Санкт-Петербургский политехнический университет Петра Великого»

**Институт компьютерных наук и технологий**

**Кафедра Компьютерных систем и программных технологий**

Лабораторная работа №2

по дисциплине "Информационное управление систем управления"

# «SQL-DDL»

Выполнил студент гр. № 43501/1 \_\_\_\_\_\_\_\_\_\_\_ Дао Суан Хоа

Преподаватель \_\_\_\_\_\_\_\_\_\_\_ Мяснов А.В.

Санкт-Петербург

2015

**Задание:**

### Спортивный клуб

Учет клиентов, посещений занятий, спортивных площадок, залов, тренажеров, упражнений. Учет времени тренеров, видов, проводимых ими занятий. Ведение учета стоимости каждого вида занятий. Учет скидок .

1. **Цель работы:**

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

1. **Программа работы**
2. Создание скрипта БД в соответствии с согласованной схемой (должны присутствовать первичные и внешние ключи, ограничения на диапазоны значений). Продемонстрировать скрипт преподавателю.
3. Создайте скрипт, заполняющий все таблицы БД данными
4. Изучите основные возможности IBExpert. Получите ER-диаграмму созданной БД с помощью **Database Designer**.
5. Автоматически сгенерируйте данные при помощи IBExpert (для трех или большего числа таблиц, не менее 100000 записей в каждой из выбранных таблиц)
6. **Выполнение программы работы**

Скрипт создания БД в соответствии с согласованной схемой приведен в листинге 1. Созданный скрипт проверен на работоспособность и продемонстрирован преподавателю.

SQL> connect 'D:/xuanhoa.fdb' user 'SYSDBA' password 'masterkey';

Database: 'D:/xuanhoa.fdb', User: SYSDBA

* CREATE TABLE COACH(

ID\_COACH INTEGER NOT NULL,

NAME\_COACH VARCHAR(30),

ADD\_COACH VARCHAR(30),

PHONE\_COACH INTEGER,

SPORT\_ID\_COACH INTEGER NOT NULL);

* CREATE TABLE SPORT(

ID\_SPORT INTEGER NOT NULL,

TYPE\_SPORT VARCHAR(30),

TRAINER VARCHAR(30));

* CREATE TABLE CLASS(

ID\_CLASS INTEGER,

NAME\_CLASS VARCHAR(30),

ROOM\_CLASS INTEGER,

SPORT\_ID\_CLASS INTEGER);

* CREATE TABLE SALE(

ID\_SALE INTEGER NOT NULL,

NAME\_SALE VARCHAR(30),

TIME\_SALE DATE,

INF\_SALE VARCHAR(30),

SPORT\_ID\_SALE INTEGER);

* CREATE TABLE STADIUM(

ID\_STADIUM INTEGER NOT NULL,

ADD\_STADIUM VARCHAR(30),

NAME\_STADIUM VARCHAR(30),

SIZE\_STADIUM INTEGER,

SPORT\_ID\_STADIUM INTEGER NOT NULL);

* CREATE TABLE CLIENTS(

ID\_CLIENT INTEGER NOT NULL,

NAME\_CLIENT VARCHAR(30),

ADD\_CLIENT VARCHAR(30),

PHONE\_CLIENT INTEGER,

NUMBER\_PLAY INTEGER,

TICKET\_ID\_CLIENTS INTEGER);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*PRIMARY KEYES\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

ALTER TABLE SPORT ADD CONSTRAINT PK\_SPORT PRIMARY KEY (ID\_SPORT);

ALTER TABLE COACH ADD CONSTRAINT PK\_COACH PRIMARY KEY (ID\_COACH);

ALTER TABLE CLASS ADD CONSTRAINT PK\_CLASS PRIMARY KEY (ID\_CLASS);

ALTER TABLE SALE ADD CONSTRAINT PK\_SALE PRIMARY KEY (ID\_SALE);

ALTER TABLE STADIUM ADD CONSTRAINT PK\_STADIUM PRIMARY KEY (ID\_STADIUM);

ALTER TABLE CLIENTS ADD CONSTRAINT PK\_CLIENTS PRIMARY KEY (ID\_CLIENT);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*FOREIGN KEYES\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

ALTER TABLE COACH ADD CONSTRAINT FK\_SPORT\_ID\_COACH FOREIGN KEY (SPORT\_ID\_COACH) REFERENCES SPORT (ID\_SPORT);

ALTER TABLE CLASS ADD CONSTRAINT FK\_SPORT\_ID\_CLASS FOREIGN KEY (SPORT\_ID\_CLASS) REFERENCES SPORT (ID\_SPORT);

ALTER TABLE SALE ADD CONSTRAINT FK\_SPORT\_\_ID\_SALE FOREIGN KEY (SPORT\_ID\_SALE) REFERENCES SPORT (ID\_SPORT);

ALTER TABLE STADIUM ADD CONSTRAINT FK\_SPORT\_ID\_STADIUM FOREIGN KEY (SPORT\_ID\_STADIUM) REFERENCES SPORT (ID\_SPORT);

ALTER TABLE CLIENTS ADD CONSTRAINT FK\_SPORT\_ID\_CLIENTS FOREIGN KEY (SPORT\_ID\_CLIENTS) REFERENCES SPORT (ID\_SPORT);

/\*\*\*\*\*\*\*\*\*\*\*\*constraint\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

ALTER TABLE COACH ADD CONSTRAINT C\_ID\_COACH CHECK (ID\_COACH > 0);

ALTER TABLE SPORT ADD CONSTRAINT CHK\_ID\_SPORT CHECK (ID\_SPORT > 0 );

ALTER TABLE TYPE\_TICKET ADD CONSTRAINT CHK\_ID\_TYPE\_TICKET CHECK (ID\_TYPE\_TICKET > 0);

ALTER TABLE TYPE\_TICKET ADD CONSTRAINT CHK\_PRICE CHECK (PRICE > 0);

ALTER TABLE EXTRA\_SEV ADD CONSTRAINT CHK\_ID\_EXTRA CHECK (ID\_EXTRA > 0);

ALTER TABLE CLASS ADD CONSTRAINT CHK\_ID\_CLASS CHECK (ID\_CLASS > 0);

ALTER TABLE CLASS ADD CONSTRAINT CHK\_ROOM\_CLASS CHECK (ROOM\_CLASS > 0);

ALTER TABLE SALE ADD CONSTRAINT CHK\_ID\_SALE CHECK (ID\_SALE > 0);

ALTER TABLE STADIUM ADD CONSTRAINT CHK\_ID\_STADIUM CHECK (ID\_STADIUM > 0);

ALTER TABLE CLIENTS ADD CONSTRAINT CHK\_ID\_CLIENT CHECK (ID\_CLIENT > 0);

ALTER TABLE CLIENTS ADD CONSTRAINT CHK\_NUMBER\_PLAY CHECK (NUMBER\_PLAY > 0);

ALTER TABLE SEASON\_TICKET ADD CONSTRAINT CHK\_ID\_SEASON\_TICKET CHECK (ID\_SEASON\_TICKET > 0);

Первичный ключ - PRIMARY KEY.

Внешний ключ - FOREIGN KEY.

**4 . Скрипт, заполняющий некоторые таблицы БД осмысленными данными приведен в листинге 2.**

|  |
| --- |
| INSERT INTO CLIENTS (ID\_SPORT, TYPE\_SPORT,TRAINER) VALUES (1,’FOOTBALL’,’BALL’);  INSERT INTO CLIENTS (ID\_SPORT, TYPE\_SPORT,TRAINER) VALUES (2,’BADMINTON’,’RACKET’);  INSERT INTO CLIENTS (ID\_SPORT, TYPE\_SPORT,TRAINER) VALUES (3,’VOLLEYBALL’,’BALL’);  INSERT INTO CLIENTS (ID\_SPORT, TYPE\_SPORT,TRAINER) VALUES (4,’TENNIS’,’BALL AND RACKET’); |

|  |
| --- |
| INSERT INTO CLIENTS (ID\_CLIENT,NAME\_CLIENT,ADD\_CLIENT,PHONE\_CLIENT,NUMBER\_PLAY,SPORT\_ID\_CLIENTS) VALUES (1,'DAOHOA','GRAZDANKYA', 235345,7,1);  INSERT INTO CLIENTS (ID\_CLIENT,NAME\_CLIENT,ADD\_CLIENT,PHONE\_CLIENT,NUMBER\_PLAY, SPORT\_ID\_CLIENTS) VALUES (2,‘ROONEY’,'GRAZDANKYA',123456,6,1);  INSERT INTO CLIENTS (ID\_CLIENT,NAME\_CLIENT,ADD\_CLIENT,PHONE\_CLIENT,NUMBER\_PLAY, SPORT\_ID\_CLIENTS) VALUES (3,'NEYMA','GRAZDANKYA',226262,11,1);  INSERT INTO CLIENTS (ID\_CLIENT,NAME\_CLIENT,ADD\_CLIENT,PHONE\_CLIENT,NUMBER\_PLAY, SPORT\_ID\_CLIENTS) VALUES (4,'MESSI','GRAZDANKYA',533312,9,1);  INSERT INTO CLIENTS (ID\_CLIENT,NAME\_CLIENT,ADD\_CLIENT,PHONE\_CLIENT,NUMBER\_PLAY, SPORT\_ID\_CLIENTS) VALUES (5,'TIGER\_WOOD','GRAZDANKYA',132454,5,2);  INSERT INTO CLIENTS (ID\_CLIENT,NAME\_CLIENT,ADD\_CLIENT,PHONE\_CLIENT,NUMBER\_PLAY, SPORT\_ID\_CLIENTS) VALUES (6,'NADAL','GRAZDANKYA',343543,,4);  INSERT INTO CLIENTS (ID\_CLIENT,NAME\_CLIENT,ADD\_CLIENT,PHONE\_CLIENT,NUMBER\_PLAY, SPORT\_ID\_CLIENTS) VALUES (7,'FEREDER','GRAZDANKYA',544545,44,4);  INSERT INTO CLIENTS (ID\_CLIENT,NAME\_CLIENT,ADD\_CLIENT,PHONE\_CLIENT,NUMBER\_PLAY, SPORT\_ID\_CLIENTS) VALUES (8,'MATTA','GRAZDANKYA',098752,33,3);  INSERT INTO CLIENTS (ID\_CLIENT,NAME\_CLIENT,ADD\_CLIENT,PHONE\_CLIENT,NUMBER\_PLAY, SPORT\_ID\_CLIENTS) VALUES (9,'TIENMINH','GRAZDANKYA',0523231,33,2);  INSERT INTO CLIENTS (ID\_CLIENT,NAME\_CLIENT,ADD\_CLIENT,PHONE\_CLIENT,NUMBER\_PLAY, SPORT\_ID\_CLIENTS) VALUES (10,'DANTRUONG','GRAZDANKYA',05554431,33,4);  COMMIT; |

|  |
| --- |
| SQL> INSERT INTO COACH (ID\_COACH,NAME\_COACH,ADD\_COACH,INF\_COACH,PHONE\_COACH,SPOR  T\_ID\_COACH) VALUES (1,'SIR\_ALEX','GRAZDANKYA','FROM ENGLAND',890567,1);  SQL> INSERT INTO COACH (ID\_COACH,NAME\_COACH,ADD\_COACH,INF\_COACH,PHONE\_COACH,SPOR  T\_ID\_COACH) VALUES (2,'BAUQUYET','GRAZDANKYA','FROM VIETNAM',894554,2);  SQL> INSERT INTO COACH (ID\_COACH,NAME\_COACH,ADD\_COACH,INF\_COACH,PHONE\_COACH,SPOR  T\_ID\_COACH) VALUES (3,'BAUMINH','GRAZDANKYA','FROM VIETNAM',235654,3);  SQL> INSERT INTO COACH (ID\_COACH,NAME\_COACH,ADD\_COACH,INF\_COACH,PHONE\_COACH,SPOR  T\_ID\_COACH) VALUES (4,'BAUTUAN','GRAZDANKYA','FROM VIETNAM',246561,4); |

|  |
| --- |
| SQL> INSERT INTO STADIUM (ID\_STADIUM,ADD\_STADIUM,NAME\_STADIUM,SIZE\_STADIUM,SPORT  \_ID\_STADIUM) VALUES (1,'HAYKA','HANGDAY',10050,1);  SQL> INSERT INTO STADIUM (ID\_STADIUM,ADD\_STADIUM,NAME\_STADIUM,SIZE\_STADIUM,SPORT  \_ID\_STADIUM) VALUES (2,'THUDUC','CUCHI',2030,2);  SQL> INSERT INTO STADIUM (ID\_STADIUM,ADD\_STADIUM,NAME\_STADIUM,SIZE\_STADIUM,SPORT  \_ID\_STADIUM) VALUES (3,'HLONG','THIENTRUONG',2030,3);  SQL> INSERT INTO STADIUM (ID\_STADIUM,ADD\_STADIUM,NAME\_STADIUM,SIZE\_STADIUM,SPORT  \_ID\_STADIUM) VALUES (4,'KK','NGHIAHUNG',1010,4); |

|  |
| --- |
| SQL> INSERT INTO CLASS (ID\_CLASS,NAME\_CLASS,ROOM\_CLASS,SPORT\_ID\_CLASS) VALUES (1,'FOOTBALL','AUDITORIA11',1);  SQL> INSERT INTO CLASS (ID\_CLASS,NAME\_CLASS,ROOM\_CLASS,SPORT\_ID\_CLASS) VALUES (2,'FOOTBALL1','AUDITORIA23',1);  SQL> INSERT INTO CLASS (ID\_CLASS,NAME\_CLASS,ROOM\_CLASS,SPORT\_ID\_CLASS) VALUES (3,'VOLLEYBALL','AUDITORIA34',3);  SQL> INSERT INTO CLASS (ID\_CLASS,NAME\_CLASS,ROOM\_CLASS,SPORT\_ID\_CLASS) VALUES (4,'BADMINTON','AUDITORIA534',2);  SQL> INSERT INTO CLASS (ID\_CLASS,NAME\_CLASS,ROOM\_CLASS,SPORT\_ID\_CLASS) VALUES (5,'TENNIS','AUDITORIA578',4); |

|  |
| --- |
| SQL> INSERT INTO SALE (ID\_SALE,NAME\_SALE,TIME\_SALE,INF\_SALE,SPORT\_ID\_SALE1) VALUES  (1,'CAREER','11.02.2015','FOR CLIENTS HAVE WELL RESULT',21;  SQL> INSERT INTO SALE (ID\_SALE,NAME\_SALE,TIME\_SALE,INF\_SALE,SPORT\_ID\_SALE1) VALUES  (2,'NOEL','25.12.2015','FOR CLIENTS HAVE WELL RESULT',2);  SQL> INSERT INTO SALE (ID\_SALE,NAME\_SALE,TIME\_SALE,INF\_SALE,SPORT\_ID\_SALE1) VALUES  ( 3,'NATIONAL','11.03.2015','COMPETATION IN THE SCHOLL',3); |

Модифицировать схему БД для удовлетворения следующим условиям:

1. Ввести типы абонементов (на определенные виды спорта, полный и пр.)
2. Ввести дополнительные услуги к абонементам.

ALTER TABLE CLIENTS DROP SPORT\_ID\_CLIENT;

ALTER TABLE CLIENTS ADD TICKET\_ID\_CLIENTS INTEGER NOT NULL;

* CREATE TABLE TYPE\_TICKET (

ID\_TYPE\_TICKET INTEGER NOT NULL,

PRICE INTEGER,

NAME\_TICKET VARCHAR (30));

ALTER TABLE TYPE\_TICKET ADD CONSTRAINT PK\_TYPE\_TICKET PRIMARY KEY (ID\_TYPE\_TICKET);

ALTER TABLE TYPE\_TICKET ADD CONSTRAINT FK\_TYPE\_TICKET FOREIGN KEY (ID\_TYPE\_TICKET) REFERENCES SPORT (ID\_SPORT);

ALTER TABLE TYPE\_TICKET ADD CONSTRAINT CHK\_ID\_TYPE\_TICKET CHECK (ID\_TYPE\_TICKET > 0);

ALTER TABLE TYPE\_TICKET ADD CONSTRAINT CHK\_PRICE CHECK (PRICE > 0);

* CREATE TABLE SEASON\_TICKET(

ID\_SEASON\_TICKET INTEGER NOT NULL,

SPORT\_ID\_SEASON INTEGER NOT NULL);

ALTER TABLE SEASON\_TICKET ADD CONSTRAINT PK\_SEASON\_TICKET PRIMARY KEY (ID\_SEASON\_TICKET);

ALTER TABLE SEASON\_TICKET ADD CONSTRAINT FK\_SEASON\_TICKET FOREIGN KEY (SPORT\_ID\_SEASON) REFERENCES

TYPE\_TICKET (ID\_TYPE\_TICKET);

ALTER TABLE SEASON\_TICKET ADD CONSTRAINT CHK\_ID\_SEASON\_TICKET CHECK (ID\_SEASON\_TICKET > 0);

* CREATE TABLE EXTRA\_SEV(

ID\_EXTRA INTEGER NOT NULL,

NAME\_EXTRA VARCHAR(30),

ID\_TICKET INTEGER);

ALTER TABLE EXTRA\_SEV ADD CONSTRAINT PK\_EXTRA\_SEV PRIMARY KEY (ID\_EXTRA);

ALTER TABLE EXTRA\_SEV ADD CONSTRAINT FK\_ID\_TICKET FOREIGN KEY (ID\_TICKET) REFERENCES

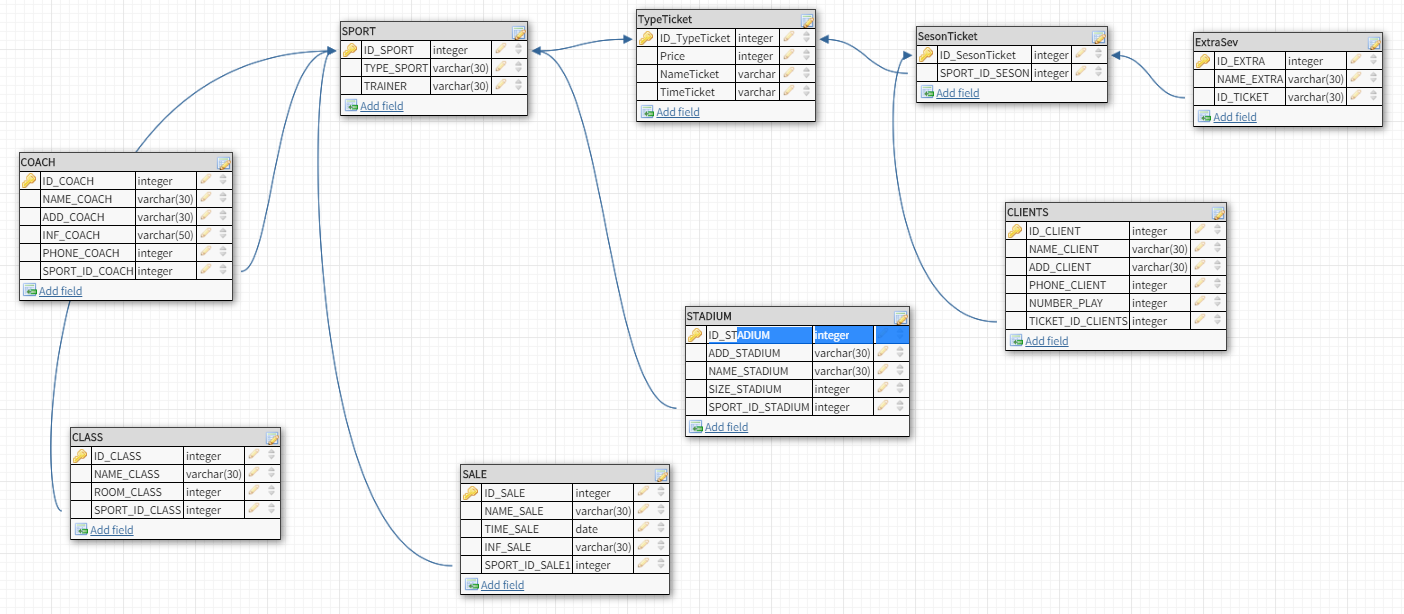
SEASON\_TICKET (ID\_SEASON\_TICKET);

ALTER TABLE EXTRA\_SEV ADD CONSTRAINT CHK\_ID\_EXTRA CHECK (ID\_EXTRA > 0);

ALTER TABLE CLIENTS ADD CONSTRAINT FK\_TICKET\_ID\_CLIENTS FOREIGN KEY (TICKET\_ID\_CLIENTS) REFERENCES

SEASON\_TICKET (ID\_SEASON\_TICKET);

Структура БД после модификации:



С помощью Database Designer в IBExpert была сгенерирована ER-диаграмма нашей БД:

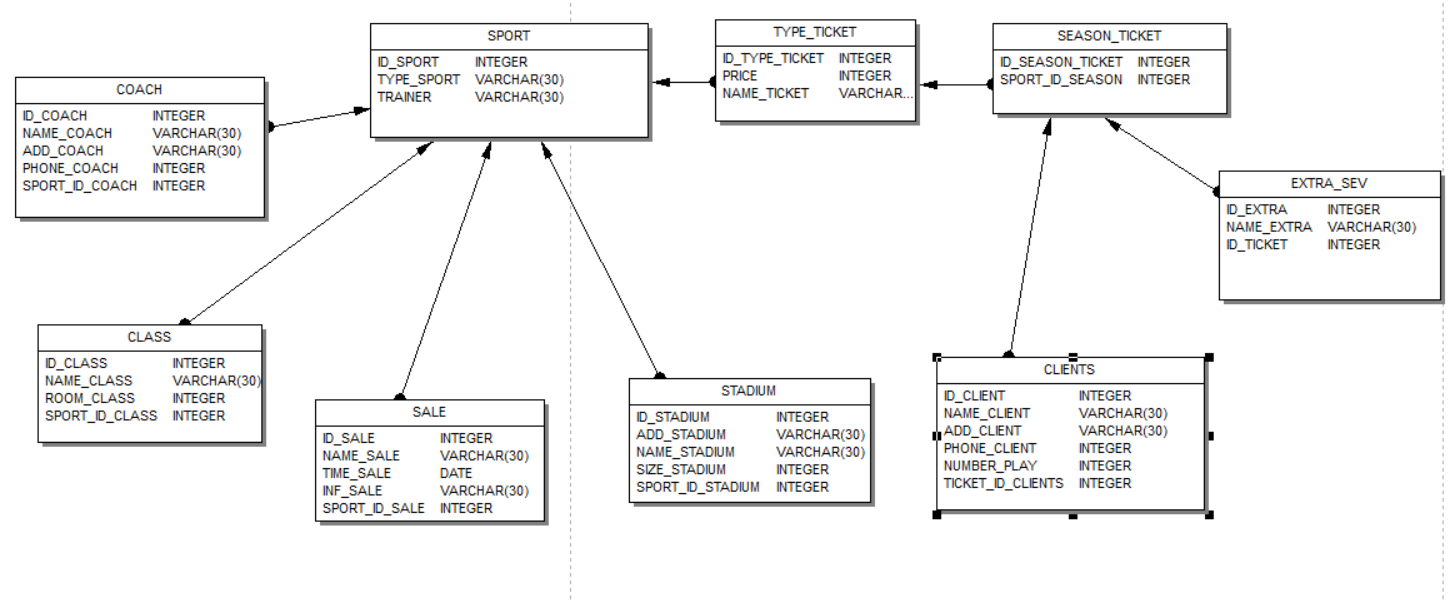


Рис 1: ER-диаграмму созданной БД с помощью **Database Designer**.

1. **.Вывод**

В результате работы было проведено знакомство и изучение языка SQL-DDL. Также были написаны скрипты для создания таблиц в БД, для заполнения их данными и для модификации БД по заданию преподавателя.

SQL-DDL (Data Definition Language) - язык определения структур и ограничений целостности баз данных. Сюда относятся команды создания и удаления баз данных; создания, изменения и удаления таблиц; управления пользователями и т.д.

Использовались такие инструменты языка DDL как <create> (создать), <alter> (изменить), <drop> (удалить). После выполнения данных команд автоматически выполняется команда <commit>, т.е. выполненные изменения уже нельзя отменить, поэтому следует быть предельно внимательным при работе с данным языком. Также можно отметить что тексты SQL-запросов, написанные на SQL DDL могут быть достаточно легко перенесены из одной СУБД в другую.

С помощью IBExpert была сгенерирована ER-диаграмма нашей БД, было произведено заполнение таблиц случайными данными.