

Санкт-Петербургский Политехнический Университет Петра Великого
Институт компьютерных наук и технологий
Кафедра компьютерных систем и программных технологий

Отчет по лабораторной работе

Дисциплина: «Базы данных»

Тема: «Язык SQL-DDL»

Работу выполнила:

Мартюшева Надежда

Группа: 43501/3

Преподаватель:

Мяснов Александр Владимирович

Санкт-Петербург
2017

1 Цели работы

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

2 Программа работы

1. Самостоятельное изучение SQL-DDL.
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. Выполнение SQL-запросов, изменяющих схему созданной БД по заданию преподавателя. Продемонстрировать их работу преподавателю.
5. Изучите основные возможности IVExpert. Получите ER-диаграмму созданной БД с помощью Database Designer.
6. Автоматически сгенерируйте данные при помощи IVExpert (для трех или большего числа таблиц, не менее 100000 записей в каждой из выбранных таблиц).

3 Ход работы

В соответствие со схемой БД, разработанной в предыдущей работе (рисунок 1) был создан SQL-скрипт, создающий базу данных.

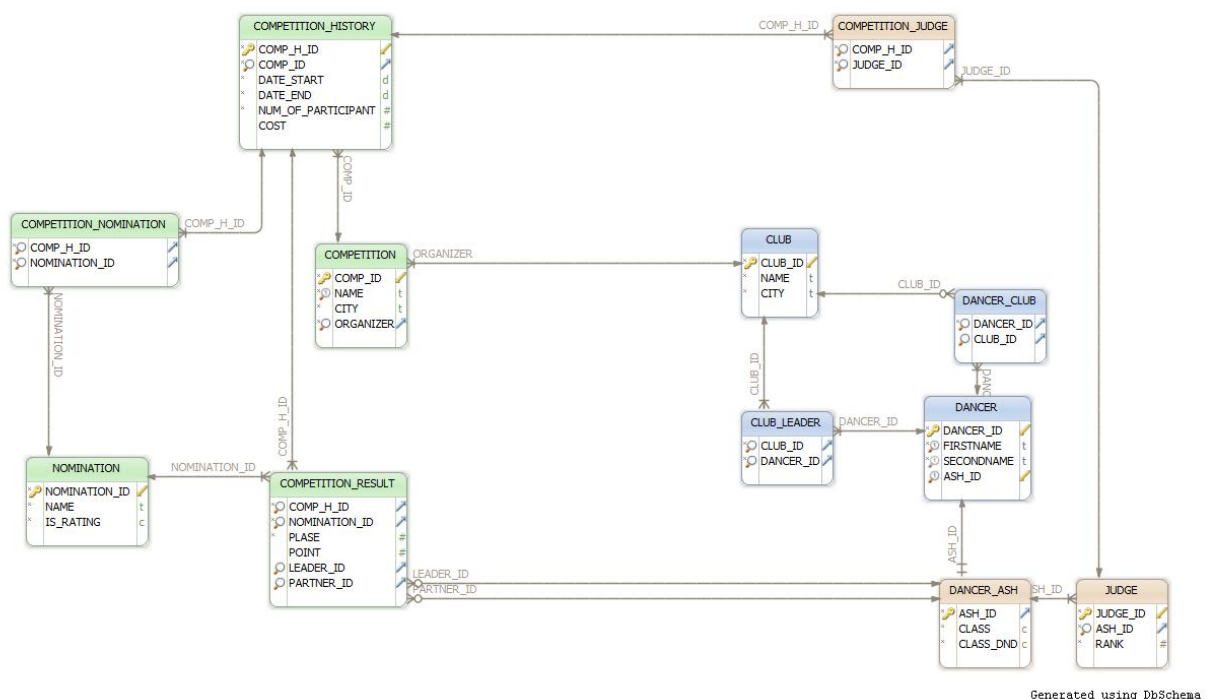


Рис. 1: ER-диаграмма разрабатываемой базы данных

Сначала создадим базу данных и подключимся к ней.

```
1 create database 'HUSTL.FBD' user 'SYSDBA' password 'masterkey';
2
3 connect 'HUSTL.FBD' user 'SYSDBA' password 'masterkey';
```

Листинг 1: Создание базы данных

Создадим несколько пользовательских типов данных при помощи доменов.

```
1 create domain dom_class as char(1)
2   default 'E'
3   not null
4   check (value in ('A', 'B', 'C', 'D', 'E'));
5
6 create domain dom_class_DnD as char(2)
7   default 'Bg'
8   not null
9   check (value in ('Bg', 'RS', 'M', 'S', 'Ch'));
10
11 create domain dom_rank as int
12   default 3
13   not null
14   check (value > 0 and value < 4);
15
16 create domain dom_rating as char(3)
17   default 'yes'
18   not null
19   check (value in ('yes', 'no'));
20
21 commit;
```

Листинг 2: Создание доменов

Теперь создадим таблицы базы данных в соответствии со схемой.

```
1 create table dancer (
2   dancer_id int primary key,
3   firstname varchar(25) not null,
4   secondname varchar(25) not null,
5   ASH_id int unique,
6   constraint name_unique unique (firstname, secondname));
7
8 create table dancer_ASH (
9   ASH_id int primary key references dancer(ASH_id),
10  class dom_class,
11  class_DnD dom_class_DnD );
12
13 create table club (
14   club_id int primary key,
15   name varchar(10) not null,
16   city varchar(10) not null);
17
18 create table club_leader (
19   club_id int not null references club(club_id),
20   dancer_id int not null references dancer(dancer_id));
21
22 create table dancer_club (
23   dancer_id int not null references dancer(dancer_id),
24   club_id int references club(club_id)
25   constraint dc_unique unique (dancer_id, club_id));
26
27 create table judge (
28   judge_id int primary key,
29   ASH_id int not null references dancer_ASH(ASH_id),
30   rank dom_rank,
31   data_valid date not null);
32
33 create table competition (
34   comp_id int primary key,
35   name varchar(20) not null unique,
36   city varchar(10) not null,
37   organizer int not null references club(club_id));
38
39 create table competition_history (
40   comp_h_id int primary key;
41   comp_id int not null references competition(comp_id),
42   date_start date not null,
43   date_end date not null,
44   num_of_participant int not null check (num_of_participant > 0),
45   cost decimal (6, 2));
46
47 create table nomination (
48   nomination_id int primary key,
49   name varchar(10) not null,
```

```

50 | is_rating dom_rating);
51 |
52 | create table competition_result (
53 |   comp_h_id int      not null references competition_history(comp_h_id),
54 |   nomination_id int  not null references nomination(nomination_id),
55 |   plase    int      not null,
56 |   point    int,
57 |   leader_id int      references dancer_ASH(ASH_id),
58 |   partner_id int     references dancer_ASH(ASH_id));
59 |
60 | create table competition_judge(
61 |   comp_h_id int      not null references competition_history(comp_h_id),
62 |   judge_id int       not null references judge(judge_id));
63 |
64 | create table competition_nomination(
65 |   comp_h_id int      not null references competition_history(comp_h_id),
66 |   nomination_id int  not null references nomination(nomination_id));
67 |
68 | commit;

```

Листинг 3: Создание таблиц базы данных

при помощи IVExpert Database Designer сгенерируем SQL-диаграмму созданной базы данных (рисунки 2).

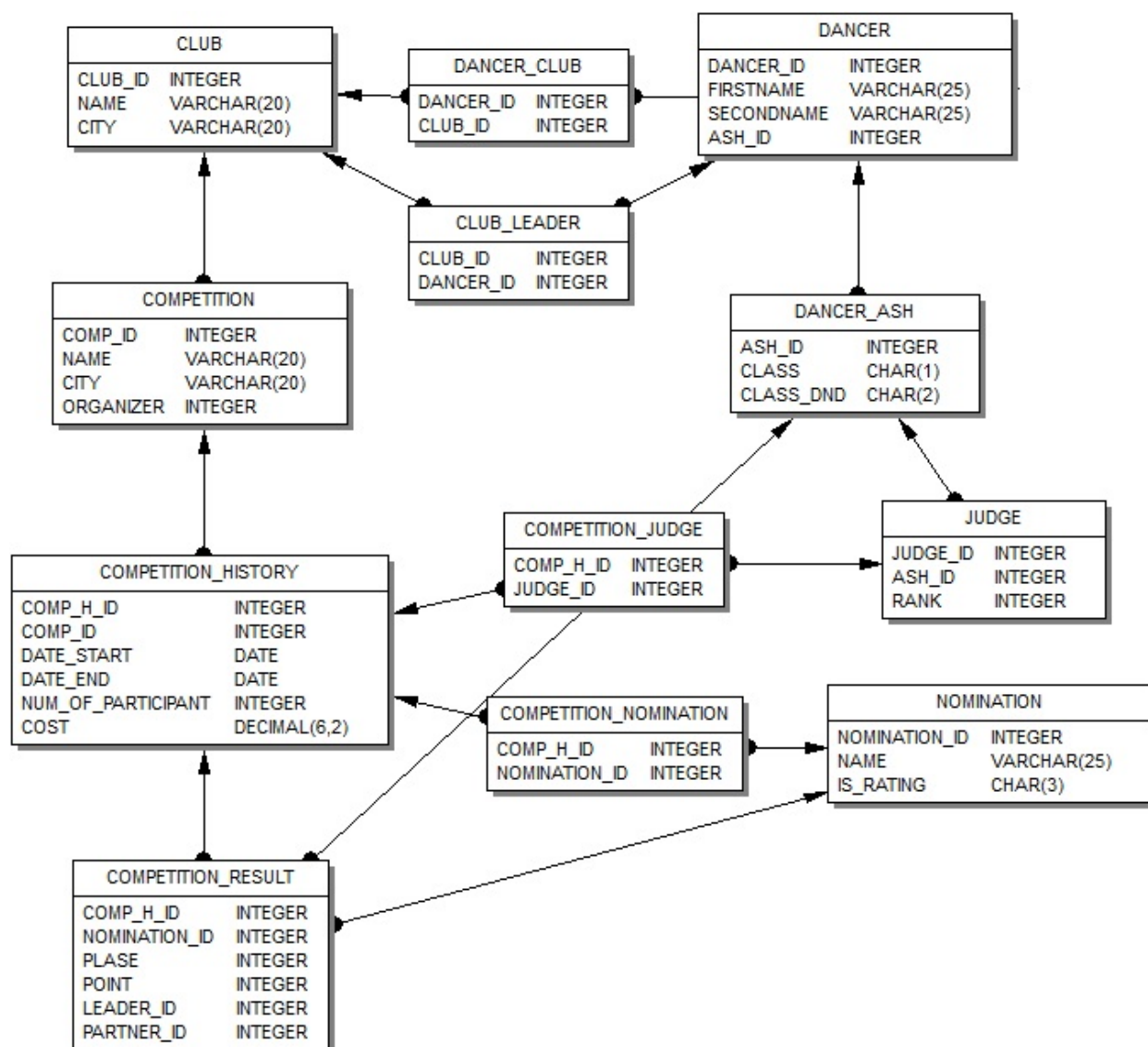


Рис. 2: SQL-диаграмма изначальной базы данных

Внесем изменения в БД в соответствие с индивидуальным заданием. Задание заключается в сле-

дующем:

1. Реализовать учет спонсорской помощи клубам и мероприятиям;
2. Реализовать учет продаж билетов на мероприятия.

Учет спонсорской помощи клубам и мероприятиям реализуем, за счет добавления следующих таблиц:

- SPONSOR – информация о спонсорах: название/имя, город, тип;
- SPONSOR CLUB реализует связь многие-ко-многим между таблицами SPONSOR и CLUB, содержит информацию о спонсорах каждого клуба ;
- SPONSOR_COMPETITION реализует связь многие-ко-многим между таблицами SPONSOR и COMPETITION_HISTORY, содержит информацию о спонсорах каждого соревнования;
- SUBSIDY_FOR_CLUB – информация о дотациях клубам: размер и дата;
- SUBSIDY_FOR_COMPETITION – информация о дотациях соревнованиям: размер, дата, тип дотации.

```
1 create domain sponsor_type as
2   varchar(4) default 'shop'
3   check (value in ('shop', 'club', 'mgzn', 'oth'));
4
5 create table sponsor (
6   sponsor_id int primary key,
7   name varchar(30) not null,
8   city varchar(25) not null,
9   type_sp sponsor_type );
10
11 create table sponsor_club(
12   subsidy_id int primary key,
13   sponsor_id int not null references sponsor(sponsor_id),
14   club_id int not null references club(club_id));
15
16 create table subsidy_for_club(
17   subsidy_id int references sponsor_club(subsidy_id),
18   subsidy_h_id int primary key,
19   sub_date date not null,
20   subsidy decimal(6, 2) not null;
21
22 create table sponsor_competition(
23   subsidy_id int primary key,
24   sponsor_id int not null references sponsor(sponsor_id),
25   comp_h_id int not null references competition_history(comp_h_id));
26
27 create table subsidy_for_competition(
28   subsidy_id int references sponsor_competition(subsidy_id),
29   subsidy_h_id int primary key,
30   sub_date date not null,
31   subsidy decimal(6, 2) not null;
32
33 commit;
```

Листинг 4: Создание таблиц для учета спонсорской помощи клубам и мероприятиям

Учет продаж билетов реализуем за счет изменения таблицы COMPETITION_HISTORY.

```
1 alter table competition_history
2 alter column cost to cost_of_participation ,
3 add num_of_guest int ,
4 add cost_of_entrance int ;
5
6 commit;
```

Листинг 5: Изменение таблицы COMPETITION_HISTORY для учета продаж входных билетов на соревнования

SQL-диаграмма базы данных после внесения изменений представлена на рисунке 3.

Заполним базу данных небольшим количеством осмысленных данных.

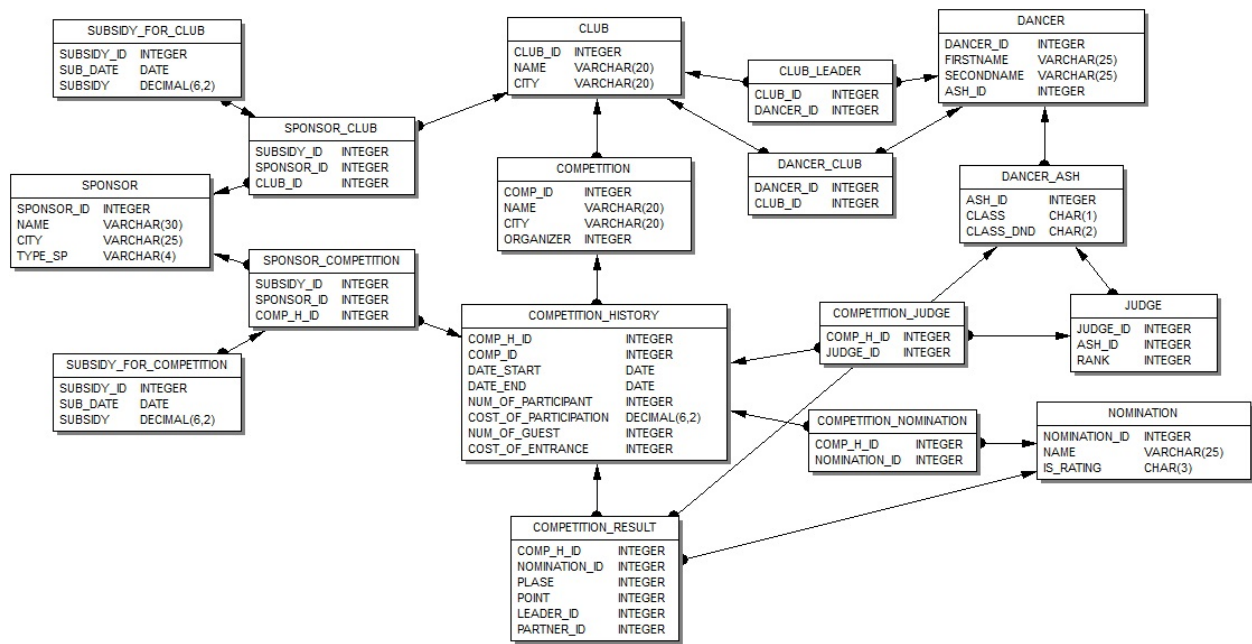


Рис. 3: SQL-диаграмма базы данных после внесения изменений

```

1  /*Клубы*/
2  insert into club values
3  (142, 'Мегаполис', 'Москва');
4
5  insert into club values
6  (143, 'Мегаполис', 'СанктПетербург-');
7
8  insert into club values
9  (144, 'Dance Drive', 'СанктПетербург-');
10
11 insert into club values
12 (146, 'MMDance', 'СанктПетербург-');
13
14 insert into club values
15 (137, 'Project', 'Екатеренбург');
16
17 insert into club values
18 (174, 'AlphaDance', 'Москва');
19
20 insert into club values
21 (183, 'YouDance', 'СанктПетербург-');
22
23 insert into club values
24 (40, 'Движение', 'Москва');
25
26 insert into club values
27 (60, 'Strikly', 'СанктПетербург-');
28
29 /*Танцоры*/
30 insert into dancer values
31 (1, 'Белякова', 'Анастасия', 627);
32
33 insert into dancer values
34 (2, 'Васинева', 'Полина', 6004);
35
36 insert into dancer values
37 (3, 'Цветкова', 'Елена', 2742);
38
39 insert into dancer values
40 (4, 'Корочкин', 'Федор', 5993);
41
42 insert into dancer values
43 (5, 'Маликова', 'Мария', 3163);
44
45 insert into dancer values

```

```

46 | (6, 'Милованов', 'Александр', 502);
47 |
48 | insert into dancer values
49 | (7, 'Пузанова', 'Ирина', 1185);
50 |
51 | insert into dancer values
52 | (8, 'Сова', 'Алексей', 1161);
53 |
54 | insert into dancer values
55 | (9, 'Свителев', 'Александр', 2207);
56 |
57 | insert into dancer values
58 | (10, 'Апрелев', 'Николай', 2767);
59 |
60 | insert into dancer values
61 | (11, 'Мартюшева', 'Надежда', 8595);
62 |
63 | insert into dancer values
64 | (12, 'Вольтман', 'Александр', 9077);
65 |
66 | /*Клуб – Руководитель*/
67 | insert into club_leader values (143, 7);
68 |
69 | insert into club_leader values (144, 9);
70 |
71 | insert into club_leader values (146, 5);
72 |
73 | insert into club_leader values (137, 3);
74 |
75 | insert into club_leader values (40, 6);
76 |
77 | insert into club_leader values (60, 8);
78 |
79 | /*Танцор – Клуб*/
80 | insert into dancer_club values (7, 143);
81 |
82 | insert into dancer_club values (10, 143);
83 |
84 | insert into dancer_club values (9, 144);
85 |
86 | insert into dancer_club values (5, 146);
87 |
88 | insert into dancer_club values (3, 137);
89 |
90 | insert into dancer_club values (6, 40);
91 |
92 | insert into dancer_club values (1, 40);
93 |
94 | insert into dancer_club values (8, 60);
95 |
96 | insert into dancer_club values (2, 144);
97 |
98 | insert into dancer_club values (4, 144);
99 |
100 | insert into dancer_club values (11, 144);
101 |
102 | /*Танцор ACX*/
103 | insert into dancer_ASH values
104 | (627, 'A', 'S');
105 |
106 | insert into dancer_ASH values
107 | (6004, 'B', 'RS');
108 |
109 | insert into dancer_ASH values
110 | (2742, 'D', 'M');
111 |
112 | insert into dancer_ASH values
113 | (5993, 'B', 'RS');
114 |
115 | insert into dancer_ASH values
116 | (3163, 'A', 'Ch');
117 |
118 | insert into dancer_ASH values
119 | (502, 'A', 'S');
120 |
121 | insert into dancer_ASH values

```

```

122 | (1185, 'A', 'S');
123 |
124 | insert into dancer_ASH values
125 | (1161, 'A', 'S');
126 |
127 | insert into dancer_ASH values
128 | (2207, 'A', 'S');
129 |
130 | insert into dancer_ASH values
131 | (2767, 'A', 'Ch');
132 |
133 | insert into dancer_ASH values
134 | (8595, 'E', 'Bg');
135 |
136 | /*Судьи ACX*/
137 | insert into judge values
138 | (1, 627, 1);
139 |
140 | insert into judge values
141 | (2, 5993, 3);
142 |
143 | insert into judge values
144 | (3, 502, 1);
145 |
146 | insert into judge values
147 | (4, 1185, 1);
148 |
149 | insert into judge values
150 | (5, 1161, 3);
151 |
152 | insert into judge values
153 | (6, 2207, 1);
154 |
155 | insert into judge values
156 | (7, 2767, 3);
157 |
158 | /*Конкурсы*/
159 | insert into competition values
160 | (1, 'Starship', 'СанктПетербург-', 143);
161 |
162 | insert into competition values
163 | (2, 'Leningrad-Cup', 'СанктПетербург-', 144);
164 |
165 | insert into competition values
166 | (3, 'Кубок Белых Ночей', 'СанктПетербург-', 144);
167 |
168 | insert into competition values
169 | (4, 'Halloween-Cup', 'СанктПетербург-', 144);
170 |
171 | insert into competition values
172 | (5, 'Кубок морской славы', 'СанктПетербург-', 0);
173 |
174 | /*Номинации*/
175 | insert into nomination values
176 | (1, 'Дебют', 'no');
177 |
178 | insert into nomination values
179 | (2, 'E', 'yes');
180 |
181 | insert into nomination values
182 | (3, 'D', 'yes');
183 |
184 | insert into nomination values
185 | (7, 'C', 'yes');
186 |
187 | insert into nomination values
188 | (4, 'Bg', 'yes');
189 |
190 | insert into nomination values
191 | (5, 'RS', 'yes');
192 |
193 | insert into nomination values
194 | (6, 'Stricly Космос-', 'no');
195 |
196 | /*История конкурсов*/
197 | insert into competition_history values

```



```

198 (1, 1, '30.7.2016', '31.7.2016', 200, 200, 20, 100);
199
200 insert into competition_history values
201 (2, 2, '24.9.2016', '24.9.2016', 340, 300, 20, 100);
202
203 insert into competition_history values
204 (3, 3, '13.6.2015', '14.6.2015', 400, 300, 20, 100);
205
206 insert into competition_history values
207 (3, 4, '25.6.2016', '25.6.2016', 380, 250, 20, 100);
208
209 insert into competition_history values
210 (5, 4, '29.10.2016', '29.10.2016', 200, 300, 230, 300);
211
212 insert into competition_history values
213 (6, 5, '20.11.2016', '20.11.2016', 300, 300, 346, 300);
214
215 /*Результаты конкурсов*/
216 insert into competition_result values
217 (4, 7, 1, 1, 5993, 6004);
218
219 insert into competition_result (comp_h_id, nomination_id, plase, point, partner_id) values
220 (2, 4, 12, 1, 8595);
221
222 insert into competition_result (comp_h_id, nomination_id, plase, point, partner_id) values
223 (5, 4, 8, 2, 8595);
224
225 insert into competition_result (comp_h_id, nomination_id, plase, point, partner_id) values
226 (6, 4, 7, 2, 8595);
227
228 insert into competition_result value
229 (6, 2, 1, 3, 9077, 8595);
230
231 /*Конкурс – Судьи*/
232 insert into competition_judge values (1, 7);
233
234 insert into competition_judge values (1, 4);
235
236 insert into competition_judge values (2, 1);
237
238 insert into competition_judge values (2, 6);
239
240 /*Конкурс – Номинации*/
241 insert into competition_nomination values (1, 2);
242
243 insert into competition_nomination values (1, 3);
244
245 insert into competition_nomination values (1, 4);
246
247 insert into competition_nomination values (2, 1);
248
249 insert into competition_nomination values (2, 4);
250
251 /*Спонсоры*/
252 insert into sponsor values
253 (1, 'Dance Line', 'СанктПетербург', 'shop');
254
255 insert into sponsor values
256 (2, 'YouDance', 'Москва', 'club');
257
258 /*Спонсор – Клуб*/
259 insert into sponsor_club values
260 (1, 2, 144);
261
262 /*Спонсорская помощь клубам*/
263 insert into subsidy_for_club values
264 ('5.12.2014', 150000, 1, 1);
265
266 insert into subsidy_for_club values
267 ('15.8.2015', 240000, 1, 2);
268
269 /*Спонсор – Конкурс*/
270 insert into sponsor_competition values
271 (1, 1, 1);
272
273 insert into sponsor_competition values

```

```

274 (2, 1, 2);
275
276 insert into sponsor_competition values
277 (3, 1, 3);
278
279 /*Спонсорская помощь соревнованиям*/
280 insert into subsidy_for_competition values
281 ('30.7.2016', 25000, 1, 1);
282
283 insert into subsidy_for_competition values
284 ('24.9.2016', 5000, 2, 2);
285
286 insert into subsidy_for_competition values
287 ('13.6.2015', 13000, 3, 3);
288

```

Листинг 6: Внесение осмысленных данных в таблицу

При помощи IBExpert были автоматически сгенерированы скрипты с данными для заполнения таблиц. Для генерации первичных ключей использовалась генерация автоинкрементом, для внешних ключей - генерация из другой таблицы, для городов - генерация из списка (был задан список городов, где есть хастл и проводятся соревнования). Все имена и названия генерировались произвольно при использовании латинских букв в нижнем регистре. Для дат проведения конкурсов задавался диапазон от начала существования ассоциации спортивного хастла до сегодняшнего дня.

4 Вывод

В ходе работы я ознакомилась с языком SQL-DDL, который служит для определения структур и ограничений целостности баз данных.

При создании структур БД необходимо соблюдать определенный порядок, поскольку структуры связаны между собой. Например, чтобы ссылаться на некоторую таблицу, необходимо, чтобы эта таблица уже была создана. Также использовать домен можно лишь после того, как он был объявлен.

Скрипты писались вручную и запускались из консоли isql. Это позволило лучше разобраться в структуре языка SQL-DDL и прочувствовать особенности непосредственного взаимодействия с базой данных.