",50,0.29
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4480",100,0.14
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4480",150,0.32
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4481",0,1.00
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4481",25,0.96
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4481",50,0.66
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4481",100,0.18
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4481",150,0.08
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4482",0,1.09
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4482",25,0.97
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4482",50,0.77
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4482",75,0.47
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4482",100,0.31
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4483",0,1.38
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4483",10,1.07
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4483",20,1.69
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4483",50,0.70
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4483",100,0.33
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4483",150,0.21
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4484",0,0.88
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4484",30,1.02
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4484",50,0.33
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4484",100,0.19
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4484",150,0.14
data update successful
transaction committed
(return status = 0)
1> 2> exec insertChlorophyllA "4485",0,0.82
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4485",10,0.94
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4485",25,0.19
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4485",30,0.14
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4485",50,0.15
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4485",100,0.05
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4485",150,0.09
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4486",0,0.53
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4486",10,0.45
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4486",25,0.38
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4486",50,0.28
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4486",75,0.30
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4487",0,0.24
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4487",10,0.22
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4487",20,0.17
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4487",30,0.17
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4487",50,0.26
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4487",75,0.24
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4488",0,0.16
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4488",25,0.09
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4488",50,0.10
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4488",100,0.07
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4489",0,0.08
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4489",10,0.05
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4489",25,0.03
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4489",30,0.07
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4489",50,0.07
exec insertChlorophyllA "4489", 100, 0.10
transaction committed
(data update successful
return status = 0)

exec insertChlorophyllA "4489", 150, 0.07
transaction committed
(data update successful
return status = 0)

exec insertChlorophyllA "4490", 0, 0.78
transaction committed
(data update successful
return status = 0)

exec insertChlorophyllA "4490", 25, 0.70
transaction committed
(data update successful
return status = 0)

exec insertChlorophyllA "4490", 50, 0.37
transaction committed
(data update successful
return status = 0)

exec insertChlorophyllA "4490", 100, 0.28
transaction committed
(data update successful
return status = 0)

exec insertChlorophyllA "4490", 150, 0.15
transaction committed
(data update successful
return status = 0)

exec insertChlorophyllA "4491", 0, 0.14
transaction committed
(data update successful
return status = 0)

exec insertChlorophyllA "4491", 25, 0.17
transaction committed
(data update successful
return status = 0)

exec insertChlorophyllA "4491", 50, 0.13
transaction committed
(data update successful
return status = 0)

exec insertChlorophyllA "4491", 100, 0.14
transaction committed
(data update successful
return status = 0)

exec insertChlorophyllA "4491", 150, 0.09
transaction committed
(data update successful
return status = 0)

exec insertChlorophyllA "4492", 0, 0.22
transaction committed
(data update successful
return status = 0)
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4492", 20, 0.23
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4492", 30, 0.19
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4492", 50, 0.20
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4492", 100, 0.08
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4493", 0, 0.32
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4493", 25, 0.36
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4493", 50, 0.16
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4493", 100, 0.32
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4493", 150, 0.08
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4494", 0, 0.62
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4494", 10, 0.43
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4494", 25, 0.38
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4494",50,0.33
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4494",75,0.70
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4494",100,0.28
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4494",150,0.30
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4495",0,0.30
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4495",50,0.33
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4495",100,0.23
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4495",150,0.24
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4496",0,0.31
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4496",20,0.15
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4496",30,0.21
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4496",100,0.38
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4496",150,0.24
data update successful
transaction committed
(return status = 0)
exec insertChlorophyllA "4497", 25, 0.21
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4497", 50, 0.15
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4497", 100, 0.19
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4497", 150, 0.19
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4498", 0, 0.18
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4498", 25, 0.19
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4498", 50, 0.24
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4498", 100, 0.19
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4498", 150, 0.20
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4499", 0, 0.27
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4499", 24, 0.32
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4499", 50, 0.35
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4499", 100, 0.48
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4499",150,0.35
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4500",0,0.37
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4500",50,0.26
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4500",100,0.24
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4500",150,0.26
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4501",0,0.34
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4501",25,0.35
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4501",50,0.21
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4501",100,0.12
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4501",150,0.17
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4502",0,0.15
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4502",25,0.17
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4502",50,0.18
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4502",100,0.12
data update successful  
transaction committed  
(return status = 0)

1>  
2> exec insertChlorophyllA "4502",150,0.12  
data update successful  
transaction committed  
(return status = 0)

1>  
2> exec insertChlorophyllA "4739",0,0.23  
data update successful  
transaction committed  
(return status = 0)

1>  
2> exec insertChlorophyllA "4739",24,0.68  
data update successful  
transaction committed  
(return status = 0)

1>  
2> exec insertChlorophyllA "4739",50,0.59  
data update successful  
transaction committed  
(return status = 0)

1>  
2> exec insertChlorophyllA "4739",75,0.26  
data update successful  
transaction committed  
(return status = 0)

1>  
2> exec insertChlorophyllA "4739",100,0.80  
data update successful  
transaction committed  
(return status = 0)

1>  
2> exec insertChlorophyllA "4739",150,0.05  
data update successful  
transaction committed  
(return status = 0)

1>  
2> exec insertChlorophyllA "4740",0,0.30  
data update successful  
transaction committed  
(return status = 0)

1>  
2> exec insertChlorophyllA "4740",10,0.28  
data update successful  
transaction committed  
(return status = 0)

1>  
2> exec insertChlorophyllA "4740",20,0.26  
data update successful  
transaction committed  
(return status = 0)

1>  
2> exec insertChlorophyllA "4740",25,0.36  
data update successful  
transaction committed  
(return status = 0)

1>  
2> exec insertChlorophyllA "4740",50,0.32  
data update successful  
transaction committed  
(return status = 0)

1>  
2> exec insertChlorophyllA "4740",100,0.02  
data update successful
transaction committed
(return status = 0)

2> exec insertChlorophyllA "4740",150,0.01
data update successful
transaction committed
(return status = 0)

2> exec insertChlorophyllA "4741",0,0.42
data update successful
transaction committed
(return status = 0)

2> exec insertChlorophyllA "4741",22,0.42
data update successful
transaction committed
(return status = 0)

2> exec insertChlorophyllA "4741",50,0.38
data update successful
transaction committed
(return status = 0)

2> exec insertChlorophyllA "4741",75,0.25
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4741",100,0.26
data update successful
transaction committed
(return status = 0)

2> exec insertChlorophyllA "4741",150,0.17
data update successful
transaction committed
(return status = 0)

2> exec insertChlorophyllA "4742",0,0.55
data update successful
transaction committed
(return status = 0)

2> exec insertChlorophyllA "4742",10,0.55
data update successful
transaction committed
(return status = 0)

2> exec insertChlorophyllA "4742",25,0.38
data update successful
transaction committed
(return status = 0)

2> exec insertChlorophyllA "4742",50,0.25
data update successful
transaction committed
(return status = 0)

2> exec insertChlorophyllA "4742",75,0.13
data update successful
transaction committed
(return status = 0)

2> exec insertChlorophyllA "4742",100,0.23
data update successful
transaction committed
exec insertChlorophyllA "4742",150,0.09
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4743",0,0.34
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4743",25,0.34
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4743",50,0.25
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4743",75,0.25
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4743",100,0.25
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4743",150,0.11
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4744",0,0.44
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4744",25,0.38
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4744",50,0.25
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4744",75,0.17
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4744",100,0.17
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4744",150,0.17
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4745",0,0.63
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4745",10,0.59
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4745",30,0.55
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4745",50,0.46
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4745",75,0.55
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4745",100,0.38
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4745",150,0.21
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4746",0,0.30
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4746",25,0.32
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4746",50,0.30
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4746",75,0.21
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4746",100,0.10
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4746",150,0.06
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4747",0,0.46
  data update successful
  transaction committed
  (return status = 0)

1>
2> exec insertChlorophyllA "4747",10,0.55
  data update successful
  transaction committed
  (return status = 0)

1>
2> exec insertChlorophyllA "4747",25,0.55
  data update successful
  transaction committed
  (return status = 0)

1>
2> exec insertChlorophyllA "4747",50,0.55
  data update successful
  transaction committed
  (return status = 0)

1>
2> exec insertChlorophyllA "4747",75,0.23
  data update successful
  transaction committed
  (return status = 0)

1>
2> exec insertChlorophyllA "4747",100,0.13
  data update successful
  transaction committed
  (return status = 0)

1>
2> exec insertChlorophyllA "4747",150,0.03
  data update successful
  transaction committed
  (return status = 0)

1>
2> exec insertChlorophyllA "4748",0,0.19
  data update successful
  transaction committed
  (return status = 0)

1>
2> exec insertChlorophyllA "4748",25,0.23
  data update successful
  transaction committed
  (return status = 0)

1>
2> exec insertChlorophyllA "4748",50,0.34
  data update successful
  transaction committed
  (return status = 0)

1>
2> exec insertChlorophyllA "4748",75,0.25
  data update successful
  transaction committed
  (return status = 0)

1>
2> exec insertChlorophyllA "4748",100,0.23
  data update successful
  transaction committed
  (return status = 0)

1>
2> exec insertChlorophyllA "4748",150,0.19
  data update successful
  transaction committed
  (return status = 0)

1>
2> exec insertChlorophyllA "4749",0,0.10
data update successful
transaction committed
(return status = 0)

1> 2> exec insertChlorophyllA "4749",10,0.09
  data update successful
  transaction committed
  (return status = 0)
1> 2> exec insertChlorophyllA "4749",23,0.07
  data update successful
  transaction committed
  (return status = 0)
1> 2> exec insertChlorophyllA "4749",50,0.18
  data update successful
  transaction committed
  (return status = 0)
1> 2> exec insertChlorophyllA "4749",75,0.15
  data update successful
  transaction committed
  (return status = 0)
1> 2> exec insertChlorophyllA "4749",100,0.14
  data update successful
  transaction committed
  (return status = 0)
1> 2> exec insertChlorophyllA "4749",150,0.01
  data update successful
  transaction committed
  (return status = 0)
1> 2> exec insertChlorophyllA "4750",0,0.18
  data update successful
  transaction committed
  (return status = 0)
1> 2> exec insertChlorophyllA "4750",10,0.18
  data update successful
  transaction committed
  (return status = 0)
1> 2> exec insertChlorophyllA "4750",30,0.17
  data update successful
  transaction committed
  (return status = 0)
1> 2> exec insertChlorophyllA "4750",50,0.17
  data update successful
  transaction committed
  (return status = 0)
1> 2> exec insertChlorophyllA "4750",75,0.13
  data update successful
  transaction committed
  (return status = 0)
1> 2> exec insertChlorophyllA "4750",100,0.05
  data update successful
  transaction committed
  (return status = 0)
1> 2> exec insertChlorophyllA "4750",150,0.04
  data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4751",0,0.13
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4751",10,0.13
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4751",23,0.12
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4751",50,0.15
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4751",75,0.14
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4751",100,0.06
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4751",150,0.01
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4752",0,0.38
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4752",10,0.27
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4752",30,0.23
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4752",50,0.21
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4752",75,0.13
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4752",100,0.05
data update successful
transaction committed
exec insertChlorophyllA "4752",150,0.08
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4753",0,0.75
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4753",10,1.00
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4753",30,2.01
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4753",50,1.51
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4753",75,1.51
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4753",100,0.34
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4753",150,0.25
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4753",150,0.25
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4754",0,1.33
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4754",10,0.82
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4754",24,0.38
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4754",50,0.69
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4754",75,0.38
transaction committed
(data update successful
(return status = 0)
exec insertChlorophyllA "4754", 100, 0.17
transaction committed
(return status = 0)
exec insertChlorophyllA "4754", 150, 0.08
transaction committed
(return status = 0)
exec insertChlorophyllA "4755", 0, 0.63
transaction committed
(return status = 0)
exec insertChlorophyllA "4755", 10, 0.63
transaction committed
(return status = 0)
exec insertChlorophyllA "4755", 25, 0.50
transaction committed
(return status = 0)
exec insertChlorophyllA "4755", 50, 0.38
transaction committed
(return status = 0)
exec insertChlorophyllA "4755", 75, 0.25
transaction committed
(return status = 0)
exec insertChlorophyllA "4755", 100, 0.09
transaction committed
(return status = 0)
exec insertChlorophyllA "4755", 150, 0.07
transaction committed
(return status = 0)
exec insertChlorophyllA "4756", 0, 0.30
transaction committed
(return status = 0)
exec insertChlorophyllA "4756", 10, 0.30
transaction committed
(return status = 0)
exec insertChlorophyllA "4756", 25, 0.34
transaction committed
(return status = 0)
exec insertChlorophyllA "4756", 50, 0.44
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4757",0,6.28
  data update successful
  transaction committed
  (return status = 0)
1>
2> exec insertChlorophyllA "4757",10,5.02
  data update successful
  transaction committed
  (return status = 0)
1>
2> exec insertChlorophyllA "4757",24,5.65
  data update successful
  transaction committed
  (return status = 0)
1>
2> exec insertChlorophyllA "4757",50,5.02
  data update successful
  transaction committed
  (return status = 0)
1>
2> exec insertChlorophyllA "4757",75,2.54
  data update successful
  transaction committed
  (return status = 0)
1>
2> exec insertChlorophyllA "4758",0,1.69
  data update successful
  transaction committed
  (return status = 0)
1>
2> exec insertChlorophyllA "4758",10,1.48
  data update successful
  transaction committed
  (return status = 0)
1>
2> exec insertChlorophyllA "4758",30,0.94
  data update successful
  transaction committed
  (return status = 0)
1>
2> exec insertChlorophyllA "4758",50,0.75
  data update successful
  transaction committed
  (return status = 0)
1>
2> exec insertChlorophyllA "4758",75,1.90
  data update successful
  transaction committed
  (return status = 0)
1>
2> exec insertChlorophyllA "4758",100,0.12
  data update successful
  transaction committed
  (return status = 0)
1>
2> exec insertChlorophyllA "4758",150,0.09
  data update successful
  transaction committed
  (return status = 0)
1>
2> exec insertChlorophyllA "4759",0,0.82
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4759",10,0.38
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4759",30,0.63
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4759",50,0.63
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4759",75,0.75
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4759",100,0.23
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4760",0,1.26
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4760",10,1.90
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4760",30,2.33
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4760",50,0.56
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4760",75,0.36
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4760",100,0.34
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4760",150,0.17
data update successful
transaction committed
(return status = 0)

1>
2> exec insertChlorophyllA "4761",0,1.19
data update successful
transaction committed
(return status = 0)

1>
2> exec insertChlorophyllA "4761",10,0.88
data update successful
transaction committed
(return status = 0)

1>
2> exec insertChlorophyllA "4761",30,1.50
data update successful
transaction committed
(return status = 0)

1>
2> exec insertChlorophyllA "4761",50,1.51
data update successful
transaction committed
(return status = 0)

1>
2> exec insertChlorophyllA "4761",75,0.88
data update successful
transaction committed
(return status = 0)

1>
2> exec insertChlorophyllA "4761",100,0.56
data update successful
transaction committed
(return status = 0)

1>
2> exec insertChlorophyllA "4761",150,0.19
data update successful
transaction committed
(return status = 0)

1>
2> exec insertChlorophyllA "4762",0,37.68
data update successful
transaction committed
(return status = 0)

1>
2> exec insertChlorophyllA "4762",10,31.40
data update successful
transaction committed
(return status = 0)

1>
2> exec insertChlorophyllA "4762",25,31.40
data update successful
transaction committed
(return status = 0)

1>
2> exec insertChlorophyllA "4762",50,10.05
data update successful
transaction committed
(return status = 0)

1>
2> exec insertChlorophyllA "4762",75,1.63
data update successful
transaction committed
(return status = 0)

1>
2> exec insertChlorophyllA "4762",100,1.26
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4762",150,0.42
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4763",0,0.69
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4763",10,1.51
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4763",47,0.88
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4763",75,0.50
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4763",100,0.36
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4763",150,0.09
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4764",0,0.63
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4764",10,0.27
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4764",25,0.40
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4764",50,0.30
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4764",75,0.27
data update successful
transaction committed
(return status = 0)
exec insertChlorophyllA "4764",100,0.34
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4765",0,9.40
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4765",10,3.38
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4765",24,3.81
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4765",50,4.23
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4765",75,11.30
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4765",100,1.00
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4765",150,0.76
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4766",0,26.32
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4766",10,28.26
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4766",25,24.32
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4766",50,6.77
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4766",75,2.26
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4766",100,1.63  
data update successful  
transaction committed  
(return status = 0)  
1>  
2> exec insertChlorophyllA "4766",150,2.01  
data update successful  
transaction committed  
(return status = 0)  
1>  
2> exec insertChlorophyllA "4766",0,0.88  
data update successful  
transaction committed  
(return status = 0)  
1>  
2> exec insertChlorophyllA "4766",10,0.88  
data update successful  
transaction committed  
(return status = 0)  
1>  
2> exec insertChlorophyllA "4766",23,1.19  
data update successful  
transaction committed  
(return status = 0)  
1>  
2> exec insertChlorophyllA "4766",50,0.38  
data update successful  
transaction committed  
(return status = 0)  
1>  
2> exec insertChlorophyllA "4766",75,0.27  
data update successful  
transaction committed  
(return status = 0)  
1>  
2> exec insertChlorophyllA "4766",100,0.23  
data update successful  
transaction committed  
(return status = 0)  
1>  
2> exec insertChlorophyllA "4766",150,0.11  
data update successful  
transaction committed  
(return status = 0)  
1>  
2> exec insertChlorophyllA "4768",0,1.19  
data update successful  
transaction committed  
(return status = 0)  
1>  
2> exec insertChlorophyllA "4768",10,0.94  
data update successful  
transaction committed  
(return status = 0)  
1>  
2> exec insertChlorophyllA "4768",25,0.82  
data update successful  
transaction committed  
(return status = 0)  
1>  
2> exec insertChlorophyllA "4768",50,0.75  
data update successful  
transaction committed  
(return status = 0)  
1>  
2> exec insertChlorophyllA "4768",75,15.13
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4768",96,0.42
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4770",0,2.33
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4770",10,1.50
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4770",25,0.69
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4770",50,0.19
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4771",0,0.17
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4771",10,0.15
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4771",24,0.11
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4771",50,0.22
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4771",75,0.14
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4771",100,0.02
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4771",150,0.01
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4772",0,0.34
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4772",10,0.30
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4772",25,0.14
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4772",50,0.17
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4772",75,0.15
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4772",100,0.11
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4772",150,0.07
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4773",0,0.07
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4773",10,0.07
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4773",22,0.07
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4773",50,0.10
data update successful
transaction committed
1> exec insertChlorophyllA "4773",75,0.16
data update successful
(transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4773",100,0.10
data update successful
(transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4773",150,0.09
data update successful
(transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4774",0,0.02
data update successful
(transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4774",10,0.07
data update successful
(transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4774",24,0.07
data update successful
(transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4774",50,0.03
data update successful
(transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4774",75,0.07
data update successful
(transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4774",100,0.11
data update successful
(transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4774",150,0.09
data update successful
(transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4775",0,0.15
data update successful
(transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4775",10,0.17
data update successful
(transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4775",25,0.15
data update successful
(transaction committed
(return status = 0)
exec insertChlorophyllA "4775",48,0.21
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4775",100,0.09
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4775",150,0.01
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4776",0,0.10
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4776",10,0.12
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4776",23,0.07
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4776",50,0.13
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4776",75,0.15
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4776",100,0.13
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4776",150,0.02
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4777",0,0.10
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4777",10,0.11
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4777",25,0.08
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4777",50,0.06  
data update successful  
(transaction status = 0)
1>
2> exec insertChlorophyllA "4777",75,0.12  
data update successful  
(transaction status = 0)
1>
2> exec insertChlorophyllA "4777",100,0.06  
data update successful  
(transaction status = 0)
1>
2> exec insertChlorophyllA "4777",150,0.01  
data update successful  
(transaction status = 0)
1>
2> exec insertChlorophyllA "4778",0,0.23  
data update successful  
(transaction status = 0)
1>
2> exec insertChlorophyllA "4778",10,0.16  
data update successful  
(transaction status = 0)
1>
2> exec insertChlorophyllA "4778",24,0.15  
data update successful  
(transaction status = 0)
1>
2> exec insertChlorophyllA "4778",50,0.17  
data update successful  
(transaction status = 0)
1>
2> exec insertChlorophyllA "4778",75,0.27  
data update successful  
(transaction status = 0)
1>
2> exec insertChlorophyllA "4779",0,0.10  
data update successful  
(transaction status = 0)
1>
2> exec insertChlorophyllA "4779",10,0.08  
data update successful  
(transaction status = 0)
1>
2> exec insertChlorophyllA "4779",25,0.07
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4779",50,0.11
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4779",75,0.15
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4779",100,0.07
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4779",150,0.01
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4780",0,0.19
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4780",10,0.17
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4780",25,0.21
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4780",50,0.17
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4780",75,0.09
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4780",100,0.05
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4780",150,0.01
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4781",0,0.21
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4781",24,0.27
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4781",50,0.23
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4781",75,0.17
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4781",100,0.09
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4781",150,0.05
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4782",0,0.21
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4782",10,0.27
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4782",24,0.30
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4782",50,0.21
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4782",75,0.19
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4782",100,0.15
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4782",150,0.09
data update successful
transaction committed
(return status = 0)

1> exec insertChlorophyllA "4783",0,1.00
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4783",10,0.75
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4783",20,0.44
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4783",30,0.63
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4783",50,0.44
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4783",75,0.21
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4783",100,0.30
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4784",0,0.15
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4784",10,0.15
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4784",23,0.13
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4784",50,0.25
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4784",75,0.10
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4784",100,0.07
data update successful
transaction committed
(return status = 0)
exec insertChlorophyllA "4784",150,0.03
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4785",0,0.30
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4785",10,0.88
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4785",25,1.00
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4785",50,0.44
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4785",75,0.27
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4785",100,0.15
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4785",150,0.44
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4786",0,0.50
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4786",10,0.21
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4786",25,0.17
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4786",50,0.27
data update successful
transaction committed
(return status = 0)

exec insertChlorophyllA "4786",75,0.17
data update successful
transaction committed
(return status = 0)
2> exec insertChlorophyllA "4786",100,0.07
data update successful
transaction committed
(return status = 0)
1>
2> exec insertChlorophyllA "4786",150,0.03
data update successful
transaction committed
(return status = 0)
Abstract: Inserting chlorophyll values for BIOMASS from a source given by F. Brandini. Preparing the insertion procedure.

**Modification history**

**1991-08-05** lpk no insertion of duplicated stations

**1991-08-05** lpk no insertion of duplicated data

**update of standard_data**

use SouthernOceanDB

if exists (select * from sysobjects where name like "insertChlorophyllA")

drop proc insertChlorophyllA

create proc insertChlorophyllA

(@sn varchar(8), /*Station_Number*/
@d int, /*Depth*/
@c_a float /*Chlorophyll_A*/
)

as

begin

declare

@bsid int,
@bsdid int,
@bsdidd int

/*Die Daten werden dupliziert und, mit einer neuen Kennung versehen, als**
**neuer datensatz in die Tabelle eingefuegt. Dies geschieht, da die**
**ursprünglichen Stationsdaten KEINE Chlorophyll A-Werte enthielten,**
**aber die neuen Werte als zu denalten Station gehörig betrachtet werden.**
**zu einem späteren Zeitpunkt können diese Stationsdaten wieder gelöscht**
**werden, wenn die BIOMASS-Daten in der nun gueltigen Form gehalten**
**werden sollen.**

/*
/*

** Waehle die Stations Id aus, die zu der gegebenen Stations_Number**
** gehoert.**

begin transaction

select @bsid = Biomass_Station_Id#
from Biomass_Station_
where Station_Number like @sn
and Cruise_Name like "BES%"

/*
** Wenn zu dieser Station und zur gegebenen Tiefe ein Datensatz existiert,**
** dann wird dieser Datensatz aktualisiert.**

if exists

{select Biomass_Standard_Data_Id#
from Biomass_Standard_Data
where Depth = @d
begin
  select @bsdid = Biomass_Standard_Data_Id#
  from Biomass_Standard_Data
  where Depth = @d
  and Biomass_Station_Id# = @bsid
  and Biomass_Standard_Data_Id# < 8800000

  update Biomass_Standard_Data
  set Chlorophyll_A = @c_a
  where Biomass_Standard_Data_Id# = @bsdid

  print "data update successful"
end

else

begin
  /**
   * Ist dieser Datensatz nicht vorhanden, dann wird der Datensatz mit
   * der groessten Tiefe gesucht, die kleiner ist als die gegebenen Tiefe.
   * */
  select @bsdid = Biomass_Standard_Data_Id#
  from Biomass_Standard_Data
  where Depth = (select max(Depth) from Biomass_Standard_Data
    where Biomass_Station_Id# = @bsid
    and Depth < @d)
  and Biomass_Station_Id# = @bsid
  and Biomass_Standard_Data_Id# < 8800000

  /*/
  ** Da bereits ein Datensatz in dieser Form existieren kann,
  ** muss auch bei den neu eingefuehrten Datenschaten geprueft
  ** werden, ob die Tiefe hineinpasst.
  */

  /*/
  ** Wenn bereits ein Datensatz eingefuegt wurde,
  ** dann nehme dessen Nummer und addiere 1 hinzu,
  ** sonst nehme die urspruengliche Id und addiere
  ** 800000 hinzu.
  ** Das verfahren funktioniert, da die Tiefen in
  ** aufsteigender Folge angeboten werden.
  */

  if exists
    (select Biomass_Standard_Data_Id#
      from Biomass_Standard_Data
      where Depth = (select max(Depth) from Biomass_Standard_Data
        where Biomass_Station_Id# = @bsid
        and Depth < @d)
      and Biomass_Station_Id# = @bsid
      and Biomass_Standard_Data_Id# > 8800000)
  begin
    select @bsdidd = Biomass_Standard_Data_Id#
    from Biomass_Standard_Data
    where Depth = (select max(Depth) from Biomass_Standard_Data
      where Biomass_Station_Id# = @bsid
      and Depth < @d)
    and Biomass_Station_Id# = @bsid
    and Biomass_Standard_Data_Id# > 8800000

    begin
      select @bsdid = @bsdidd + 1
      end
    else
      begin
        select @bsdid = @bsdid + 800000
        end

  /*
  ** Dann fuege den neuen Datensatz hinzu.
*/
insert Biomass_Standard_Data
(Biomass_Standard_Data_Id#,
  Biomass_Station_Id#,
  Depth,
  Chlorophyll_A,
  PrDepth,
  DepthType,
  ObsType)
values(@bsdid,@bsid,@d,@c_a,@d,"-","-")

print "data insert successful"
end

if @@error = 0

begin
  print "transaction committed"
  commit transaction
endif

else

begin
  print "transaction rolled back"
  rollback transaction
endif
/** SYBASE script
** Creation: 1990-12-12
** Creator: L.-P. Kurdelski
** FILE: COPY_AARI_TO_CRUISE.SYBASE
***/
use SouthernOceanDB
go
if exists (select * from sysobjects where name like "Copy_Aari_To_Cruise")
drop proc Copy_Aari_To_Cruise

create procedure Copy_Aari_To_Cruise
as
/** Copy the Cruise Number from Aari Cruise_Bck
*** and the Ship_Id# from Ship to taEle
*** Cruise
*** L.-P. Kurdelski 1990-12-12
*/
insert into Cruise
select Cruise Number, Ship_Id#
from Aari Cruise_Bck a, Ship s
where a.Ship = s.Ship_Name
    and a.Country = s.Country

go
/*
** Zum Einfügen von Chlorophyll A-Werten in die Biomass Standard Data der SouthernOceanDB
**
** Autoren: D. Hinz, J. Begemann
** Datum: 22-Juli-1991
*/

use SouthernOceanDB
exec insertChlorophyllA "4465",15,0.50
data insert successful!
transaction committed
(return status = 0)

exec insertChlorophyllA "4467",25,0.24
data insert successful!
transaction committed
(return status = 0)

exec insertChlorophyllA "4474",28,0.13
data insert successful!
transaction committed
(return status = 0)
begin transaction
exec updateCruise 9890, "unknown", "unknown"
exec updateCruise 9768, "Ushakov", "USSR"
exec updateCruise 11669, "unknown", "unknown"
exec updateCruise 10589, "Prof. Zubov", "USSR"
exec updateCruise -22011, "Prof. Zubov", "USSR"
exec updateCruise 890, "unknown", "unknown"
exec updateCruise 619, "Glacier", "USA"
exec updateCruise 891, "unknown", "unknown"
exec updateCruise -667, "Lesnoi", "USSR"
exec updateCruise 4231, "unknown", "unknown"
exec updateCruise 10097, "unknown", "unknown"
exec updateCruise 608, "unknown", "unknown"
if @@error != 0
begin
print "transaction rollback"
rollback transaction
end
else
begin
print "transaction commit"
commit transaction
end
transaction commit
/*
** SYASE Script
**
** CREATED: 1991-10-08
**
** CREATOR: L.-P. Kurdelski
**
** FILE: ShipCruiseCorrection081091.script
**
*/

use SouthernOceanDB

begin transaction

exec updateCruise -22013, "Prof. Vieze", "USSR"
(return status = 0)

if @@error != 0
    print "transaction rollback"
    rollback transaction
else
    print "transaction commit"
    commit transaction
end

transaction commit
use SouthernOceanDB
exec insert_into_cruise 55071, "unknown", "unknown" (return status = 0)
begin transaction
(update Cruise
set Ship Id# = 1000000
where Ship_Id# in (select Ship_Id#
    from Ship
    where Ship_Name like "unknown"
    and Country like "unknown"
    and Ship_Id# < 1000000)
(27 rows affected)
if @@error != 0
begin
    rollback transaction
    print "No update"
end
else
begin
    commit transaction
    print "Unknown ships updated"
end
Unknown ships updated
Dieses Script geht davon aus, dass die Datenbank SouthernOceanDB bereits existiert. 

begin
print 'SouthernOceanDB does not exist !!!'
return
end

use SouthernOceanDB

/* Erzeugen der Tabellen */
if exists (select * from sysobjects where name= "Argentine_Station")
drop table Argentine_Station
create table Argentine_Station
(Argentine_Station_Id# int, /* internal identification */
Cruise_Number int, /* Argentine nc */
Station_Number int, /* Argentine ns */
Longitude float, /* Argentine longitude */
Latitude float, /* Argentine latitude */
Date_Time datetime
/* Argentine year month day hour */
Bottom_Depth int NULL, /* Argentine nde */
Max_Obse_Depth int NULL, /* Argentine mode */
Number_Obse int, /* Argentine hz */
Marsden_Square# int NULL /* Argentine msq */)

if exists (select * from sysobjects where name= "Argentine_Standard_Data")
drop table Argentine_Standard_Data
create table Argentine_Standard_Data
(Argentine_Standard_Data_Id# int,
Argentine_Station_Id# int,
Depth int, /* Argentine zst */
Temperature float NULL, /* Argentine tst */
Salinity float NULL, /* Argentine sst */
Oxygen float NULL /* Argentine oxst */)
New foreign key added.
(return status = 0)

1> /* Erzeugen der Views */
2> /* Erzeugen der Prozeduren */
3> /* Eintragen der Nutzer */
4> grant select on Argentine_Station to PHYSIK1
5> grant select on Argentine_Standard_Data to PHYSIK1
/*
** SYBASE Script
** Created 1991-10-29
** Creator: L.-P. Kurdelski
** FILE: ARGENTINELOAD_COPY.SCRIPT
*/
use SouthernOceanDB
if @@error != 0
begin
    print 'Fehler beim Oeffnen der Datenbank.'
    return
end
if exists (select *
        from sysobjects
        where name like "Argentine_copy_all")
drop procedure Argentine_copy_all
create procedure Argentine_copy_all
    as
/* Fill Station and Standard_Data */
begin
    insert Station (Station_Id#, Cruise_Number, Station_Number,
        Date_Time, Longitude, Latitude, Bottom_Depth, Max_Obse_Depth,
        Number_Obse, Marsden_Square#)
        select Argentine_Station_Id#, Cruise_Number, Station_Number,
            Date_Time, Longitude, Latitude, Bottom_Depth, Max_Obse_Depth,
            Number_Obse, Marsden_Square#
        from Argentine_Station
    insert Standard_Data (Standard_Data_Id#, Station_Id#,
        Depth, Temperature, Salinity, Oxygen)
        select Argentine_Standard_Data_Id#, Argentine_Station_Id#,
            Depth, Temperature, Salinity, Oxygen
        from Argentine_Standard_Data
end
execute Argentine_copy_all
(return status = 0)
Dieses Script geht davon aus, dass die Datenbank SouthernOceanDB bereits existiert.

Use master

/* Erzeugen der Datenbank */
if not exists (select * from master.dbo.sysdatabases
where name = "SouthernOceanDB")
begin
print 'SouthernOceanDB does not exist !!!'
return
end

Use SouthernOceanDB

/* Erzeugen der Tabellen */
if exists (select * from sysobjects where name="AWI_Station")
drop table AWI_Station
create table AWI_Station
(AWI_Station_Id# int, /* internal identification */
Cruise_Number int, /* AWI ns */
Station_Number int, /* AWI ns */
Longitude float, /* AWI longitude */
Latitude float, /* AWI latitude */
Date_Time datetime NULL,
Bottom_Depth int NULL,/* AWI year month day hour */
Max_Obse_Depth int NULL,/* AWI mode */
Number_Obse int, /* AWI nz */
Marsden_Square# int NULL /* AWI mode */)
execute sp_primarykey AWI_Station, AWI_Station_Id#
execute sp_primarykey AWI_Standard_Data, AWI_Standard_Data_Id#
New primary key added.
New primary key added.

/* Definition der Sekundaerschluessel in den Tabellen */
execute sp_foreignkey AWI_Standard_Data, AWI_Station, AWI_Station_Id#

execute sp_primarykey AWI_Standard_Data, AWI_Standard_Data_Id#
New primary key added.
New primary key added.
New primary key added.
New foreign key added.
(return status = 0)
1> /* Erzeugen der Views */
2> /* Erzeugen der Prozeduren */
3> /* Eintragen der Nutzer */
4> grant select on AWI_Station to PHYSIK1
5> grant select on AWI_Standard_Data to PHYSIK1
SYBASE script
CREATOR: L.-P. Kurdelski
CREATIONDATE: 1991-11-06
FILE: REBUILD_ARGENTINE.SCRIPT

Deleting Argentine data from Station and Standard_Data and preparing the tables Argentine_Station and Argentine_Standard_Data for reload.

use SouthernOceanDB

declare @maxs int, @mins int

/*
select @maxs = max(Argentine_Station_Id#),
@mins = min(Argentine_Station_Id#)
from Argentine_Station
*/

select @maxs = 3005000
select @mins = 3000000

delete from Station where Station_Id# between @mins and @maxs
print "Station deleted (Data delete via trigger)"
(1 row affected)
(1 row affected)
Records in tables Station and Standard_Data deleted!
(4744 rows affected)

Station deleted (Data delete via trigger)

truncatem table Argentine_Station

truncatem table Argentine_Standard_Data

print "DONE"
SYBASE script
Created: 1991-08-29
Creator: L.-P. Kurdelski

FILE: changeShipNameAndCountryInCruise.script

Change the ship ids in table Cruise due to the change of name from old ship to new ship.

An old ship name, a new ship name and two cruise number are used as parameters. If the second number is 0 then only the first number is used (just as you expect). Otherwise both numbers are used in a junction.

If both cruise numbers are 0 it is ok. if the first cruise number is 0 and the second is not 0 this ** is an error.

Summary

** Cru1 Cru2 Comment
0 0 change all ship_id for the old ship
!=0 0 ... only for cruise Cru1
!=0 !=0 ... only for cruise Cru1 and Cru2

/ *
use SouthernOceanDB

if exists (select * from sysobjects

where name like "changeShipNameAndCountryInCruise")

drop proc changeShipNameAndCountryInCruise

create proc changeShipNameAndCountryInCruise

@oldShip varchar(40),
@oldCountry varchar(40),
@newShip varchar(40),
@newCountry varchar(40),
@cru1 int,
@cru2 int

as

/*

Change the ship ids in table Cruise due to the change of name from old ship to new ship.

An old ship name, an old country name, a new ship name, a new country name, and two cruise number are used as parameters. If the second number is 0 then only the first number is used (just as you expect). Otherwise both numbers are used in a junction.

If both cruise numbers are 0 it is ok. if the first cruise number is 0 and the second is not 0 this ** is an error.

Summary

** Cru1 Cru2 Comment
0 0 change all ship_id for the old ship
!=0 0 ... only for cruise Cru1
!=0 !=0 ... only for cruise Cru1 and Cru2
0 !=0 illegal

declare @sid int,
begin transaction
select @sidalt = Ship_Id#
from Ship
where Ship_Name like @oldShip
and Country like @oldCountry
select @sid = Ship_Id#
from Ship
where Ship_Name like @newShip
and Country like @newCountry
if @sidalt is null
begin
select @msg = "Old ship '"+@oldShip+" '+@oldCountry+' not found"
print @msg
return
end
if @sid is null
begin
select @sid = max(Ship_Id#) + 1
from Ship
where Ship_Id# < 100000
insert Ship values (@sid,@newShip,@newCountry,NULL)
end
if @cru2 = 0
begin
if @crul = 0
begin
update Cruise
set Ship_Id# = @sid
where Ship_Id# = @sidalt
end
else
begin
update Cruise
set Ship_Id# = @sid
where Ship_Id# = @sidalt and
Cruise_Number# = @crul
end
else
begin
update Cruise
set Ship_Id# = @sid
where Ship_Id# = @sidalt and
(Cruise_Number# = @crul or Cruise_Number# = @cru2)
end
end
if @@error != 0
begin
rollback transaction
return
end
begin
rollback transaction
select @msg = "ERROR occurred during update of " + @oldShip
end
else
begin
commit transaction
select @msg = "Update of " + @oldShip + " to " + @newShip + " successful."
end
print @msg
end
Dieses Script geht davon aus, dass die Datenbank SouthernOceanDB bereits existiert.

```
use master

if not exists (select * from master.dbo.sysdatabases
    where name = "SouthernOceanDB")
begin
    print 'SouthernOceanDB does not exist !!!'
    return
end

use SouthernOceanDB

/* Erzeugen der Tabellen */
if exists (select * from sysobjects where name= "Schlitzer_Station")
drop table Schlitzer_Station
create table Schlitzer_Station
(
    Schlitzer_Station_Id# int, /* internal identification */
    Cruise_Number int, /* Schlitzer nc */
    Station_Number int, /* Schlitzer ns */
    Longitude float, /* Schlitzer longitude */
    Latitude float, /* Schlitzer latitude */
    Date_Time datetime NULL, /* Schlitzer date */
    Bottom_Depth int NULL, /* Schlitzer nde */
    Max_Obse_Depth int NULL, /* Schlitzer mede */
    Number_Obse int, /* Schlitzer hz */
    Marsden_Square# int NULL /* Schlitzer msq */
)
if exists (select * from sysobjects where name="Schlitzer_Standard_Data")
drop table Schlitzer_Standard_Data
create table Schlitzer_Standard_Data
(
    Schlitzer_Standard_Data_Id# int, /* internal identification */
    Schlitzer_Station_Id# int, /* Schlitzer sst */
    Depth int, /* Schlitzer dth */
    Temperature float NULL, /* Schlitzer temp */
    Salinity float NULL, /* Schlitzer sal */
    Oxygen float NULL, /* Schlitzer oxy */
    Phosphate float NULL,
    Silicate float NULL,
    Nitrate float NULL
)
execute sp_primarykey Schlitzer_Station, Schlitzer_Station_Id#
execute sp_primarykey Schlitzer_Standard_Data, Schlitzer_Standard_Data_Id#
```

New primary key added.
(return status = 0)
New primary key added.
(return status = 0)

/* Definition der Sekundaerschluessel in den Tabellen */
execute sp_foreignkey Schlitzer_Standard_Data, Schlitzer_Station,
    Schlitzer_Station_Id#
New foreign key added.
(return status = 0)
1>
2> /* Erzeugen der Views */
3> /* Erzeugen der Prozeduren */
4>
5> /* Eintragen der Nutzer */
6> grant select on Schlitzer_Station to PHYSIK1
1>
2> grant select on Schlitzer_Standard_Data to PHYSIK1
/*
** SYBASE Script
** Created: 1991-10-09
** Creator: L.P. Kurdelski
** FILE: SchlitzerLOAD_UPD.SCRIPT
** Script um missing Values auf NULL zu setzen
** Schlitzer_Station
** Schlitzer_Standard_Data
*/

use SouthernOceanDB

dump tran SouthernOceanDB with truncate_only
update Schlitzer_Station
set Date_Time = NULL
where Date_Time="Jan 01 1900 00:00:00.0"
(0 rows affected)
dump tran SouthernOceanDB with truncate_only
update Schlitzer_Station
set Bottom_Depth = NULL
where Bottom_Depth = -9999
(0 rows affected)
dump tran SouthernOceanDB with truncate_only
update Schlitzer_Standard_Data
set Temperature = NULL
where Temperature < -10.0
(199 rows affected)
dump tran SouthernOceanDB with truncate_only
update Schlitzer_Standard_Data
set Salinity = NULL
where Salinity < 0
(213 rows affected)
dump tran SouthernOceanDB with truncate_only
update Schlitzer_Standard_Data
set Oxygen = NULL
where Oxygen <= -9.0
(506 rows affected)
dump tran SouthernOceanDB with truncate_only
update Schlitzer_Standard_Data
set Phosphate = NULL
where Phosphate < 0
(384 rows affected)
dump tran SouthernOceanDB with truncate_only
update Schlitzer_Standard_Data
set Silicate = NULL
where Silicate < 0
(277 rows affected)
dump tran SouthernOceanDB with truncate_only
update Schlitzer_Standard_Data
set Nitrate= NULL
where Nitrate< 0
(1080 rows affected)
dump tran SouthernOceanDB with truncate_only
update Schlitzer_Station
set Marsden_Square# = NULL
where Marsden_Square# <= 0
(209 rows affected)
dump tran SouthernOceanDB with truncate_only
1> /*
2> ** SYBASE Script
3> ** Created 1991-10-29
4> ** Creator: L.-P. Kurdelski
5> **
6> **FILE: SchlitzerLOAD_COPY.SCRipt
7> */
8> use SouthernOceanDB
9> if @@error != 0
10> begin
11> print 'Fehler beim Oeffnen der Datenbank.'
12> return
13> end
14> if exists (select * from sysobjects where name like "Schlitzer_copy_all")
15> drop procedure Schlitzer_copy_all
16> create procedure Schlitzer_copy_all
17> as
18> /* Fill Station and Standard_Data with Schlitzer Data*/
19> /* L.-P. Kurdelski 1991-10-28 */
20> begin
21> insert Station (Station_Id#, Cruise_Number, Station_Number,
22> Date_Time, Longitude, Latitude, Bottom_Depth, Max_Obse_Depth,
23> Number_Obse, Marsden_Square#)
24> select Schlitzer_Station_Id#, Cruise_Number, Station_Number,
25> Date_Time, Longitude, Latitude, Bottom_Depth, Max_Obse_Depth,
26> Number_Obse, Marsden_Square#
27> from Schlitzer_Station
28> insert Standard_Data (Standard_Data_Id#, Station_Id#, Depth, Temperature, Salinity, Oxygen)
29> select Schlitzer_Standard_Data_Id#, Schlitzer_Station_Id#, Depth, Temperature, Salinity, Oxygen
30> from Schlitzer_Standard_Data
31> end
32> execute Schlitzer_copy_all
33> (return status = 0)
/*
** SYBASE script
** CREATOR: L.-P. Kurdelski
** CREATIONDATE: 1991-12-04
** FILE: SHIP_UPD_04121991.SCRIPT
** Updating ships
** use SouthernOceanDB
** Change the names of two ships
*/
if exists ( select * from Ship where Ship_Name like 'Dege' )
begin
update Ship
set Ship_Name = 'Degei'
where Ship_Name like 'Dege'
end
else
print "Ship 'Dege' does not exist."
ship 'Dege' does not exist.

if exists ( select * from Ship where Ship_Name like 'Diagnita' )
begin
update Ship
set Ship_Name = 'Diaguita'
where Ship_Name like 'Diagnita'
end
else
print "Ship 'Diagnita' does not exist."
ship 'Diagnita' does not exist.

** Change relation between Ship and Cruise including the country. */
exec changeShipNameAndCountryInCru
"unknown", "unknown", "Th. Thompson", "USA", 12220, 0
Update of unknown to Th. Thompson successful.
(return status = 0)
exec changeShipNameAndCountryInCru
"unknown", "unknown", "Degei", "Australia", 2157, 0
Update of unknown to Degei successful.
(return status = 0)
exec changeShipNameAndCountryInCru
"unknown", "unknown", "Saga", "Australia", 1078, 0
Update of unknown to Saga successful.
(return status = 0)
exec changeShipNameAndCountryInCru
"Boa", "Argentina", "Cap. Canepa", "Argentina", 5052, 0
Update of Boa to Cap. Canepa successful.
(return status = 0)
exec changeShipNameAndCountryInCru
(return status = 0)
exec changeShipNameAndCountryInCru
"Boa", "Argentina", "Cap. Canepa", "Argentina", 5054, 0
Update of Boa to Cap. Canepa successful.
(return status = 0)
1> exec changeShipNameAndCountryInCru
2> "unknown", "unknown", "Cap. Canepa", "Argentina", 6525, 0
Update of unknown to Cap. Canepa successful.
(return status = 0)
1>
2> /*
3> ** Inserting the ships into a temporary table "tempships"
4> */
5> */
6> ** Clearing the table.
7> */
8> truncate table tempships
1> /*
2> */
3>/*
4> insert tempships (Id#, cn, ship, country)
5> values (1, 58001, "Gen. San-Martin", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (2, 58002, "Bahia Blanca", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (3, 58003, "Madryn", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (4, 58803, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (5, 58804, "Yamana", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (6, 58805, "Republica", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (7, 58805, "Ranquel", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (8, 58807, "Diaguita", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (9, 58004, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (10, 58005, "Gen. San-Martin", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (11, 58006, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (12, 58008, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (13, 58009, "Gen. San-Martin", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (14, 58010, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (15, 58011, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (16, 58012, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (17, 58013, "Gen. San-Martin", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (18, 58014, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (19, 58015, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (20, 58016, "Gen. San-Martin", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (21, 58017, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (22, 58018, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (23, 58019, "Gen. San-Martin", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (24, 58020, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (25, 58021, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (26, 58022, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (27, 58024, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (28, 58025, "Gen. Zapiola", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (29, 58027, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (30, 58028, "Comodoro Augusto Lasserre", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (31, 58029, "Gen. San-Martin", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (32, 58030, "Gen. Zapiola", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (33, 58031, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (34, 58032, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (35, 58033, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (36, 58034, "Gen. San-Martin", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (37, 58035, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (38, 58036, "Cap. Canepa", "Argentina")
(1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (39, 58037, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (40, 58038, "Gen. San-Martin", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (41, 58039, "Gen. Zapiola", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (42, 58040, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (43, 58041, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (44, 58042, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (45, 58044, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (46, 58045, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (47, 58046, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (48, 58047, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (49, 58048, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (50, 58049, "Gen. San-Martin", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (51, 58050, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (52, 58051, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (53, 58052, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (54, 58053, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (55, 58054, "Gen. San-Martin", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (56, 58055, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (57, 58056, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (58, 58057, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (59, 58058, "Cap. Canepa", "Argentina")
   3. (1 row affected)
1. insert tempships (Id#, en, ship, country)
   2. values (60, 58059, "Goyena", "Argentina")
   3. (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (61, 58060, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (62, 58061, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (63, 58063, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (64, 58064, "Goyena", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (65, 58065, "Gen. San-Martin", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (66, 58066, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (67, 58067, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (68, 58068, "Gen. San-Martin", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (69, 58069, "Goyena", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (70, 58070, "Cap. Canepa", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (71, 58071, "Gen. San-Martin", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (72, 58073, "Goyena", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (73, 58074, "Goyena", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (74, 58075, "Goyena", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (75, 58076, "Goyena", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (76, 58078, "Islas Orcadas", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (77, 58079, "Islas Orcadas", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (78, 58080, "Islas Orcadas", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (79, 58081, "Islas Orcadas", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (80, 58084, "Islas Orcadas", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (81, 58085, "Islas Orcadas", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (82, 58089, "Islas Orcadas", "Argentina")
(1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (83, 58100, "Gen. San-Martin", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (84, 58101, "San Juan", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (85, 58808, "San Luis", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (86, 58109, "Alferez Mackinlay", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (87, 58102, "Gen. Zapiola", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (88, 58103, "Goyena", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (89, 58104, "Islas Orcadas", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (90, 58105, "Islas Orcadas", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (91, 58106, "Islas Orcadas", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (92, 58107, "Cap. Canepa", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (93, 58108, "San Juan", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (94, 58109, "Islas Orcadas", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (95, 58300, "Walther Herwig", "FRG")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (96, 58301, "Walther Herwig", "FRG")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (97, 58302, "Walther Herwig", "FRG")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (98, 58306, "unknown", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (99, 58307, "unknown", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (100, 58308, "unknown", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (101, 58309, "unknown", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (102, 58310, "unknown", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (103, 58311, "unknown", "Argentina")
3> (1 row affected)
1> insert tempships (Id#, en, ship, country)
2> values (104, 58312, "unknown", "Argentina")
3> (1 row affected)
<table>
<thead>
<tr>
<th>Id#</th>
<th>En</th>
<th>Ship</th>
<th>Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>105</td>
<td>58313</td>
<td>unknown</td>
<td>Argentina</td>
</tr>
<tr>
<td>106</td>
<td>58314</td>
<td>Gen. San-Martin</td>
<td>Argentina</td>
</tr>
<tr>
<td>107</td>
<td>58315</td>
<td>Gen. San-Martin</td>
<td>Argentina</td>
</tr>
<tr>
<td>108</td>
<td>58316</td>
<td>Gen. San-Martin</td>
<td>Argentina</td>
</tr>
<tr>
<td>109</td>
<td>58317</td>
<td>Gen. San-Martin</td>
<td>Argentina</td>
</tr>
<tr>
<td>110</td>
<td>58318</td>
<td>Doctor Holmberg</td>
<td>Argentina</td>
</tr>
<tr>
<td>111</td>
<td>58319</td>
<td>Doctor Holmberg</td>
<td>Argentina</td>
</tr>
<tr>
<td>112</td>
<td>58320</td>
<td>Doctor Holmberg</td>
<td>Argentina</td>
</tr>
<tr>
<td>113</td>
<td>58321</td>
<td>Doctor Holmberg</td>
<td>Argentina</td>
</tr>
<tr>
<td>114</td>
<td>58322</td>
<td>Doctor Holmberg</td>
<td>Argentina</td>
</tr>
<tr>
<td>115</td>
<td>58323</td>
<td>Doctor Holmberg</td>
<td>Argentina</td>
</tr>
<tr>
<td>116</td>
<td>58324</td>
<td>Doctor Holmberg</td>
<td>Argentina</td>
</tr>
<tr>
<td>117</td>
<td>58325</td>
<td>Doctor Holmberg</td>
<td>Argentina</td>
</tr>
<tr>
<td>118</td>
<td>58326</td>
<td>Doctor Holmberg</td>
<td>Argentina</td>
</tr>
<tr>
<td>119</td>
<td>58327</td>
<td>Doctor Holmberg</td>
<td>Argentina</td>
</tr>
<tr>
<td>120</td>
<td>58328</td>
<td>Doctor Holmberg</td>
<td>Argentina</td>
</tr>
<tr>
<td>121</td>
<td>58329</td>
<td>Doctor Holmberg</td>
<td>Argentina</td>
</tr>
<tr>
<td>122</td>
<td>58330</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>123</td>
<td>58331</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>124</td>
<td>58332</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>125</td>
<td>58333</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>126</td>
<td>58334</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>Id#</td>
<td>en</td>
<td>ship</td>
<td>country</td>
</tr>
<tr>
<td>-----</td>
<td>---------</td>
<td>----------</td>
<td>------------</td>
</tr>
<tr>
<td>127</td>
<td>58335</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>128</td>
<td>58336</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>129</td>
<td>58337</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>130</td>
<td>58338</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>131</td>
<td>58339</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>132</td>
<td>58340</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>133</td>
<td>58341</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>134</td>
<td>58342</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>135</td>
<td>58343</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>136</td>
<td>58344</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>137</td>
<td>58345</td>
<td>Cap. Canepa</td>
<td>Argentina</td>
</tr>
<tr>
<td>138</td>
<td>58346</td>
<td>unknown</td>
<td>Argentina</td>
</tr>
<tr>
<td>139</td>
<td>58347</td>
<td>unknown</td>
<td>Argentina</td>
</tr>
<tr>
<td>140</td>
<td>58348</td>
<td>Austral</td>
<td>Argentina</td>
</tr>
<tr>
<td>141</td>
<td>58349</td>
<td>Austral</td>
<td>Argentina</td>
</tr>
<tr>
<td>142</td>
<td>58350</td>
<td>Austral</td>
<td>Argentina</td>
</tr>
<tr>
<td>143</td>
<td>58351</td>
<td>Austral</td>
<td>Argentina</td>
</tr>
<tr>
<td>144</td>
<td>58352</td>
<td>Austral</td>
<td>Argentina</td>
</tr>
<tr>
<td>145</td>
<td>58353</td>
<td>Puerto Deseado</td>
<td>Argentina</td>
</tr>
<tr>
<td>146</td>
<td>58354</td>
<td>Austral</td>
<td>Argentina</td>
</tr>
<tr>
<td>147</td>
<td>58355</td>
<td>Austral</td>
<td>Argentina</td>
</tr>
<tr>
<td>148</td>
<td>58356</td>
<td>Austral</td>
<td>Argentina</td>
</tr>
</tbody>
</table>
```sql
1> insert tempships (Id#, cn, ship, country)
2> values (149, 58603, "Austral", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (150, 58604, "unknown", "Argentina")
(1 row affected)
1> insert tempships (Id#, cn, ship, country)
2> values (151, 58605, "unknown", "Argentina")
(1 row affected)
1>
2> /*
3> ** Inserting ships into the Ship and Cruise tables
4> */
5> begin transaction
1>
2> exec insert_new_ships
3> (return status = 0)
4>
5> if @@error != 0
6> begin
7> rollback transaction
8> end
9> else
10> begin
11> commit transaction
12> end
13>
14> print "DONE"
```

DONE
use SouthernOceanDB

exec changeShipNameAndCountryInCru
 unused, unused, "Fuji-Maru", "Japan", -25000, 0
Update of unused to Fuji-Maru successful. (return status = 0)

exec changeShipNameAndCountryInCru
 unused, unused, "Fuji-Maru", "Japan", -25001, 0
Update of unused to Fuji-Maru successful. (return status = 0)

exec changeShipNameAndCountryInCru
 unused, unused, "Fuji-Maru", "Japan", -25002, 0
Update of unused to Fuji-Maru successful. (return status = 0)

print "DONE"
DONE
1> /*
2> ** SYBASE script
3> **
4> ** CREATOR: L.-P. Kurdelski
5> **
6> ** CREATIONDATE: 1991-12-05
7> **
8> ** FILE: SHIP_UPD_05121991_2.SCRIPT
9> **
10> ** Updating ships
11> */
12> use SouthernOceanDB
13> /*
14> ** Updating the ship cruise relation
15> */
16> exec changeShipNameAndCountryInCru
17> "unknown", "unknown", "Chatyr-Dag", "USSR", 10350, 0
18> Update of unknown to Chatyr-Dag successful.
19> (return status = 0)
20>
21> exec changeShipNameAndCountryInCru
22> "unknown", "unknown", "Gizhiga", "USSR", 10097, 0
23> Update of unknown to Gizhiga successful.
24> (return status = 0)
25>
26> exec changeShipNameAndCountryInCru
27> "unknown", "unknown", "Prof. Zubov", "USSR", -22012, 0
28> Update of unknown to Prof. Zubov successful.
29> (return status = 0)
30>
31> exec changeShipNameAndCountryInCru
32> "unknown", "unknown", "Evrica", "USSR", 4231, 0
33> Update of unknown to Evrica successful.
34> (return status = 0)
35>
36> print "DONE"
37> DONE
1> /*
2> ** Sybase script
3> **
4> ** Written 1992-02-03 by L.-P. Kurdelski
5> **
6> ** FILE: updateshipcruise030292.script
7> **
8> ** Update of Ship and Cruise in SouthernOceanDB
9> **
10> ** Data from V. Guretzky
11> **
12> */
13>
14> /* assume SouthernOceanDB is available */
15> use SouthernOceanDB
16>
17> exec insertOneShip "Spectrum", "USSR"
Ship Spectrum, Country USSR inserted
(return status = 0)
18>
19> exec insertOneShip "Quantum", "USSR"
Ship Quantum, Country USSR inserted
(return status = 0)
20>
21> exec changeShipNameInCruise "unknown", "Spectrum", 11319, 0
Update of unknown to Spectrum successful.
(return status = 0)
22>
23> exec changeShipNameInCruise "unknown", "Quantum", 9890, 0
Update of unknown to Quantum successful.
(return status = 0)
24>
25> exec changeShipNameInCruise "unknown", "Skif", 9230, 0
Update of unknown to Skif successful.
(return status = 0)
/* SYBASE script by L.-P. Kurdelski */

** Inserting new Ship by presenting the Ship_Name, the Country **

created 1991-02-03

FILE: INSERTONESHIP.SCRIPT

use master

if not exists (select * from master.dbo.sysdatabases

where name = "SouthernOceanDB")

begin

print 'SouthernOceanDB does not exist !!!'

return

end

use SouthernOceanDB

if exists (select *

from sysobjects

where name like "insertOneShip")

drop proc insertOneShip

create proc insertOneShip (@ship varchar(80),

@country varchar(80))

as

/* SYBASE script by L.-P. Kurdelski */

** Inserting new Ship by presenting the Ship_Name, the Country **

created 1991-02-03

FILE: INSERTONESHIP.SCRIPT

*/

begin

declare @shipid int,

@msg varchar(80)

select @msg = "Ship " + @ship + ", Country " + @country

if exists (select Ship_Id#

from Ship

where Ship_Name like @ship

and Country like @country)

begin

select @msg = @msg + " already exists"

end

else

begin

select @shipid = (select max(Ship_Id#)

from Ship

where Ship_Id# < 100000)

if @shipid is null

select @shipid = 1

else

select @shipid = @shipid + 1

insert Ship

values (@shipid, @ship, @country, NULL)

select @msg = @msg + " inserted"

end
43> print @msg
44> end
Dieses Script geht davon aus, dass die Datenbank SouthernOceanDB bereits existiert.

* Erzeugen der Datenbank

Die Datenbank muss nicht mehr erzeugt werden.

if not exists (select * from master.dbo.sysdatabases
    where name = "SouthernOceanDB")
begin
    print 'SouthernOceanDB does not exist !!!'
    return
end

use SouthernOceanDB (0 rows affected)

* Erzeugen der Tabellen *

if exists (select * from sysobjects where name="BSH_Station")
drop table BSH_Station
create table BSH_Station
    (BSH_Station_Id# int, /* internal identification */
     Cruise_Number int, /* BSH nc */
     Station_Number int, /* BSH ns */
     Longitude float, /* BSH longitude */
     Latitude float, /* BSH attitude */
     Date_Time datetime NULL,
     Bottom_Depth int NULL,/* BSH nyear nmonth nda nho */
     Max_Obse_Depth int NULL,/* BSH mode */
     Number_Obse int, /* BSH nz */
     Marsden_Square# int NULL /* BSH msq */)

if exists (select * from sysobjects where name="BSH_Standard_Data")
drop table BSH_Standard_Data
create table BSH_Standard_Data
    (BSH_Standard_Data_Id# int, /* internal identification */
     BSH_Station_Id# int, /* internal identification */
     Depth int, /* BSH zst */
     Temperature float NULL, /* BSH tst */
     Salinity float NULL, /* BSH sst */
     Oxygen float NULL, /* BSH Oxygen */
     Phosphate float NULL, /* BSH Phosphate */
     Silicate float NULL, /* BSH Silicate */
     Nitrage float NULL /* BSH Nitrage */)

/* Definition der Primärschlüssel in den Tabellen */
execute sp_primarykey BSH_Station, BSH_Station_Id#
execute sp_primarykey BSH_Standard_Data, BSH_Standard_Data_Id#
New primary key added.
New primary key added.
/* Definition der Sekundärschlüssel in den Tabellen */
execute sp_foreignkey BSH_Standard_Data, BSH_Station,
BSH_Station_Id#
New foreign key added.
(return status = 0)

/* Erzeugen der Views */
/* Erzeugen der Prozeduren */
/* Eintragen der Nutzer */
grant select on BSH_Station to PHYSIK1
grant select on BSH_Standard_Data to PHYSIK1
/*
 ** SYBASE Script
 ** Created: 1992-02-04
 ** Creator: L.P. Kurdelski
 ** FILE: BSHLOAD_UPD.SCRIPT
 **
 ** Script um missing Values auf NULL zu setzen
 **
 ** BSH_Station
 ** BSH_Standard_Data
 */

use SouthernOceanDB

dump tran SouthernOceanDB with truncate_only
update BSH_Station
set Date_Time = NULL
where Date_Time="Jan 01 1900 00:00:00.0"
(0 rows affected)
dump tran SouthernOceanDB with truncate_only
update BSH_Station
set Bottom_Depth = NULL
where Bottom_Depth = -9999
(179 rows affected)
dump tran SouthernOceanDB with truncate_only
update BSH_Standard_Data
set Temperature = NULL
where Temperature < -10.0
(127 rows affected)
dump tran SouthernOceanDB with truncate_only
update BSH_Standard_Data
set Salinity = NULL
where Salinity < 0
(243 rows affected)
dump tran SouthernOceanDB with truncate_only
update BSH_Standard_Data
set Oxygen = NULL
where Oxygen < 0
(2405 rows affected)
dump tran SouthernOceanDB with truncate_only
update BSH_Standard_Data
set Phosphate = NULL
where Phosphate < 0
(4259 rows affected)
dump tran SouthernOceanDB with truncate_only
update BSH_Standard_Data
set Silicate = NULL
where Silicate < 0
(4259 rows affected)
dump tran SouthernOceanDB with truncate_only
update BSH_Station
set Marsden_Square# = NULL
where Marsden_Square# <= 0
(0 rows affected)
dump tran SouthernOceanDB with truncate_only
use SouthernOceanDB

if @@error != 0 begin
    print 'Fehler beim Öffnen der Datenbank.'
    return
end

if exists (select * from sysobjects where name like 'BSH_copy_all')
    drop procedure BSH_copy_all

create procedure BSH_copy_all
    as
/* Fill Station and Standard_Data with BSH Data*/
/* L.-P. Kurdelski 1992-02-04*/

begin
    insert Station (Station_Id#, Cruise_Number, Station_Number,
        Date_Time, Longitude, Latitude, Bottom_Depth, Max_Obse_Depth,
        Number_Obse, Marsden_Square#)
        select BSH_Station_Id#, Cruise_Number, Station_Number,
        Date_Time, Longitude, Latitude, Bottom_Depth, Max_Obse_Depth,
        Number_Obse, Marsden_Square#
        from BSH_Station

    insert Standard_Data (Standard_Data_Id#, Station_Id#, Depth, Temperature, Salinity)
        select BSH_Standard_Data_Id#, BSH_Station_Id#, Depth, Temperature, Salinity
        from BSH_Standard_Data

end

execute BSH_copy_all
(return status = 0)
Dieses Script geht davon aus, dass die Datenbank SouthernOceanDB bereits existiert.

/* Erzeugen der Datenbank */
/* Die Datenbank muss nicht mehr erzeugt werden. */
if not exists (select *
from master.dbo.sysdatabases
where name = "SouthernOceanDB")
begin
print 'SouthernOceanDB does not exist !!!'
return
end
(0 rows affected)

use SouthernOceanDB

create procedure updateNODC (@ship varchar(80),
@counry varchar(80),
@NODC varchar(80))
as
if exists (select *
from Ship
where Ship_Name like @ship and
Country like @counry)
begin
update Ship
set Ship_Code = @NODC
where Ship_Name like @ship and
Country like @counry
end
else
begin
select @msg = "Ship "+#ship+", "+#counry+" not found."
print @msg
end
1> /*
2> ** SYBASE script
3> **
4> ** CREATOR: Lutz-Peter Kurdelski
5> **
6> ** CREATION: 1991-10-29
7> **
8> ** FILE: updateNODC200292.SCRIPT
9> **
10> * Changes in the Ship, "Cruise Table
11> source. oth$daten: [socean.dhi]cru3.dat
12> *
13> */
14> use SouthernOceanDB
15>
16> create table #newships (id int,
17> nodc varchar(5),
18> cruise int,
19> ship varchar(80),
20> country varchar(80))
21> begin transaction
22> insert #newships values (1, "GA09", 70133, "Gascoyne", "Australia")
23> insert #newships values (2, "GA09", 70059, "Gascoyne", "Australia")
24> insert #newships values (3, "GA25", 70014, "Galathea", "Denmark")
25> insert #newships values (4, "CN31", 70002, "Carnegie", "USA")
26> insert #newships values (5, "GL31", 70591, "Glacier", "USA")
27> insert #newships values (6, "BI31", 70592, "Burton Island", "USA")
28> insert #newships values (7, "SI31", 70613, "Spaten Island", "USA")
29> insert #newships values (8, "SB31", 70802, "S.F. Baird", "USA")
30> insert #newships values (9, "BI31", 71214, "Burton Island", "USA")
31> insert #newships values (10, "AR31", 72046, "Argo", "USA")
32> insert #newships values (11, "ET31", 72701, "Eltanin", "USA")
33> insert #newships values (12, "OC31", 72994, "Oceanographer", "USA")
34> insert #newships values (13, "CO35", 78348, "Coriolis", "France")
35> insert #newships values (14, "CO35", 78351, "Coriolis", "France")
36> insert #newships values (15, "SY49", 70709, "Albatross", "Japan")
37> insert #newships values (16, "CY49", 70865, "Kaiyo-Maru", "Japan")
38> insert #newships values (17, "K649", 71003, "Koyo-Maru", "Japan")
39> insert #newships values (18, "K649", 71481, "Koyo-Maru", "Japan")
40> insert #newships values (19, "K649", 71481, "Koyo-Maru", "Japan")
41> insert #newships values (20, "TI61", 70004, "Tui", "New Zealand")
42> insert #newships values (21, "TU61", 70005, "Taranui", "New Zealand")
43> insert #newships values (22, "TU61", 70010, "Taranui", "New Zealand")
44> insert #newships values (23, "TU61", 70012, "Tui", "New Zealand")
45> insert #newships values (24, "TU61", 70032, "Taranui", "New Zealand")
46> insert #newships values (25, "TU61", 70033, "Taranui", "New Zealand")
47> insert #newships values (26, "TU61", 70041, "Taranui", "New Zealand")
48> insert #newships values (27, "TU61", 70044, "Viti", "New Zealand")
49> insert #newships values (28, "TU61", 70045, "Tui", "New Zealand")
50> insert #newships values (29, "PD90", 70247, "Prof. Deryugin", "USSR")
51> insert #newships values (30, "AM90", 70303, "Acad. Korolev", "USSR")
52> insert #newships values (31, "V90", 70345, "Vojeikov", "USSR")
53> insert #newships values (32, "V90", 70434, "Vojeikov", "USSR")
54> insert #newships values (33, "AB90", 70499, "Alba", "USSR")
55> insert #newships values (34, "TI90", 70862, "Vityaz", "USSR")
56> insert #newships values (35, "PY90", 77090, "Friboy", "USSR")
(1 row affected)
(1 row affected)
if @@error != 0
rollback transaction
else
commit transaction

declare @i int,
@m int,
@cruise int,
@node varchar(5),
@ship varchar(80),
@country varchar(80)

select @m = max(id) from #newships
select @i = 1

while @i <= @m
begin
select @node = node,
@ship = ship,
@country = country,
@cruise = cruise
from #newships
where id = @i

exec updateNODC @ship, @country, @node
drop table _cruise
exec insert into _cruise @cruise,
select @i = @i + 1
end

(1 row affected)

(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
(1 row affected)
(return status = 0)
(return status = 0)
(1 row affected)
(return status = 0)
(1 row affected)
1>
2> drop table #newships
** **
** SYBASE script **
** CREATOR: Lutz-Peter Kurdelski **
** CREATION: 1992-03-01 **
** FILE: resetNODC200292.SCRIPT **
** RESET **
changes in the Ship, "Cruise Table source. oth$daten:[socean.dhi]bshcru3.dat **

use SouthernOceanDB

create table #newships (id int, node varchar(5), cruise int, ship varchar(80), country varchar(80))

begin transaction
insert #newships values (1, "GA09", 70133, "Gascoyne", "Australia")
insert #newships values (2, "GA09", 70059, "Gascoyne", "Australia")
insert #newships values (3, "GA26", 70014, "Galathea", "Denmark")
insert #newships values (4, "CN31", 70002, "Carnegie", "USA")
insert #newships values (5, "GL31", 70591, "Glacier", "USA")
insert #newships values (6, "BI31", 70592, "Burton Island", "USA")
insert #newships values (7, "ST31", 70613, "Staten Island", "USA")
insert #newships values (8, "SB31", 70802, "S.F. Baird", "USA")
insert #newships values (9, "BI31", 71214, "Burton Island", "USA")
insert #newships values (10, "AR31", 72046, "Argo", "USA")
insert #newships values (11, "ET31", 72701, "Eltanin", "USA")
insert #newships values (12, "OC31", 72994, "Oceanographer", "USA")
insert #newships values (13, "CO35", 78348, "Coriolis", "France")
insert #newships values (14, "CO35", 78351, "Coriolis", "France")
insert #newships values (15, "SY49", 70709, "Albatross", "Japan")
insert #newships values (16, "CY49", 70865, "Koyo-Maru", "Japan")
insert #newships values (17, "K649", 71003, "Kaiyo-Maru", "Japan")
insert #newships values (18, "CY49", 71481, "Koyo-Maru", "Japan")
insert #newships values (19, "TI61", 70004, "Tui", "New Zealand")
insert #newships values (20, "TU61", 70005, "Taranui", "New Zealand")
insert #newships values (21, "TU61", 70010, "Taranui", "New Zealand")
insert #newships values (22, "TI61", 70012, "Tui", "New Zealand")
insert #newships values (23, "TU61", 70032, "Taranui", "New Zealand")
insert #newships values (24, "TU61", 70033, "Taranui", "New Zealand")
insert #newships values (25, "TU61", 70041, "Taranui", "New Zealand")
insert #newships values (26, "VT61", 70044, "Viti", "New Zealand")
insert #newships values (27, "TI61", 70951, "Tui", "New Zealand")
insert #newships values (28, "TI61", 70952, "Tui", "New Zealand")
insert #newships values (29, "PD90", 70247, "Prof. Deryugin", "USSR")
insert #newships values (30, "AM90", 70303, "Acad. Korolev", "USSR")
insert #newships values (31, "VK90", 70345, "Vojeikov", "USSR")
insert #newships values (32, "VK90", 70434, "Vojeikov", "USSR")
insert #newships values (33, "AB90", 70499, "Alba", "USSR")
insert #newships values (34, "VI90", 70862, "Vityaz", "USSR")
insert #newships values (35, "PY90", 77090, "Friboy", "USSR")

(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
1> if @@error != 0
2> rollback transaction
3> else
4> commit transaction
5> declare @i int,
6> @m int,
7> @cruise int,
8> @nodc varchar(5),
9> @ship varchar(80),
10> @country varchar(80)
11> select @m = max(id) from #newships
12> select @i = 1
13> while @i <= @m
14> begin
15> select @nodc = nodc,
16> @ship = ship,
17> @country = country,
18> @cruise = cruise
19> from #newships
20> where id = @i
21> delete from Cruise where Cruise_Number# = @cruise
22> select @i = @i + 1
23> end
(1 row affected)
(5 rows affected)
(1 row affected)
(1 row affected)
(5 rows affected)
(1 row affected)
(1 row affected)
(5 rows affected)
(1 row affected)
(1 row affected)
(5 rows affected)
(1 row affected)
(1 row affected)
(5 rows affected)
(1 row affected)
(1 row affected)
(5 rows affected)
(1 row affected)
(1 row affected)
(5 rows affected)
(1 row affected)
(1 row affected)
(5 rows affected)
(1 row affected)
(1 row affected)
(5 rows affected)
(1 row affected)
(1 row affected)
(5 rows affected)
(1 row affected)
(1 row affected)
(5 rows affected)
(1 row affected)
(1 row affected)
1>
2> drop table #newships
use SouthernOceanDB

drop trigger Update_Cruise

create table #tempships (id# int, 
  cruise int, 
  oldShip varchar(80), 
  newShip varchar(80), 
  oldCountry varchar(80), 
  newCountry varchar(80))

insert #tempships values ( 1, 10522, "unknown", "Foton", "USSR")
insert #tempships values ( 2, 4576, "unknown", "Priliv", "USSR")
insert #tempships values ( 3, 2970, "unknown", "Priboy NPS", "USSR")
insert #tempships values ( 4, 9689, "unknown", "Salehard", "USSR")
insert #tempships values ( 5, 11319, "unknown", "Spektrum", "USSR")
insert #tempships values ( 6, 896, "unknown", "Prof. Zubov", "USSR")
insert #tempships values ( 7, 6468, "unknown", "Hudson", "Canada")
insert #tempships values ( 8, 2968, "unknown", "Okean", "USSR")
insert #tempships values ( 9, 2501, "unknown", "Argus", "USSR")
insert #tempships values (10, 4813, "unknown", "Evrica", "USSR")
insert #tempships values (11, 4375, "unknown", "Lesnoi", "USSR")
insert #tempships values (12, 4395, "unknown", "Profnoz", "USSR")
insert #tempships values (13, 4597, "unknown", "Salehard", "USSR")
insert #tempships values (14, 4564, "unknown", "Acad. Kurchatov", "USSR")
insert #tempships values (15, 3070, "unknown", "Vema", "USA")
insert #tempships values (16, 13968, "unknown", "Yelcho", "Chile")
insert #tempships values (17, 202, "unknown", "Prognoz", "USSR")
insert #tempships values (18, 266, "unknown", "Belogorsk", "USSR")
insert #tempships values (19, 4859, "unknown", "Walther Herwig", "FRG")
insert #tempships values (20, 3997, "unknown", "Dana", "Denmark")
insert #tempships values (21, 891, "unknown", "Davydov", "USSR")
insert #tempships values (22, 890, "unknown", "Bellinsgauzen", "USSR")
<table>
<thead>
<tr>
<th>id#</th>
<th>cruise</th>
<th>oldShip</th>
<th>newShip</th>
<th>oldCountry</th>
<th>newCountry</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10522</td>
<td>unknown</td>
<td>Foton</td>
<td>unknown</td>
<td>USSR</td>
</tr>
<tr>
<td>2</td>
<td>4576</td>
<td>unknown</td>
<td>Priliv</td>
<td>unknown</td>
<td>USSR</td>
</tr>
<tr>
<td>3</td>
<td>2970</td>
<td>unknown</td>
<td>Priboy</td>
<td>NFS</td>
<td>NPS</td>
</tr>
<tr>
<td>No.</td>
<td>Name</td>
<td>Country</td>
<td>Code</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>----------</td>
<td>----------</td>
<td>------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>unknown</td>
<td>USSR</td>
<td>9689</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>unknown</td>
<td>USSR</td>
<td>11319</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>unknown</td>
<td>USSR</td>
<td>896</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>unknown</td>
<td>USSR</td>
<td>6468</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>unknown</td>
<td>Canada</td>
<td>2968</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>unknown</td>
<td>USSR</td>
<td>2501</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>unknown</td>
<td>USSR</td>
<td>4813</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>unknown</td>
<td>USSR</td>
<td>4375</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
unknown
Lesnoi
unknown
USSR
12 4395
unknown
Prognoz
unknown
USSR
13 4597
unknown
Salehard
unknown
USSR
14 4564
unknown
Acad. Kurchatov
unknown
USSR
15 3070
unknown
Vema
unknown
USA
16 13968
unknown
Yelcho
unknown
Chile
17 202
unknown
Prognoz
unknown
USSR
18 266
unknown
Belogorsk
unknown
<table>
<thead>
<tr>
<th>Country</th>
<th>ID</th>
<th>Name</th>
<th>Old</th>
<th>New</th>
</tr>
</thead>
<tbody>
<tr>
<td>USSR</td>
<td>19</td>
<td>Walther Herwig</td>
<td>unknown</td>
<td>4859</td>
</tr>
<tr>
<td>FRG</td>
<td>20</td>
<td>Dana</td>
<td>unknown</td>
<td>3997</td>
</tr>
<tr>
<td>Denmark</td>
<td>21</td>
<td>Davydov</td>
<td>unknown</td>
<td>891</td>
</tr>
<tr>
<td>USSR</td>
<td>22</td>
<td>Bellinsgauen</td>
<td>unknown</td>
<td>890</td>
</tr>
<tr>
<td>USSR</td>
<td>23</td>
<td>Quantum</td>
<td>unknown</td>
<td>9890</td>
</tr>
</tbody>
</table>

(23 rows affected)

```sql
DECLARE @os VARCHAR(80), @oc VARCHAR(80), @ns VARCHAR(80), @nc VARCHAR(80), @er INT, @id INT, @ma INT, @msg VARCHAR(80)

SELECT @id = 1
SELECT @ma = MAX(id#) FROM #tempships
WHILE @id <= @ma
BEGIN
    SELECT @os = oldShip,
           @oc = oldCountry,
           @ns = newShip,
```

```sql
@nc = newCountry,
@cr = cruise
from #tempships
where id# = @id
select @msg = @os+" "+@oc+" "+@ns+" "+@nc
print @msg
exec changeShipNameAndCountryInCru @os, @oc, @ns, @nc, @cr, 0
select @id = @id + 1
end
```

unknown unknown - Foton USSR
Update of unknown to Foton successful.
(return status = 0)
unknown unknown - Priliv USSR
Update of unknown to Priliv successful.
(return status = 0)
unknown unknown - Priboy NPS USSR
Update of unknown to Priboy NPS successful.
(return status = 0)
unknown unknown - Salehard USSR
Update of unknown to Salehard successful.
(return status = 0)
unknown unknown - Spektrum USSR
Update of unknown to Spektrum successful.
(return status = 0)
unknown unknown - Prof. Zubov USSR
Update of unknown to Prof. Zubov successful.
(return status = 0)
unknown unknown - Hudson Canada
Update of unknown to Hudson successful.
(return status = 0)
unknown unknown - Okean USSR
Update of unknown to Okean successful.
(return status = 0)
unknown unknown - Argus USSR
Update of unknown to Argus successful.
(return status = 0)
unknown unknown - Evrica USSR
Update of unknown to Evrica successful.
(return status = 0)
(1 row affected)
(1 row affected)
(1 row affected)
unknown unknown - Lesnoi USSR
Update of unknown to Lesnoi successful.
(return status = 0)
(1 row affected)
(1 row affected)
(1 row affected)
unknown unknown - Prognoz USSR
Update of unknown to Prognoz successful.
(return status = 0)
(1 row affected)
(1 row affected)
(1 row affected)
unknown unknown - Salehard USSR
Update of unknown to Salehard successful.
(return status = 0)
(1 row affected)
(1 row affected)
(1 row affected)
unknown unknown - Acad. Kurchatov USSR
Update of unknown to Acad. Kurchatov successful.
(return status = 0)
(1 row affected)
(1 row affected)
(1 row affected)
unknown unknown - Verna USA
Update of unknown to Verna successful.
(return status = 0)
(1 row affected)
(1 row affected)
(1 row affected)
unknown unknown - Yelcho Chile
Update of unknown to Yelcho successful.
(return status = 0)
(1 row affected)
(1 row affected)
(1 row affected)
unknown unknown - Prognoz USSR
Update of unknown to Prognoz successful.
(return status = 0)
(1 row affected)
(1 row affected)
(1 row affected)
unknown unknown - Belogorsk USSR
Update of unknown to Belogorsk successful.
(return status = 0)
(1 row affected)
(1 row affected)
(1 row affected)
unknown unknown - Walther Herwig FRG
Update of unknown to Walther Herwig successful.
(return status = 0)
(1 row affected)
(1 row affected)
(1 row affected)
unknown unknown - Dana Denmark
Update of unknown to Dana successful.
(return status = 0)
(1 row affected)
(1 row affected)
(1 row affected)
unknown unknown - Davydov USSR
Update of unknown to Davydov successful.
(return status = 0)
(1 row affected)
(1 row affected)
(1 row affected)
unknown unknown - Bellingsgausen USSR
Update of unknown to Bellingsgausen successful.
(return status = 0)
(1 row affected)
(1 row affected)
(1 row affected)
unknown unknown - Quantum USSR
Update of unknown to Quantum successful.
(return status = 0)
(1 row affected)
>
2> drop table #tempships
use SouthernOceanDB

exec insert_into_cruise 60001, "unknown", "unknown"
exec insert_into_cruise 60002, "unknown", "unknown"
exec insert_into_cruise 60003, "unknown", "unknown"
exec insert_into_cruise 60004, "unknown", "unknown"
exec insert_into_cruise 58806, "unknown", "Argentina"
exec insert_into_cruise 999, "Lena", "USSR"

(return status = 0)
(return status = 0)
(return status = 0)
(return status = 0)
(return status = 0)

Sybase script

Written 1992-03-27 by L.-P. Kurdelski

Update of Ship and Cruise in SouthernOceanDB

Data from V. Guretzky, file kurdel21.note

File: UpdateShipCruise270392.script

Assume SouthernOceanDB is available

use SouthernOceanDB

done.

exec changeCountryOfShip "Argo", "Canada", "USA", -987

done.

exec changeCountryOfShip "Staten Island", "Australia", "USA", -927

done.

exec changeShipNameAndCountryInCruise "unknown", "unknown", "Melville", "USA", 60001, 0

Update of unknown to Melville successful.

exec changeShipNameAndCountryInCruise "unknown", "unknown", "Hakuho-Maru", "Japan", 60002, 0

Update of unknown to Hakuho-Maru successful.

exec changeShipNameAndCountryInCruise "unknown", "unknown", "Hudson", "Canada", 60003, 0

Update of unknown to Hudson successful.

declare @maxid int

select @maxid = max(Ship_Id#) from Ship

where Ship_Id# < 100000

select @maxid = @maxid + 1

insert Ship

values (@maxid,"T.G.Thompson","USA",NULL)

exec changeShipNameAndCountryInCruise

"J.G.Thompson", "USA", "T.G.Thompson", "USA", 12219, 0

(1 row affected)

exec changeShipNameAndCountryInCruise

"Th. Thompson", "USA", "T.G.Thompson", "USA", 12220, 0

(1 row affected)

Update of J.G.Thompson to T.G.Thompson successful.

Update of Th. Thompson to T.G.Thompson successful.

update Ship

set Country = "Germany"

WHERE Ship_Name like "Mowe"

and Country like "unknown"

(1 row affected)
/*
** SYBASE script
**
** AUTHOR: Lutz-Peter Kurdelski
**
** DATE: 1992-03-27
**
** WHY: changing a country of a ship
**
** FILE: changeCountryOfShip.script
*/

use SouthernOceanDB

if exists ( select * from sysobjects where name like "changeCountryOfShip")
    drop procedure changeCountryOfShip

create proc changeCountryOfShip
    @ship varchar(40),
    @oldCountry varchar(40),
    @newCountry varchar(40),
    @cru int
as
begin
    /*
    ** Change the ships country
    */
    declare @sid int,
            @msg varchar(80)

    begin transaction

    select @sid = Ship_Id#
    from Cruise
    where Cruise_Number# = @cru

    if @sid is not NULL
        begin
            update Ship
            set Country = @newCountry
            where Ship_Id# = @sid

            select @msg = "Done."

            commit transaction
        end
    else
        begin
            select @msg = "Ship '"+@ship+'" not found"

            rollback transaction
        end

    print @msg
end
Sybase script

Written 1992-04-06 by L.-P. Kurdelski

Update of Ship and Cruise in SouthernOceanDB

Data from V. Guretzky, file kurdel22.note

File: UpdateShipCruise060492.script

Update of unknown to Dana successful. (return status = 0)

Update of Explorer to Explorer successful. (return status = 0)

Ship Polarstern and Cruise 59001 inserted
(5 rows affected, return status = 0)

Ship Polarstern and Cruise 59002 inserted
(5 rows affected, return status = 0)

Ship Polarstern and Cruise 59003 inserted

Ship

------------------ - --------------------------------- - ----------------- - ---------
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern

Ship Polarstern and Cruise 59003 inserted

(5 rows affected, return status = 0)
1> exec insertShip "Polarstern", "FRG", 59004

Cruise_Number
Ship
------------------ - --------------------------------- - ----------------- - ---------
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern

Ship Polarstern and Cruise 59004 inserted

(5 rows affected, return status = 0)
1> exec insertShip "Polarstern", "FRG", 59005

Cruise_Number
Ship
------------------ - --------------------------------- - ----------------- - ---------
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern

Ship Polarstern and Cruise 59005 inserted

(5 rows affected, return status = 0)
1> exec insertShip "Polarstern", "FRG", 59006

Cruise_Number
Ship
------------------ - --------------------------------- - ----------------- - ---------
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern
54019
Polarstern

54019
Polarstern
54019
Polarstern

Ship Polarstern and Cruise 59006 inserted

(5 rows affected, return status = 0)
/* ** SYBASE Script
4> ** Created: 1991-10-09
5> ** Creator: L.P. Kurdelski
6> ** FILE: ARGENTINELOAD_UPD.SCRIPT
7> **
8> ** Script um missing Values auf NULL zu setzen
9> **
10> ** Argentine_Station
11> ** Argentine_Standard_Data
12> */
13> use SouthernOceanDB
14> dump tran SouthernOceanDB with truncate_only
15> update Argentine_Station
16> set Date_Time = NULL
17> where Date_Time="Jan 01 1900 00:00:00.0"
(3 rows affected)
18> dump tran SouthernOceanDB with truncate_only
19> update Argentine_Station
20> set Bottom_Depth = NULL
21> where Bottom_Depth = -9999
(0 rows affected)
22> dump tran SouthernOceanDB with truncate_only
23> update Argentine_Standard_Data
24> set Temperature = NULL
25> where Temperature< -10.0
(1008 rows affected)
26> dump tran SouthernOceanDB with truncate_only
27> update Argentine_Standard_Data
28> set Salinity = NULL
29> where Salinity < 0
(2477 rows affected)
30> dump tran SouthernOceanDB with truncate_only
31> update Argentine_Standard_Data
32> set Oxygen = NULL
33> where Oxygen <= -9.0
(17579 rows affected)
34> dump tran SouthernOceanDB with truncate_only
35> update Argentine_Station
36> set Marsden_Square# = NULL
37> where Marsden_Square# <= 0
(0 rows affected)
38> dump tran SouthernOceanDB with truncate_only
/* SYBASE Script
** Created 1991-11-11
** Creator: L.-P. Kurdelski
** FILE: AWILOAD_COPY.SCRIPT
*/

use SouthernOceanDB

if @@error != 0
begin
print 'Fehler beim Oeffnen der Datenbank.'
return
end

if exists (select * from sysobjects where name like "AWI_copy_all")

drop procedure AWI_copy_all

create procedure AWI_copy_all

as
/* Fill Station and Standard_Data with AWI Data*/
L.-P. Kurdelski 1991-11-11*/

begin

insert Station (Station_Id#, Cruise_Number, Station_Number, Date_Time, Longitude, Latitude, Bottom_Depth, Max_Obse_Depth, Number_Obse, Marsden_Square#)

select AWI_Station_Id#, Cruise_Number, Station_Number, Date_Time, Longitude, Latitude, Bottom_Depth, Max_Obse_Depth, Number_Obse, Marsden_Square#

insert Standard_Data (Standard_Data_Id#, Station_Id#, Depth, Temperature, Salinity)

select AWI_Standard_Data_Id#, AWI_Station_Id#, Depth, Temperature, Salinity

from AWI_Standard_Data

end

execute AWI_copy_all

(return status = 0)
1> /*
2> ** SYBASE Script
3> **
4> ** Created: 1991-11-11
5> **
6> ** Creator: L.P. Kurdelski
7> **
8> ** FILE: AWILOAD_UPD.SCRIPT
9> **
10> ** Script um missing Values auf NULL zu setzen
11> **
12> ** AWI_Station
13> ** AWI_Standard_Data
14> */
15> use SouthernOceanDB
16> dump tran SouthernOceanDB with truncate_only
17> update AWI_Station
18> set Date_Time = NULL
19> where Date_Tirne="Jan 01 1900 00:00:00.0"
20> (1 row affected)
21> dump tran SouthernOceanDB with truncate_only
22> update AWI_Station
23> set Bottorn_Depth = NULL
24> where Bottorn_Depth = -9999
25> (0 rows affected)
26> dump tran SouthernOceanDB with truncate_only
27> update AWI_Standard_Data
28> set Temperature = NULL
29> where Temperature < -10.0
30> (0 rows affected)
31> dump tran SouthernOceanDB with truncate_only
32> update AWI_Standard_Data
33> set Salinity = NULL
34> where Salinity < 0
35> (3 rows affected)
36> dump tran SouthernOceanDB with truncate_only
37> update AWI_Station
38> set Marsden_Square# = NULL
39> where Marsden_Square# <= 0
40> (536 rows affected)
41> dump tran SouthernOceanDB with truncate_only
/*
** SYBASE script
**
** CREATED: 1991-10-09
**
** CREATOR: L.-P. Kurdelski
**
** FILE: CHANGE_VALIDATION_FLAGS.SCRIPT
**
** ABSTRACT: The validation flags will be set to a default of 0 instead of NULL.
** This is a correction to the existing data. After this correction is done there will be a default bound to the column of every table containing a validation flag that will be set the validation flag to zero.
**
*/

use SouthernOceanDB

/*
** update Station
**
** set Validation_Flag = 0
**
** where Validation_Flag is null
**
** go
**
** dump tran SouthernOceanDB with no_log
**
** go
**
*/

if exists (select * from sysobjects where name like 'updateValidationFlags')
    drop procedure updateValidationFlags

create procedure updateValidationFlags (@start int, @end int)
    as
begin
    update Standard_Data
    set Validation_Flag = 0
    where Validation_Flag is null
    and Standard_Data_Id# between @start and @end
    if @@error != 0
        print "no update"
    else
        print "updated"
end

/* exec updateValidationFlags 1, 400000
**
** dump tran SouthernOceanDB with no_log
**
** go
**
** exec updateValidationFlags 400001, 500000
**
** go
**
** dump tran SouthernOceanDB with no_log
**
** go
**
** exec updateValidationFlags 500001, 750000
**
** go
**
** dump tran SouthernOceanDB with no_log
**
** go
**
** exec updateValidationFlags 750001, 1000000
**
** go
**
** dump tran SouthernOceanDB with no_log
**
** go
**
** exec updateValidationFlags 1000001, 1250000
**
** go
**
** dump tran SouthernOceanDB with no_log
**
** go
**
** exec updateValidationFlags 1250001, 2000000
*/
** go
?
dump tran SouthernOceanDB with no_log
Msg 113, Level 15, State 1:
Server 'SYBASE401', Line 26:
Missing end comment mark '*/'.
> exec updateValidationFlags 3000001, 3500000
Msg 1105, Level 17, State 1:
Server 'SYBASE401', Procedure 'UpdateStandard_DataTgr', Line 31:
Can't allocate space for object 'UpdateData' in database 'SouthernOceanDB' because the 'default' segment is full. If you ran out of space in Syslogs, dump the transaction log. Otherwise, use ALTER DATABASE or sp_extendsegment to increase the size of the segment.
(return status = -7)
> dump tran SouthernOceanDB with no_log
> exec updateValidationFlags 3500001, 4000000
No rows selected !! Update aborted
(return status = 0)
> dump tran SouthernOceanDB with no_log
> exec updateValidationFlags 4000001, 4500000
Msg 1105, Level 17, State 1:
Server 'SYBASE401', Procedure 'UpdateStandard_DataTgr', Line 31:
Can't allocate space for object 'UpdateData' in database 'SouthernOceanDB' because the 'default' segment is full. If you ran out of space in Syslogs, dump the transaction log. Otherwise, use ALTER DATABASE or sp_extendsegment to increase the size of the segment.
(return status = -7)
> dump tran SouthernOceanDB with no_log
> exec updateValidationFlags 4500001, 5000000
No rows selected !! Update aborted
(return status = 0)
> dump tran SouthernOceanDB with no_log
> exec updateValidationFlags 5000001, 5500000
Msg 1105, Level 17, State 1:
Server 'SYBASE401', Procedure 'UpdateStandard_DataTgr', Line 31:
Can't allocate space for object 'UpdateData' in database 'SouthernOceanDB' because the 'default' segment is full. If you ran out of space in Syslogs, dump the transaction log. Otherwise, use ALTER DATABASE or sp_extendsegment to increase the size of the segment.
(return status = -7)
> dump tran SouthernOceanDB with no_log
> exec updateValidationFlags 5500001, 6000000
No rows selected !! Update aborted
(return status = 0)
> dump tran SouthernOceanDB with no_log
> exec updateValidationFlags 6000001, 6500000
Msg 1105, Level 17, State 1:
Server 'SYBASE401', Procedure 'UpdateStandard_DataTgr', Line 31:
Can't allocate space for object 'UpdateData' in database 'SouthernOceanDB' because the 'default' segment is full. If you ran out of space in Syslogs, dump the transaction log. Otherwise, use ALTER DATABASE or sp_extendsegment to increase the size of the segment.
(return status = -7)
> dump tran SouthernOceanDB with no_log
> exec updateValidationFlags 6500001, 7000000
No rows selected !! Update aborted
(return status = 0)
> dump tran SouthernOceanDB with no_log
> exec updateValidationFlags 7000001, 8000000
Msg 1105, Level 17, State 1:
Server 'SYBASE401', Procedure 'UpdateStandard_DataTgr', Line 31:
Can't allocate space for object 'UpdateData' in database
'SouthernOceanDB' because the 'default' segment is full. If you ran out of space in Syslogs, dump the transaction log. Otherwise, use ALTER DATABASE or sp_extendsegment to increase the size of the segment.

1> dump tran SouthernOceanDB with no_log
1> exec updateValidationFlags 7000001, 7500000
Msg 1105, Level 17, State 1:
Server 'SYBASE401', Procedure 'UpdateStandard_DataTgr', Line 31:
Can't allocate space for object 'UpdateData' in database 'SouthernOceanDB' because the 'default' segment is full. If you ran out of space in Syslogs, dump the transaction log. Otherwise, use ALTER DATABASE or sp_extendsegment to increase the size of the segment.

(return status = -7)
1> dump tran SouthernOceanDB with no_log
1> exec updateValidationFlags 7500001, 9000000
No rows selected !! Update aborted
(return status = 0)
1> dump tran SouthernOceanDB with no_log
1> exec updateValidationFlags 9000001, 9500000
Msg 1105, Level 17, State 1:
Server 'SYBASE401', Procedure 'UpdateStandard_DataTgr', Line 31:
Can't allocate space for object 'UpdateData' in database 'SouthernOceanDB' because the 'default' segment is full. If you ran out of space in Syslogs, dump the transaction log. Otherwise, use ALTER DATABASE or sp_extendsegment to increase the size of the segment.

(return status = -7)
1> dump tran SouthernOceanDB with no_log
1> exec updateValidationFlags 9500001, 10000000
No rows selected !! Update aborted
(return status = 0)
1> dump tran SouthernOceanDB with no_log
1> exec updateValidationFlags 10000001, 15000000
No rows selected !! Update aborted
(return status = 0)
1> dump tran SouthernOceanDB with no_log
1> exec updateValidationFlags 15000001, 20000000
No rows selected !! Update aborted
(return status = 0)
1> dump tran SouthernOceanDB with no_log
1> exec updateValidationFlags 20000001, 25000000
Msg 1105, Level 17, State 1:
Server 'SYBASE401', Procedure 'UpdateStandard_DataTgr', Line 31:
Can't allocate space for object 'UpdateData' in database 'SouthernOceanDB' because the 'default' segment is full. If you ran out of space in Syslogs, dump the transaction log. Otherwise, use ALTER DATABASE or sp_extendsegment to increase the size of the segment.

(return status = -7)
1> dump tran SouthernOceanDB with no_log
1> exec updateValidationFlags 25000001, 30000000
No rows selected !! Update aborted
(return status = 0)
1> dump tran SouthernOceanDB with no_log
1> exec updateValidationFlags 30000001, 35000000
No rows selected !! Update aborted
(return status = 0)
1> dump tran SouthernOceanDB with no_log
1> exec updateValidationFlags 35000001, 40000000
No rows selected !! Update aborted
(return status = 0)
1> print "DONE"
DONE
/** Sybase-Script made in December 1990 by Lutz-Peter Kurdelski
 ** AWI-Rechnergruppe
 ** FILE: cruise_trigger.script
 ** Es werden Trigger auf delete, insert, update gelegt,
 ** so dass nachvollziehbar ist, wer was wann geendert hat.
 ** Dieses Script geht davon aus, dass die Datenbank SouthernOceanDB
 ** bereits existiert.
 */

use master
/* Erzeugen der Datenbank
 ** Die Datenbank muss nicht mehr erzeugt werden.
 */
if not exists (select *
from master.dbo.sysdatabases
where name = "SouthernOceanDB")
begin
print 'Southern OceanDB does not exist !!!'
return
end

use SouthernOceanDB
(0 ·ows affected)
/* Erzeugen von Trigger */
if exists ( select *
from sysobjects where name like "Insert_Cruise"
) drop trigger Insert_Cruise

create trigger Insert_Cruise
on Cruise
for insert
as
begin
if not exists (select *
from Ship, inserted
where Ship.Ship_Id# = inserted.Ship_Id#)
begin
select @sid = Ship_Id#
from inserted
select @msg = "INSERT Cruise: Please insert the ship #" +
convert(varchar(20),@sid) + " first into Ship"
print @msg
rollback transaction
end
if exists ( select *
from sysobjects where name like "Delete_Cruise"
) drop trigger Delete_Cruise

create trigger Delete_Cruise
on Cruise
for delete
as
begin
if exists (select *
from Station, deleted
where Station.Cruise_Number = deleted.Cruise_Number#)
begin
print "DELETE Cruise: There exists a station corresponding to this Cruise_Number"
rollback transaction
end
else
begin
insert into Cruise Deleted
select deleted.Cruise_Number#, deleted.Ship_Id#, suser_id(), getdate()
from deleted

if exists ( select * from sysobjects where name like "Update_Cruise" )
    drop trigger Update_Cruise

create trigger Update_Cruise
on Cruise
for update
as
begin
    declare @msg varchar(80), @sid int
    if not exists ( select * from Ship, inserted where Ship.Ship_Id# = inserted.Ship_Id# )
        begin
            select @sid = Ship_Id# from inserted
            select @msg = "UPDATE Cruise: Please insert the ship #" +
                convert(varchar(20),@sid) + " first into Ship"
        print @msg
        rollback transaction
        end
    else
        begin
            if update(Cruise_Number#)
                begin
                    if exists ( select * from Station, deleted where Station.Cruise_Number = deleted.Cruise_Number# )
                        begin
                            select @sid = Cruise_Number# from deleted
                            select @msg = "UPDATE Cruise: There exists at least one station"
                                + "(" + convert(varchar(20),@sid) + ")" +
                                + " corresponding to this Cruise_Number"
                            print @msg
                            rollback transaction
                        end
                    else
                        begin
                            insert into Cruise
                                select inserted.Cruise_Number#, inserted.Ship_Id#,
                                    deleted.Cruise_Number#, deleted.Ship_Id#, suser_id(), getdate()
                            print "UPDATE Cruise successful"
                        end
                end
        end
    else
    begin
        insert into Cruise Deleted
        select deleted.Cruise_Number#, deleted.Ship_Id#,
            deleted.Cruise_Number#, deleted.Ship_Info#, suser_id(), getdate()
        from deleted
        print "UPDATE Cruise successful"
    end
end

if exists ( select * from sysobjects where name like "Delete_Cruise" )
    drop trigger Delete_Cruise

create trigger Delete_Cruise
on Ship
for delete
as
begin
declare @msg varchar (80),
@sid int
if exists (select *
from Cruise, deleted
where Cruise.Ship_Id# = deleted.Ship_Id#)
begin -
select @sid = Ship_Id# from deleted
select @msg = "DELETE Ship: Please delete the cruises for ship #" +
convert(varchar(20),@sid) + " first"
print @msg
rollback transaction
end
end

if exists ( select * from sysobjects where name like "Update_Ship" )
drop trigger Update_Ship
create trigger Update_Ship
on Ship
for update
as
begin
insert into Ship_Updated
select deleted.Ship_Id#,
inserted.Ship_Name,
inserted.Country,
inserted.Ship_Code,
deleted.Ship_Name,
deleted.Country,
deleted.Ship_Code,
suser_id(),
getdate()
from inserted, deleted
SYBASE script by L.-P. Kurdelski

** Inserting new Cruises to Cruise and Ship
** from Marion Dufresne Data
** created 1991-09-13
** FILE: MARION_DUFRESNE_CRUISE.SCRIPT
*/

```sql
use master
if not exists (select *
from master.dbo.sysdatabases
where name = "SouthernOceanDB")
begin
print 'SouthernOceanDB does not exist !!!'
return
end

use SouthernOceanDB
(0 rows affected)

exec insertShip 'Marion Dufresne', 'France', 57001
Ship

```

Ship Marion Dufresne and Cruise 57001 inserted

(17 rows affected, return status = 0)

```sql
exec insertShip 'Marion Dufresne', 'France', 57002
Ship

```
<table>
<thead>
<tr>
<th>Cruise_Number</th>
<th>Ship</th>
</tr>
</thead>
<tbody>
<tr>
<td>55065</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57001</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57002</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57003</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57004</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57005</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57006</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57007</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>55065</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57002</td>
<td>Marion Dufresne</td>
</tr>
</tbody>
</table>

Ship Marion Dufresne and Cruise 57002 inserted

(17 rows affected, return status = 0)

1>
2> exec insertShip 'Marion Dufresne', 'France', 57003

<table>
<thead>
<tr>
<th>Cruise_Number</th>
<th>Ship</th>
</tr>
</thead>
<tbody>
<tr>
<td>55065</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57001</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57002</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57003</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57004</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57005</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57006</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57007</td>
<td>Marion Dufresne</td>
</tr>
</tbody>
</table>
55065
Marion Dufresne
-21420
Marion Dufresne
55065
Marion Dufresne
-21420
Marion Dufresne
55065
Marion Dufresne
-21420
Marion Dufresne
55065
Marion Dufresne
-21420
Marion Dufresne
55065
Marion Dufresne
-21420
Marion Dufresne
Ship Marion Dufresne and Cruise 57003 inserted

(17 rows affected, return status = 0)

1>
2> exec insertShip 'Marion Dufresne', 'France', 57004
Cruise_Number
Ship

<table>
<thead>
<tr>
<th>Ship</th>
<th>Marion Dufresne and Cruise 57004 inserted</th>
</tr>
</thead>
<tbody>
<tr>
<td>55065</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td></td>
<td>-21420</td>
</tr>
<tr>
<td></td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57001</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57002</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57003</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57004</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57005</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57006</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57007</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>55065</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td></td>
<td>-21420</td>
</tr>
<tr>
<td></td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>55065</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td></td>
<td>-21420</td>
</tr>
<tr>
<td></td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>55065</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td></td>
<td>-21420</td>
</tr>
<tr>
<td></td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>55065</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td></td>
<td>-21420</td>
</tr>
<tr>
<td></td>
<td>Marion Dufresne</td>
</tr>
</tbody>
</table>

Ship Marion Dufresne and Cruise 57004 inserted

(17 rows affected, return status = 0)
<table>
<thead>
<tr>
<th>Cruise_Number</th>
<th>Ship</th>
</tr>
</thead>
<tbody>
<tr>
<td>55065</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>-21420</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57001</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57002</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57003</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57004</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57005</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57006</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>57007</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>55065</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>-21420</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>55065</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>-21420</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>55065</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>-21420</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>55065</td>
<td>Marion Dufresne</td>
</tr>
<tr>
<td>-21420</td>
<td>Marion Dufresne</td>
</tr>
</tbody>
</table>

Ship Marion Dufresne and Cruise 57005 inserted

(17 rows affected, return status = 0)
Ship Marion Dufresne and Cruise 57006 inserted

(17 rows affected, return status = 0)

1> 2> exec insertShip 'Marion Dufresne', 'France', 57007
Cruise_Number
Ship

--------------------------------------------------------------------------------·

Ship Marion Dufresne and Cruise 57007 inserted

(17 rows affected, return status = 0)
/* SYBASE script by L.-P. Kurdelski
**
** Inserting new Cruises to Cruise and Ship
** from Marion Dufresne Data
**
** created 1991-09-13
**
** FILE: MARION_DUFRESNE_CRUISE.SCRIPT
*/

use master
go
if not exists (select * from master.dbo.sysdatabases
    where name = 'SouthernOceanDB')
begin
    print 'SouthernOceanDB does not exist !!!'
    return
end

use SouthernOceanDB

exec insertShip 'Marion Dufresne', 'France', 57001
exec insertShip 'Marion Dufresne', 'France', 57002
exec insertShip 'Marion Dufresne', 'France', 57003
exec insertShip 'Marion Dufresne', 'France', 57004
exec insertShip 'Marion Dufresne', 'France', 57005
exec insertShip 'Marion Dufresne', 'France', 57006
exec insertShip 'Marion Dufresne', 'France', 57007

/* end of Marion_Dufresne_Cruise.script */
Dieses Script geht davon aus, dass die Datenbank SouthernOceanDB bereits existiert.

use master

if not exists (select *
from master.dbo.sysdatabases
where name = "SouthernOceanDB")
begin
print 'SouthernOceanDB does not exist !!!'
return
end

use SouthernOceanDB

create default Validation_Flag_Default

exec sp_bindefault Validation_Flag_Default, "Station. Validation_Flag"
Default bound to column.

exec sp_bindefault Validation_Flag_Default, "Standard_Data. Validation_Flag"
Default bound to column.
use SouthernOceanDB

if exists (select * from sysobjects where name like "updateValidationFlags")
    drop procedure updateValidationFlags

create procedure updateValidationFlags (@start int, @end int)
    as
    begin
        update Standard_Data
        set Validation_Flag = 0
        where Validation_Flag is null
        and Standard_Data_Id# between @start and @end
        if @@error != 0
            print "no update"
        else
            print "updated"
    end

exec updateValidationFlags 1250001, 2000000
updated
(return status = 0)

dump tran SouthernOceanDB with truncate_only
Dieses Script geht davon aus, dass die Datenbank SouthernOceanDB bereits existiert.

```sql
use master

if not exists (select * from master.dbo.sysdatabases where name = 'SouthernOceanDB')
begin
  print 'SouthernOceanDB does not exist !!!'
  return
end

exec sp_addlogin "so_user", "ocean", "SouthernOceanDB"
(0 rows affected)
New login created.
(return status = 0)

use SouthernOceanDB

exec sp_adduser so_user
New user added.
(return status = 0)

grant select on Station to so_user

grant select on Standard_Data to so_user
```
use master

if not exists (select * from master.dbo.sysdatabases
    where name = "SouthernOceanDB")
begin
    print 'SouthernOceanDB does not exist !!!'
    return
end

use SouthernOceanDB
update Ship
set Ship_Name = "unknown"
where Ship_Id# = 299
(1 row affected)
1> /* Sybase-Script made in Dec 1993 by M. Reinke */
2> /* AWI-Rechnergruppe */
3> /*
4> ** FILE: AariLdgo.SCRIPT
5> */
6> /* Dieses Script geht davon aus, dass die Datenbank SouthernOceanDB */
7> /* bereits existiert. */
8> use master
9> /* Erzeugen der Datenbank */
10> /* Die Datenbank muss nicht mehr erzeugt werden. */
11> if not exists (select *
12> from master.dbo.sysdatabases
13> where name = "SouthernOceanDB")
14> begin
15> print 'SouthernOceanDB does not exist !!!'
16> return
17> end
18> /* Erzeugen der Tabellen */
19> if exists (select *
20> from sysobjects where name="AariLdgo_Station"
21> ) drop table AariLdgo_Station
22> create table AariLdgo_Station
23> (AariLdgo_Station_Id# int,
24> Cruise_Number int,
25> Station_Number int,
26> Longitude float,
27> Latitude float,
28> Date_Time datetime
29> Bottom_Depth int,
30> Max_Obse_Depth int,
31> Number_Obse int,
32> Marsden_Square# int,
33> /* internal identification */
34> /* AariLdgo nc */
35> /* AariLdgo ns */
36> /* AariLdgo longitude */
37> /* AariLdgo latitude */
38> NULL,
39> /* AariLdgo nyear nmonth nda nho */
40> /* AariLdgo nde */
41> /* AariLdgo mode */
42> /* AariLdgo hz */
43> /* AariLdgo msq */
44> )
45> if exists (select *
46> from sysobjects where name= " AariLdgo_Standard_Data"
47> ) drop table AariLdgo_Standard_Data
48> create table AariLdgo_Standard_Data
49> (AariLdgo_Standard_Data_Id# int,
50> AariLdgo_Station_Id# int,
51> Depth int,
52> Temperature float,
53> Salinity float,
54> Oxygen float
55> /* internal identification */
56> /* internal identification */
57> /* AariLdgo zst */
58> /* AariLdgo tst */
59> /* AariLdgo sst */
60> /* AariLdgo oxst */
61> )
62> /* Definition der Primärschlüssel in den Tabellen */
63> execute sp_primarykey AariLdgo_Station, AariLdgo_Station_Id#
64> execute sp_primarykey AariLdgo_Standard_Data, AariLdgo_Standard_Data_Id#
65> New primary key added.
66> (return status = 0)
67> New primary key added.
68> (return status = 0)
69> /* Definition der Sekundärschlüssel in den Tabellen */
70> execute sp_foreignkey AariLdgo_Standard_Data, AariLdgo_Station,
71> AariLdgo_Station_Id#
72> New foreign key added.
73> (return status = 0)
74> /* Erzeugen der Views */
75> /* Erzeugen der Prozeduren */
76> grant select on AariLdgo_Station to PHYSIK1
77> grant select on AariLdgo_Standard_Data to PHYSIK1