

Санкт-Петербургский Политехнический университет Петра Великого

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

Отчёт о лабораторной работе №2

Дисциплина: Базы данных

Тема: Изучение языка SQL-DDL

Выполнил студент гр. 43501/1:

_____ Нагорнов А. А

(подпись)

Работу принял:

_____ Мяснов А.В.

(подпись)

«__» _____ 2016 г.

Санкт-Петербург

2016

1. Цели работы

Познакомить студентов с основами проектирования схемы БД, языком описания сущностей и ограничений БД SQL-DDL.

2. Программа работы

- 2.1. Самостоятельное изучение SQL-DDL
- 2.2. Создание скрипта БД в соответствии с согласованной схемой (должны присутствовать первичные и внешние ключи, ограничения на диапазоны значений). Продемонстрировать скрипт преподавателю.
- 2.3. Создайте скрипт, заполняющий все таблицы БД данными
- 2.4. Выполнение SQL-запросов, изменяющих схему созданной БД по заданию преподавателя. Продемонстрировать их работу преподавателю.
- 2.5. Изучите основные возможности IBExpert. Получите ER-диаграмму созданной БД с помощью Database Designer.
- 2.6. Автоматически сгенерируйте данные при помощи IBExpert (для трех или большего числа таблиц, не менее 100000 записей в каждой из выбранных таблиц)

3. Выполнение работы

3.1. Изучение SQL-DDL

Язык DDL служит для создания и модификации структуры БД, т.е. для создания/изменения/удаления таблиц и связей. Основные команды создания и редактирования таблиц:

- CREATE TABLE имя_таблицы (перечисление полей и их типов, ограничений) – служит для создания новой таблицы в текущей БД;
- DROP TABLE имя_таблицы – служит для удаления таблицы из текущей БД;
- ALTER TABLE имя_таблицы ALTER COLUMN имя_столбца ... – служит для обновления типа столбца или для изменения его настроек (например для задания характеристики NULL или NOT NULL);
- ALTER TABLE имя_таблицы ADD CONSTRAINT имя_ограничения PRIMARY KEY(поле1, поле2,...) – добавление первичного ключа к уже существующей таблице;
- ALTER TABLE имя_таблицы DROP CONSTRAINT имя_ограничения – удаление ограничения из таблицы.
- ALTER TABLE имя_таблицы ADD перечень_полей_с_характеристиками – позволяет добавить новые поля в таблицу;
- ALTER TABLE имя_таблицы DROP COLUMN перечень_полей – позволяет удалить поля из таблицы;
- ALTER TABLE имя_таблицы ADD CONSTRAINT имя_ограничения FOREIGN KEY(поля) REFERENCES таблица_справочник(поля) – позволяет определить связь между таблицей и таблицей справочником.

3.2. Создание скрипта БД

Скрипт БД в соответствии с согласованной схемой:

```
create database 'C:\Users\Andrey\Desktop\BD\3\music2.fdb' user 'sysdba' password 'masterkey' ;
connect 'C:\Users\Andrey\Desktop\BD\3\music2.fdb' user 'sysdba' password 'masterkey' ;
```

```
CREATE TABLE Artists (
    Name varchar(50) NOT NULL,
    Biography varchar(255),
    Country varchar(50),
    PRIMARY KEY (Name)
);
CREATE TABLE Country (
    Country varchar(50) NOT NULL,
    PRIMARY KEY (Country)
);
CREATE TABLE Tracks (
    TrackID INT NOT NULL,
    TrackName varchar(50) NOT NULL,
    Artist varchar(50),
    Duration varchar(30),
    YearD INT,
    Language varchar(30),
    PRIMARY KEY (TrackID)
);
CREATE TABLE Albums (
    AlbumID INT NOT NULL,
    Artist varchar(50) NOT NULL,
    AlbumName varchar(50) NOT NULL,
    PRIMARY KEY (AlbumID)
);
CREATE TABLE MusicStyles (
    Style varchar(30) NOT NULL,
    PRIMARY KEY (Style)
);
CREATE TABLE Languages (
    Labguage varchar(30) NOT NULL,
    PRIMARY KEY (Labguage)
);
CREATE TABLE AlbumsSongs (
    ID INT NOT NULL,
    AlbumID INT NOT NULL,
    TrackID INT NOT NULL,
    PRIMARY KEY (ID)
);
CREATE TABLE MusicSOT (
    ID INT NOT NULL,
    StyleID varchar(30),
    TrackID INT,
    PRIMARY KEY (ID)
);
```

```

ALTER TABLE Artists
ADD CONSTRAINT Artists_fk0 FOREIGN KEY (Country) REFERENCES Country(Country);

ALTER TABLE Tracks
ADD CONSTRAINT Tracks_fk0 FOREIGN KEY (Artist) REFERENCES Artists(Name);

ALTER TABLE Tracks
ADD CONSTRAINT Tracks_fk1 FOREIGN KEY (Duration) REFERENCES MusicStyles(Style);

ALTER TABLE Tracks
ADD CONSTRAINT Tracks_fk2 FOREIGN KEY (Language) REFERENCES Languages(Labguage);

ALTER TABLE Albums
ADD CONSTRAINT Albums_fk0 FOREIGN KEY (Artist) REFERENCES Artists(Name);

ALTER TABLE AlbumsSongs
ADD CONSTRAINT AlbumsS_fk0 FOREIGN KEY (AlbumID) REFERENCES Albums(AlbumID);

ALTER TABLE AlbumsSongs
ADD CONSTRAINT AlbumsS_fk1 FOREIGN KEY (TrackID) REFERENCES Tracks(TrackID);

ALTER TABLE MusicSOT
ADD CONSTRAINT MusicSOT_fk0 FOREIGN KEY (StyleID) REFERENCES MusicStyles(Style);

ALTER TABLE MusicSOT
ADD CONSTRAINT MusicSOT_fk1 FOREIGN KEY (TrackID) REFERENCES Tracks(TrackID);

```

3.3. Создание скрипта заполнения БД

Скрипт, заполняющий все таблицы БД данными:

```

connect 'C:\Users\Andrey\Desktop\BD\3\music2.fdb' user 'sysdba' password 'masterkey' ;

INSERT INTO Languages VALUES ('Russian');
INSERT INTO Languages VALUES ('Latvian');
INSERT INTO Languages VALUES ('English');

INSERT INTO MusicStyles VALUES ('Punk');
INSERT INTO MusicStyles VALUES ('Classic');
INSERT INTO MusicStyles VALUES ('Rap');

INSERT INTO Country VALUES ('Russia');
INSERT INTO Country VALUES ('Latvia');

INSERT INTO Artists VALUES ('Shnurov', null, 'Russia');
INSERT INTO Artists VALUES ('Tsoi', null, 'Russia');

INSERT INTO Albums VALUES (1, 'Tsoi', 'Blood Type');

INSERT INTO Tracks VALUES (1, 'War', 'Tsoi', 'Punk', 1988, 'Russian');
INSERT INTO AlbumsSongs VALUES (1, 1, 1);
INSERT INTO MusicSOT VALUES (1, 'Punk', 1);

```

3.4. Изменение БД по заданию преподавателя

Необходимо:

- Реализовать учет отдельных исполнителей, групп с возможностью множественного участия исполнителей в группах.

Связь **многие-ко-многим** создадим с помощью трех таблиц. Одна из таблиц будет соединительной.

```
CREATE TABLE GroupN (  
    GroupID INT NOT NULL,  
    Name varchar(30) NOT NULL,  
    PRIMARY KEY (GroupID)  
);  
  
CREATE TABLE GroupMembers (  
    ID INT NOT NULL,  
    GroupID INT NOT NULL,  
    ArtistName varchar(50) NOT NULL,  
    PRIMARY KEY (ID)  
);  
  
-- Связь многие-ко-многим состоит из двух связей один-ко-многим.  
ALTER TABLE GroupMembers  
ADD CONSTRAINT Group_fk0 FOREIGN KEY (GroupID) REFERENCES GroupN(GroupID);  
ALTER TABLE GroupMembers  
ADD CONSTRAINT Art_fk1 FOREIGN KEY (ArtistName) REFERENCES Artists (Name);  
-- Ограничение внешнего ключа запрещает изменение данных в таблице первичного ключа
```

- Реализовать учет состава групп при записи каждой композиции.

Композиция может быть записана несколькими группами и группа может быть не в полном составе.

```
CREATE TABLE GroupAccounting (  
    ID INT NOT NULL,  
    GroupMem INT NOT NULL,  
    TrackID INT NOT NULL,  
    PRIMARY KEY (TrackID)  
);  
  
ALTER TABLE GroupAccounting  
ADD CONSTRAINT Ga_fk0 FOREIGN KEY (TrackID) REFERENCES Tracks(TrackID);  
ALTER TABLE GroupAccounting  
ADD CONSTRAINT Ga_fk1 FOREIGN KEY (GroupMem) REFERENCES GroupMembers (ID);
```

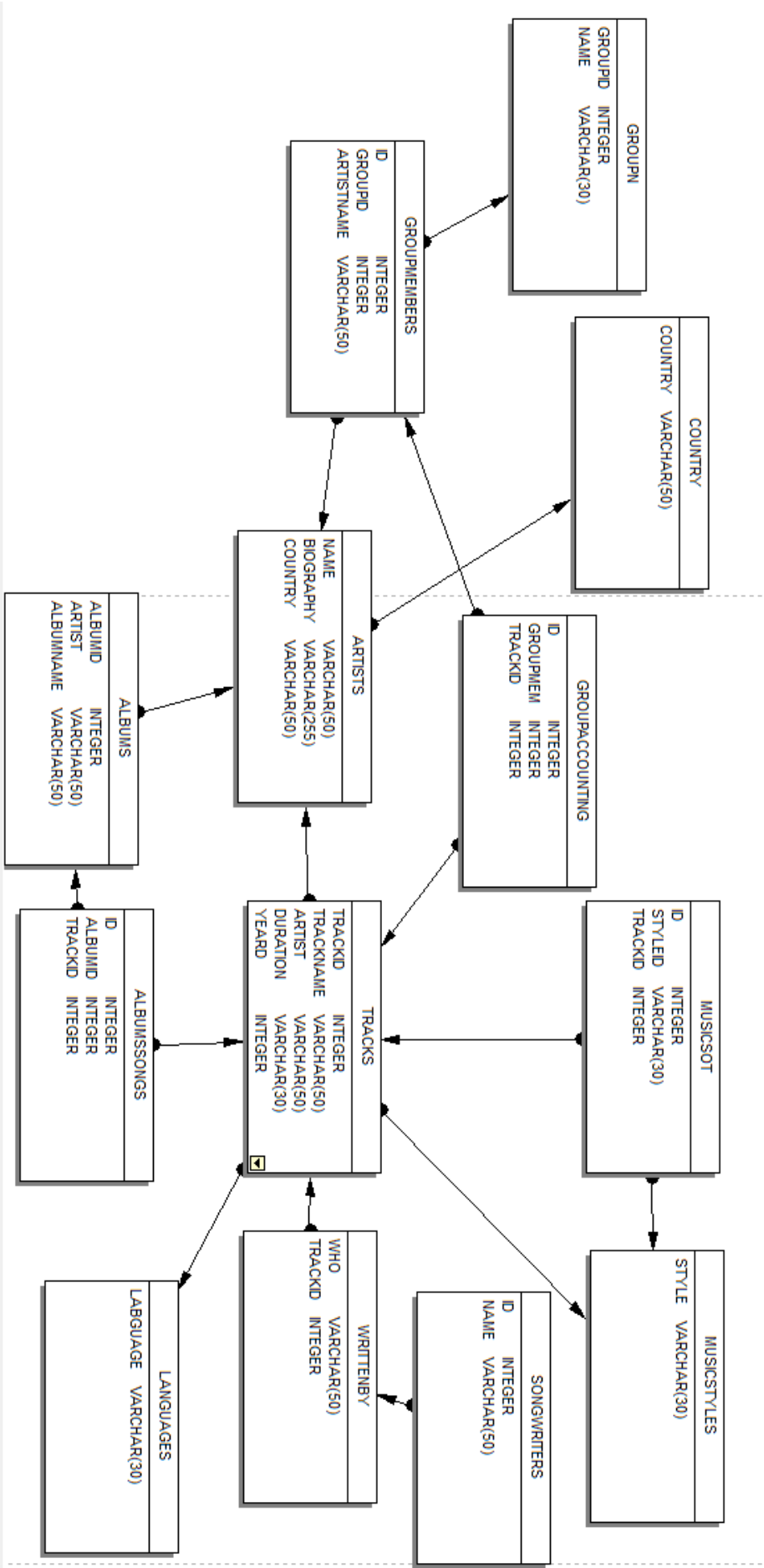
- Реализовать учет авторства композиций.

В данном случае, будем использовать связь **один-ко-многим**, так как будем считать, что у композиции может быть только один автор, при этом автор может написать несколько композиций. Из этого следует, что:

- Множество объектов из таблицы «Композиция» могут относиться к одному объекту из таблицы «Автор»
- Только один объект из таблицы «Автор» может относиться к объекту из таблицы «Композиция».

```
CREATE TABLE WrittenBy (  
    ID INT NOT NULL,  
    Who varchar(50) NOT NULL,  
    TrackID INT NOT NULL,  
    PRIMARY KEY (ID)  
);  
  
CREATE TABLE SongWriters (  
    Name varchar(50) NOT NULL,  
    PRIMARY KEY (Name)  
);  
  
ALTER TABLE WrittenBy  
ADD CONSTRAINT Wb_fk0 FOREIGN KEY (TrackID) REFERENCES Tracks(TrackID);  
ALTER TABLE WrittenBy  
ADD CONSTRAINT Sw_fk0 FOREIGN KEY (Who) REFERENCES SongWriters(Name);
```

3.5. ER-диаграмма созданной БД с помощью Database Designer



3.6. Генерация данных для БД

Для таблиц Country, SongWriters и Languages сгенерируем по 100 000 тестовых данных.

Запись №: 8296	Font size: 8	102089 records
COUNTRY		
eD^F^fKcQ)m~3z2{q,4_5}!mVg/=p)		
eHqx68o<\$#\$Qx		
eMqK<v[B>y5vWx3YlUaY\$*YpIza67%#A \\\$N\$wKB		
eQPQ)MBvInY#wwYvh<S~P5 }++]9jBHrOd		
e =R,8IT#qzb(&\$Dl8%<ogNDQO6.-E\\u"oN`LCQo0.~		
e_O488^fjweCo,epj@MT[^{6yy\$6mE^kpu~2F?Lz^tq+"		
ekc,hjb>xZ@;q		
er:lxIPLsMl9+\$`97R4fi#b>X-g		
fjM2qSKcWP%W/LHC		
f8zasoX8uG7%uf-W*ZJL		
f(yEvB+Ykm[H]a1TlwSB)O l h93%G"W<>MB5=\$(0plo(=		
f+o64F<KAm: (P		
fR{-%gK7d6AvqGCC/Ow=) PDFj,xD2r-)[_JE2<]gxa<Y0c		
fLlK3lVoI3<sA^hox(k<~WrP!j8,1@C]] lU_bw		
fa)a?:ClpFc.sbD_apiz<KS(8Fr&{W3CI		
frimLu){Nn)C}>F^a07 em~510=WWW+#3aM~L9~j}fWu(
gik^KOxEP=qCQ\$77/*N		
g\$7k(Pa;@8ITgl:		
g8~B8.Y3%8W&?YP;\\		
gb2PHU"]l84H[d=Jpm5^cqU7Hcg7U,		
g*,G8>wY1l(^AYT9t@2{(@d)NI^S9Rzs2\$8i=U2tj8SD0*		
g-{#Aev)VVK2d glv5L+y-^+O(1D6b6		
gA#_ {Lq^z_r45		
gC2lh,7lB8-a!TuOaIbz3rAC \$B		
gDq+ly+hzPvC		
1. Сетка 2. Форма 3. Печать		

Запись №: 21375	Font size: 8	100000 records fetched
LANGUAGE		
4WWE?Amjd>		
4MY4%~ at"CeXl^f5g		
4VlIrhoYl{X.6p,T;c		
4vdFT3k(RwvY36F]8uT		
4N "jehL[
4N-y@/useoTX68lQg		
4N>9(5)2;lI<?n8A3maXQza8A1'		
4N><Oq\$lb)u)lh		
4N?1[-hw;G-Dtt		
4N05)llwgU>A		
4V17)%Pwa@n		
4Vl254Gb_		
4Ve8&az=j V :lq(W//~ml+		
4V?;48@53Yf6kbWGrGmb		
4VnXl7(~6>)g		
4Vql^QRPa`+z>S6		
4Vn)C%k{^~2i}>s		
4OIFT7XgG4{YA: {dhPSLU VhIT		
4O#>yf%>D\\MJaU		
4OAY^4^13;Y+		
4OGGK, #LLo)WgF@G193(tlITI)V		
4ORl5f^okn@S%V.V?{2k&X(wW		
4OZ9x7;aQ-		
4Og8ox4^,89J~9Gu0=<~!;		
4Om1l]#roZVESvSK3@h>K0H;{KB		

Запись №: 1	Font size: 8	100628 records
NAMES		
.\$Jln<P, lB0ZcG~		
,QwW->>{i}nQF		
A+VH2^6znR?;b78D]4rLse,G3q8i6Q'dt^qW_		
Cgc^R_yz0{X}l#y.sT@Gwh.E?l		
FCI` (g)80D		
FUHxAwAaXO7l)=78X/M		
G0~0l'mvm8lJA6^kn1~		
S:TjshH35_wO;/83`S%.q;;(M9 0lI^r]UhMDj		
^Yhg0, +bb		
orfv!x+eM8k"l;l		
p+z+l0=OmS86le?lZWfM&dajempp0;Fg2^45jd1{ll6sx		
zy]D3a0o~?sImyH2<,o^6z		
!# 8ueJw8VRetl a~u<#~4v YIV`lRh2^7m^r!uE)dr1lJ		
!#y~4r.nm=wCKBO{\\WW#m\$^D3		
l,*,M[N\$,a8]dyF.		
lFeKQ47l2k`b-g%4M#pAX]FBKNU6lJb(D9		
lTfx)-\$Uzjrj&QR'0J(lp'G.Ql'r'igU,`[Oq;S8K8		
lI3Q=Eqg<		
lsnctv=s=,w		
lu:~?TJPre7h^hml6@eb.`3Bg_0#h("kh")vO(>lx9#lTM		
!{0?P;?%;1.dgERdg8T		
"s?lBTvx~@lH9YVZlS!CvDH?+D:u^F 8Z1[:8&n~L ZkC		
%89:~vx,0+Qc?]V4L?lCwJB0dk#		
]Khk)-cJ#~lZ8!o>qCf g6mSR(,lJ%2.FW9%)\$89f		
*+<[t\$]5~wZY8R~nruDX)JNZw+VlDU7VKcBuImxI~3c]		

4. Выводы

Язык DDL служит для создания и модификации структуры БД, т.е. для создания/изменения/удаления таблиц и связей. С помощью скриптов была создана база данных музыки, которая в последствии была модифицирована по заданию преподавателя. Следует понимать, что, если в базе данных уже хранятся какие-то записи, необходимо с осторожностью подходить к ее модификации.

При заполнении базы данных в среде IVExpert, мы убедились в одном из основных свойств связей, а именно что ограничение внешнего ключа запрещает изменение данных в таблице первичного ключа. В случае неверного указания связей, может возникнуть ошибка вставки данных.