Приложение C

SQL Скрипт за миграция на данни от старата база

*-- ############## Импортиране на основни номенклатури***INSERT INTO "OSTROVA"**.**"CLUB" "CLUB\_ID"**, **"NAME"**, **"ZALA\_PRICE"**, **"ADDRESS" FROM** OSTROVA\_IMP.CLUB;  
**INSERT INTO "OSTROVA"**.**"SALOON" SELECT "ID"**, **"NAME"**, **"CLUB\_ID"**, **'1' FROM** OSTROVA\_IMP.SALON;  
**INSERT INTO "OSTROVA"**.**"SUPPLIER" SELECT "SUP\_ID"**, **"NAME"**, **"DESCRIPTION" FROM** OSTROVA\_IMP.**"SUPPLIER"**;  
  
*-- Поправка на sequence, така че при добавяне на нови записи да не се получи конфликт с прехвърлените от старата  
-- система идентификатори***drop** sequence **"OSTROVA"**.**"CLUB\_SQ"**;  
**CREATE** SEQUENCE **"OSTROVA"**.**"CLUB\_SQ"** START **WITH** 1000;  
  
**drop** sequence **"OSTROVA"**.**"SALOON\_SQ"**;  
**CREATE** SEQUENCE **"OSTROVA"**.**"SALOON\_SQ"** START **WITH** 1000;  
  
**drop** sequence **"OSTROVA"**.**"SUPPLIER\_SQ"**;  
**CREATE** SEQUENCE **"OSTROVA"**.**"SUPPLIER\_SQ"** START **WITH** 1000;  
  
  
 *-- ############## Импортиране на съществуващите потребителски профили,  
 -- парола по подразбиране 12345678* **INSERT INTO** OSTROVA.auth\_user  
 **SELECT** OSTROVA.AUTH\_USER\_SQ.NEXTVAL,  
 **'pbkdf2\_sha256$30000$9aXR09FpiobP$agauh1MJYeL407Z+4rqXUPrX8erGQTRbsTPgD7T2los='**,  
 **null**,  
 1,  
 username,  
 name,  
 grupa,  
 **null**,  
 1,  
 1,  
 sysdate  
**FROM** OSTROVA\_IMP.**"STAFF"**;  
  
*-- ############## Импортиране на артикули - доставка***insert into** ostrova.article\_group  
**select** ostrova.article\_group\_sq.nextval,  
 STOKA\_GROUP,  
 sysdate,  
 1,  
 0  
 **from** (  
 **select** STOKA\_GROUP  
 **from** ostrova\_imp.stoka  
**group by** STOKA\_GROUP);  
  
**insert into** ostrova.article  
**select** stoka\_id,  
 descr,  
 **null**,  
 price,  
 **null**,  
 laST\_PRICE,  
 **null**,  
 LAST\_UPDATE\_DATE,  
 nvl((**select** ID **from** OSTROVA.ARTICLE\_GROUP z **where** z.NAME = X.STOKA\_GROUP),0) GROUP\_ID  
 **from** ostrova\_imp.stoka x  
;  
  
*-- Ръчна Поправка на sequence за артикули,  
-- тъй като ще използваме оригиналните ID-та от старата база, за да се избегне нуждата от прегрупиране***drop** sequence OSTROVA.ARTICLE\_SQ;  
  
**declare** v\_sql varchar2(1000);  
**begin  
SELECT 'CREATE SEQUENCE OSTROVA.ARTICLE\_SQ MINVALUE 0 START WITH '**||to\_char(**MAX**(id)+1)||**' INCREMENT BY 1 '  
 INTO** v\_sql **FROM** OSTROVA.ARTICLE;  
**EXECUTE IMMEDIATE** v\_sql;  
**end**;  
/  
  
*-- Първите два записа в оригиналната база са тестови и са непълни, затова се премахват***delete from** OSTROVA\_IMP.RECORD\_ORDERS **where** order\_id **in** (1,2);  
  
*-- ############## Импортиране на доставки***INSERT INTO** OSTROVA.DELIVERY  
**SELECT**ORDER\_ID x,  
ORDER\_DATE,  
STATUS,  
DOSTAVKA\_DATE,  
FAKT\_NO,  
FIRM\_FAKT\_NO,  
PLATENO,  
NOTES,  
ORDER\_AMOUNT,  
sysdate,  
(**select** ID **from** OSTROVA.CLUB z **where** z.NAME = X.CLUB\_NAME) CLUB\_ID,  
nvl((**select** ID **from** OSTROVA.AUTH\_USER z **where** z.FIRST\_NAME = X.SLUJITEL),1) user\_ID,  
kasa\_no,  
nvl((**select** nvl(ID,0) **from** OSTROVA.SUPPLIER z **where** z.NAME = X.FIRM),0) SUP\_ID  
**FROM** OSTROVA\_IMP.RECORD\_ORDERS X;  
  
*-- ############## Импортиране на доставки - детайни записи***INSERT INTO** OSTROVA.DELIVERY\_DETAIL  
**SELECT**ostrova.delivery\_detail\_sq.nextval,  
cnt,  
price,  
PRICE\_ACTUAL,  
ARTICE\_ID,  
STOKA\_ID,  
ORDER\_ID,  
nvl((**select** ID **from** OSTROVA.ARTICLE\_GROUP z **where** z.NAME = X.GROUP\_NAME),0) GROUP\_ID  
**FROM** OSTROVA\_IMP.RECORD\_ORDERS\_DETAIL X;  
  
  
*-- ############## Импортиране на артикулни - поръчка***INSERT INTO** OSTROVA.ARTICLE\_GROUP  
 **SELECT** ostrova.article\_group\_sq.nextval,  
 NAME,  
 LAST\_UPDATE\_DATE,  
 0,  
 1  
 **FROM** OSTROVA\_IMP.GROUPS;  
  
*-- добавяне на служебна група - артикули с липсваща група (съществувала и изтрита)***insert into** OSTROVA.article\_group **values**(0, **'MISSING'**, sysdate, 0,1);  
  
*-- добавяне на служебна група - липсващи артикули (изтрити)***insert into** OSTROVA.article\_group **values**(1, **'DELETED'**, sysdate, 0,1);  
  
*-- добавяне на "паразитна колона"***alter Table** ostrova.article **add** old\_id number;  
  
*-- ############## Импортиране на артикули - поръчка***INSERT INTO** OSTROVA.ARTICLE  
 **SELECT** OSTROVA.ARTICLE\_SQ.nextval,  
 NAME,  
 **null**,  
 **null**,  
 PRICE,  
 **null**,  
 PRICE,  
 LAST\_UPDATE\_DATE,  
 (**select** ID  
 **from** ostrova.article\_group  
 **where** name = nvl((**select** name **from** OSTROVA\_IMP.GROUPS z **where** z.ID = X.GROUP\_ID),**'MISSING'**) **and** order\_type = 1) grupa,  
 MIARKA,  
 1,  
 ID  
 **FROM** OSTROVA\_IMP.**"ARTIKUL\_TYPES"** x;  
  
*-- Артикулите с липсваща група се правят неактивни***update** OSTROVA.ARTICLE **set** active = 0 **where** group\_fk\_id = 0;  
  
*-- Добавяне на лиспващите стари артикули от старите поръчки (изпълнява се на 3 паса за да се обхванат всички случаи)  
-- Всички такива артикули се добавят към група DELETED (id=1)  
-- Артикулите се добавят само с името си, тъй като присъстват единствено в старата таблица за поръчки  
-- Това е необходимо за да не се наруши master/detail отношението в таблица order/detail  
-- Всички артикули се добавят като неактивни (is\_active=0)***INSERT INTO** OSTROVA.ARTICLE  
**select** OSTROVA.ARTICLE\_SQ.nextval,  
 NAME,  
 **NULL**,  
 **NULL**,  
 PRICE,  
 **NULL**,  
 PRICE,  
 sysdate,  
 0,  
 **null**,  
 0,  
 ARTICLE\_TYPE\_ID  
**from "OSTROVA\_IMP"**.**"RECORD\_ARTICLE"  
where** ARTICLE\_TYPE\_ID **in** (  
 **select  
 max**(ARTICLE\_TYPE\_ID)  
 **from  
 "OSTROVA\_IMP"**.**"RECORD\_ARTICLE"  
 where** ARTICLE\_TYPE\_ID **not in** (**select** old\_id **from** ostrova.article)  
 **group by** NAME  
);  
  
**INSERT INTO** OSTROVA.ARTICLE  
**select** OSTROVA.ARTICLE\_SQ.nextval,  
 NAME,  
 **NULL**,  
 **NULL**,  
 0,  
 **NULL**,  
 **NULL**,  
 **NULL**,  
 sysdate,  
 1, *-- група DELETED* **null**,  
 0, *-- IS\_ACTIVE 0* ARTICLE\_TYPE\_ID  
**from** (  
**select** NAME,  
 ARTICLE\_TYPE\_ID  
**from "OSTROVA\_IMP"**.**"RECORD\_ARTICLE"  
where** ARTICLE\_TYPE\_ID **in** (  
 **select  
 max**(ARTICLE\_TYPE\_ID)  
 **from  
 "OSTROVA\_IMP"**.**"RECORD\_ARTICLE"  
 where** ARTICLE\_TYPE\_ID **not in** (**select** old\_id **from** ostrova.article **where** old\_id **is not null**)  
 **group by** NAME  
) **group by** name, article\_type\_id  
)  
;  
  
**INSERT INTO** OSTROVA.ARTICLE  
**select** OSTROVA.ARTICLE\_SQ.nextval,  
 NAME,  
 **NULL**,  
 **NULL**,  
 0,  
 **NULL**,  
 **NULL**,  
 **NULL**,  
 sysdate,  
 1, *-- група DELETED* **null**,  
 0, *-- IS\_ACTIVE 0* ARTICLE\_TYPE\_ID  
**from** (  
**select** NAME,  
 ARTICLE\_TYPE\_ID  
**from "OSTROVA\_IMP"**.**"RECORD\_ARTICLE"  
where** ARTICLE\_TYPE\_ID **in** (  
 **select  
 max**(ARTICLE\_TYPE\_ID)  
 **from  
 "OSTROVA\_IMP"**.**"RECORD\_ARTICLE"  
 where** ARTICLE\_TYPE\_ID **not in** (**select** old\_id **from** ostrova.article **where** old\_id **is not null**)  
 **group by** NAME  
) **group by** name, article\_type\_id  
)  
;  
  
*-- ############## Импортиране на поръчки***INSERT INTO** OSTROVA.**"ORDER"  
 SELECT** REC\_ID,  
 REC\_DATE,  
 REC\_TIME\_SLOT,  
 REC\_TIME,  
 REC\_TIME\_END,  
 PHONE,  
 PARENT,  
 CHILD,  
 AGE,  
 CHILD\_COUNT,  
 ADULT\_COUNT,  
 ZALA\_COUNT,  
 ZALA\_PRICE,  
 KAPARO,  
 DISCOUNT,  
 PRICE\_FINAL,  
 KAPARO\_DATE,  
 PAYMENT\_DATE,  
 VALIDITY\_DATE,  
 OTKAZ\_DATE,  
 OTKAZ\_REASON,  
 ZAIAVKA\_DATE,  
 NOTES,  
 ADDRESS,  
 EMAIL,  
 CREATE\_DATE,  
 LAST\_UPDATE\_DATE,  
 KAPARO2,  
 KAPARO2\_DATE,  
 UPDATE\_STATE,  
 LOCKED,  
 PAYED\_FINAL,  
 NOTES\_TORTA,  
 NOTES\_KITCHEN,  
 nvl((**select** ID **from** OSTROVA.saloon z **where** z.NAME = X.ADULT\_SALOON **and** z.club\_fk\_id = x.club\_id), 14 ) xx,  
 nvl((**select** ID **from** OSTROVA.AUTH\_USER z **where** z.FIRST\_NAME = X.priel\_porachka),1) user\_ID,  
 CLUB\_ID,  
 CHANGES  
 **FROM** OSTROVA\_IMP.Record\_data x **where** rec\_id > 57;  
  
*-- ############## Импортиране на доставки - детайни записи  
-- Артикулите/Артикулите групи се издирват по OLD\_ID и се прикачат към новите си идентификатори  
-- Артикулите/Артикулите групи, които са били изтрити се издирват само по име***INSERT INTO** OSTROVA.ORDER\_DETAIL  
 **select** OSTROVA.OSTROVACALENDAR\_ORDERDETAIL\_SQ.NEXTVAL,  
 CNT,  
 PRICE,  
 (**select** ID  
 **from** ostrova.article z  
 **where** (z.old\_id = x.ARTICLE\_TYPE\_ID **and** z.name = x.NAME **and** z.group\_fk\_id = 1) **or** (z.old\_id = x.ARTICLE\_TYPE\_ID **and** z.group\_fk\_id <> 1)  
 ) article\_id,  
 (**select** GROUP\_FK\_ID  
 **from** ostrova.article z  
 **where** (z.old\_id = x.ARTICLE\_TYPE\_ID **and** z.name = x.NAME **and** z.group\_fk\_id = 1) **or** (z.old\_id = x.ARTICLE\_TYPE\_ID **and** z.group\_fk\_id <> 1)  
 ) article\_id,  
 REC\_ID  
 **from  
 "OSTROVA\_IMP"**.**"RECORD\_ARTICLE"** x  
 **where** rec\_id **in** (**select** id **from** OSTROVA.**"ORDER"**);  
  
*-- Корекция на лиспащ артикул "Детско меню", който не е бил намерен***INSERT INTO** OSTROVA.ARTICLE  
**values**(  
 OSTROVA.ARTICLE\_SQ.nextval,  
 **'Детско меню'**,  
 **NULL**,  
 **NULL**,  
 0,  
 **NULL**,  
 **NULL**,  
 **NULL**,  
 sysdate,  
 1,  
 **null**,  
 0,  
 9998);  
  
*-- Ръчна корекция на невалидни записи от старата база***update** OSTROVA.ORDER\_DETAIL **set** article\_fk\_id = (**select max**(id) **from** OSTROVA.ARTICLE) , group\_fk\_id = 1 **where** article\_fk\_id **is null and** order\_fk\_id **in** (6148,6205,7815,7856);  
  
**commit**;  
  
**alter Table** ostrova.article **drop column** old\_id;