

Санкт-Петербургский Политехнический Университет Петра Великого

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

ОТЧЕТ
по лабораторной работе

«Язык SQL-DDL»

Базы данных

Работу выполнил студент

группа 3530901/60202 Курякин Д.А.

Работу принял преподаватель

Мяснов А.В.

Санкт-Петербург

2020

Содержание

1	Цель работы	3
2	Программа работы	3
3	Теоретическая информация	3
4	Выполнение работы	4
4.1	Структура базы данных	4
4.2	Скрипт создания структуры базы данных	4
4.3	Скрипт заполнения таблиц тестовыми данными	5
4.4	Структура базы данных после изменения	6
4.5	Скрипт создания структуры базы данных после изменения . . .	7
4.6	Скрипт заполнения таблиц тестовыми данными после изменения	8
4.7	Скрипт создания структуры базы данны первичный ключ serial	10
4.8	Скрипт заполнения таблиц тестовыми данными первичный ключ serial	11
5	Выводы	13

1. Цель 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. Демонстрация скрипта преподавателю.
3. Создание скрипта, заполняющего все таблицы БД данными.
4. Выполнение SQL-запросов, изменяющих схему созданной БД по заданию преподавателя. Демонстрация их работы преподавателю.

3. Теоретическая информация

Язык SQL (Structured Query Language) – язык структурированных запросов. Он позволяет формировать весьма сложные запросы к базам данных. В SQL определены два подмножества языка:

- **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.
- **SQL-DML** (Data Manipulation Language) – язык манипулирования данными: добавление, изменение, удаление и извлечение данных, управления транзакциями. Функции SQL-DML определяются первым словом в предложении (часто называемом запросом), которое является глаголом: **SELECT** («выбрать»), **INSERT** («вставить»), **UPDATE** («обновить»), и **DELETE** («удалить»).

4. Выполнение работы

4.1. Структура базы данных

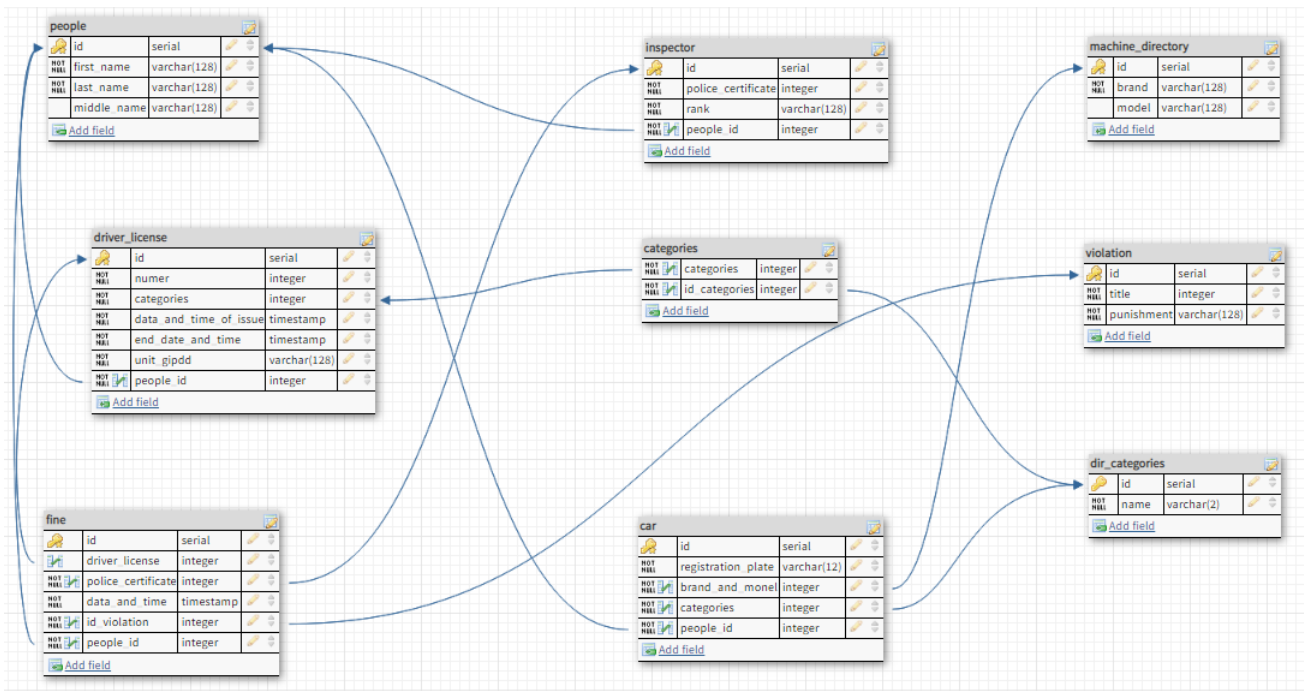


Рис. 4.1: Схема модели

4.2. Скрипт создания структуры базы данных

```
1 CREATE TABLE "people" (  
2     "id" integer PRIMARY KEY,  
3     "first_name" varchar(128) NOT NULL,  
4     "last_name" varchar(128) NOT NULL,  
5     "middle_name" varchar(128)  
6 );  
7  
8 CREATE TABLE "driver_license" (  
9     "id" integer PRIMARY KEY,  
10    "number" integer NOT NULL,  
11    "categories" integer UNIQUE,  
12    "data_and_time_of_issue" TIMESTAMP NOT NULL,  
13    "end_date_and_time" TIMESTAMP NOT NULL,  
14    "unit_gipdd" varchar(128) NOT NULL,  
15    "people_id" integer REFERENCES people (id) NOT NULL  
16 );  
17  
18 CREATE TABLE "inspector" (  
19     "id" integer PRIMARY KEY,  
20     "police_certificate" integer NOT NULL,  
21     "rank" varchar(128) NOT NULL,  
22     "people_id" integer REFERENCES people (id) NOT NULL  
23 );  
24
```

```

25 CREATE TABLE "machine_directory" (
26     "id" integer PRIMARY KEY,
27     "brand" varchar(128) NOT NULL,
28     "model" varchar(128)
29 );
30
31 CREATE TABLE "violation" (
32     "id" integer PRIMARY KEY,
33     "title" integer NOT NULL,
34     "punishment" varchar(128) NOT NULL
35 );
36
37 CREATE TABLE "dir_categories" (
38     "id" integer PRIMARY KEY,
39     "name" varchar(2) NOT NULL
40 );
41
42 CREATE TABLE "categories" (
43     "categories" integer REFERENCES driver_license (categories) NOT NULL,
44     "id_categories" integer REFERENCES dir_categories (id) NOT NULL
45 );
46
47 CREATE TABLE "car" (
48     "id" integer PRIMARY KEY,
49     "registration_plate" varchar(12) NOT NULL UNIQUE,
50     "brand_and_monel" integer NOT NULL,
51     "categories" integer REFERENCES dir_categories (id) NOT NULL,
52     "people_id" integer REFERENCES people (Id) NOT NULL
53 );
54
55 CREATE TABLE "fine" (
56     "id" integer PRIMARY KEY,
57     "driver_license" integer REFERENCES driver_license (Id) UNIQUE,
58     "police_certificate" integer REFERENCES inspector (Id) NOT NULL,
59     "data_and_time" TIMESTAMP NOT NULL,
60     "id_violation" integer NOT NULL,
61     "people_id" integer REFERENCES people (Id) NOT NULL
62 );

```

4.3. Скрипт заполнения таблиц тестовыми данными

```

1 INSERT INTO people ( id, first_name, last_name, middle_name)
2 VALUES (1, 'Александр', 'Сидоров', 'Александрович'),
3 (2, 'Иван', 'Иванов', 'Иванович'),
4 (3, 'Петров', 'Григорий', 'Антонович');
5
6 INSERT INTO driver_license ( id, numer, categories, data_and_time_of_issue,
7                             end_date_and_time, unit_gipdd, people_id)
8 VALUES (1, 782212, 1, '2010-05-01', '2020-05-01', 'ГИБДД 7816', 1),
9 (2, 782245, 2, '2012-08-01', '2022-08-01', 'ГИБДД 7816', 2);
10
11 INSERT INTO inspector ( id, police_certificate, rank, people_id)
12 VALUES (1, 12345, 'Лейтенант', 1),
13 (2, 45678, 'Лейтенант', 2);
14
15 INSERT INTO machine_directory ( id, brand, model)
16 VALUES (1, 'LADA', 'Vesta'),
17 (2, 'LADA', 'Xray'),

```

```

18 (3, 'Reno', 'Lagan');
19
20 INSERT INTO violation ( id, title, punishment)
21 VALUES (1, 1000, 'Превышение скорости'),
22 (2, 500, 'Неправильная парковка');
23
24 INSERT INTO dir_categories ( id, name)
25 VALUES (1, 'A'),
26 (2, 'B'),
27 (3, 'C'),
28 (4, 'D');
29
30 INSERT INTO categories ( categories, id_categories)
31 VALUES (1, 1),
32 (1, 1);
33
34 INSERT INTO car ( id, registration_plate, brand_and_monel, categories, people_id)
35 VALUES (1, 'B123KX178', 1, 2, 1),
36 (2, 'A777AA64', 2, 2, 2);
37
38 INSERT INTO fine ( id, driver_license, police_certificate, data_and_time, id_violation,
39                   people_id)
39 VALUES (1, 1, 2, '2018-12-10', 1, 1);

```

После внесения дополнительных требований преподавателя была изменена структура базы данных, добавлены таблицы и добавлены тестовые данные.

4.4. Структура базы данных после изменения

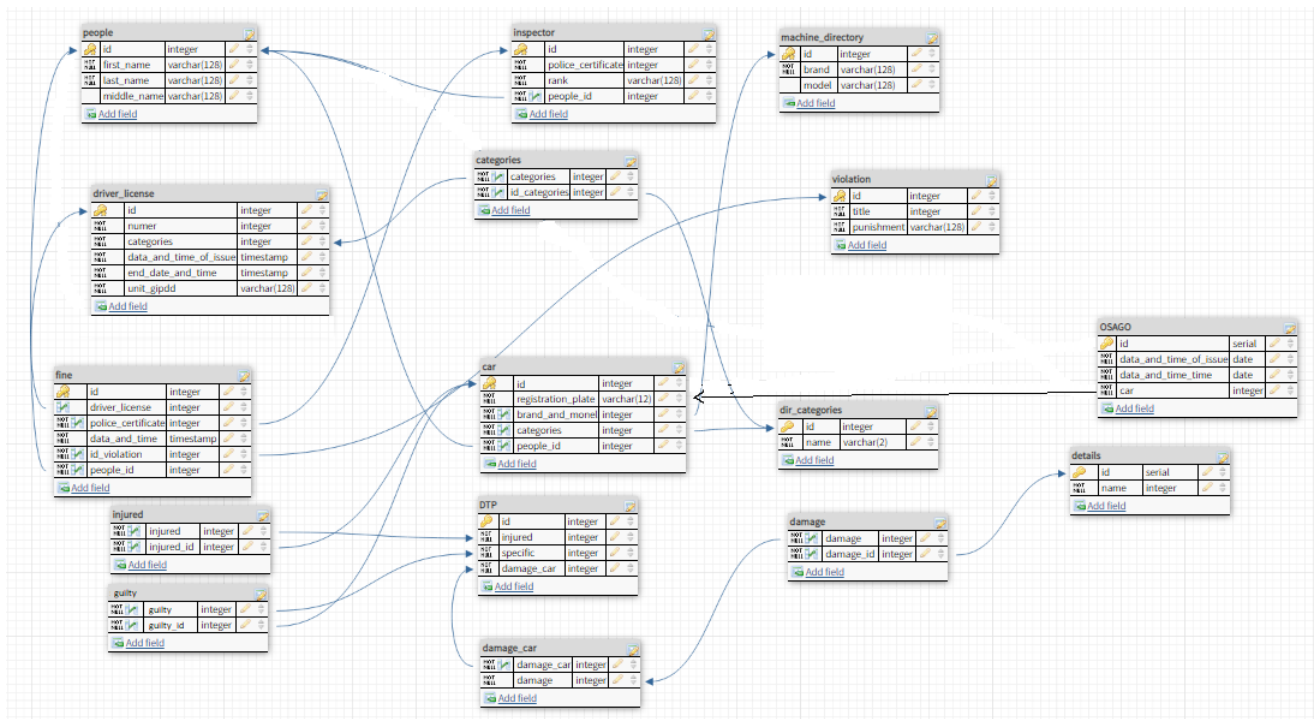


Рис. 4.2: Схема модели 2

4.5. Скрипт создания структуры базы данных после изменения

```
1 CREATE TABLE "people" (  
2     "id" integer PRIMARY KEY,  
3     "first_name" varchar(128) NOT NULL,  
4     "last_name" varchar(128) NOT NULL,  
5     "middle_name" varchar(128)  
6 );  
7  
8 CREATE TABLE "driver_license" (  
9     "id" integer PRIMARY KEY,  
10    "number" integer NOT NULL,  
11    "categories" integer UNIQUE,  
12    "data_and_time_of_issue" TIMESTAMP NOT NULL,  
13    "end_date_and_time" TIMESTAMP NOT NULL,  
14    "unit_gipdd" varchar(128) NOT NULL,  
15    "people_id" integer REFERENCES people (Id) NOT NULL  
16 );  
17  
18 CREATE TABLE "inspector" (  
19     "id" integer PRIMARY KEY,  
20     "police_certificate" integer NOT NULL,  
21     "rank" varchar(128) NOT NULL,  
22     "people_id" integer REFERENCES people (Id) NOT NULL  
23 );  
24  
25 CREATE TABLE "machine_directory" (  
26     "id" integer PRIMARY KEY,  
27     "brand" varchar(128) NOT NULL,  
28     "model" varchar(128)  
29 );  
30  
31 CREATE TABLE "violation" (  
32     "id" integer PRIMARY KEY,  
33     "title" integer NOT NULL,  
34     "punishment" varchar(128) NOT NULL  
35 );  
36  
37 CREATE TABLE "dir_categories" (  
38     "id" integer PRIMARY KEY,  
39     "name" varchar(2) NOT NULL  
40 );  
41  
42 CREATE TABLE "categories" (  
43     "categories" integer REFERENCES driver_license (categories) NOT NULL,  
44     "id_categories" integer REFERENCES dir_categories (id) NOT NULL  
45 );  
46  
47 CREATE TABLE "car" (  
48     "id" integer PRIMARY KEY,  
49     "registration_plate" varchar(12) NOT NULL UNIQUE,  
50     "brand_and_monel" integer NOT NULL,  
51     "categories" integer REFERENCES dir_categories (id) NOT NULL,  
52     "people_id" integer REFERENCES people (Id) NOT NULL  
53 );  
54  
55 CREATE TABLE "fine" (  
56     "id" integer PRIMARY KEY,  
57     "driver_license" integer REFERENCES driver_license (Id) NOT NULL,  
58     "police_certificate" integer REFERENCES inspector (Id) NOT NULL,
```

```

59     "data_and_time" TIMESTAMP NOT NULL,
60     "id_violation" integer NOT NULL,
61     "people_id" integer REFERENCES people (Id) NOT NULL
62 );
63
64 CREATE TABLE "dtp" (
65     "id" integer PRIMARY KEY,
66     "injured" integer UNIQUE,
67     "guilty" integer UNIQUE,
68     "damage_car" integer UNIQUE
69 );
70
71 CREATE TABLE "details" (
72     "id" integer PRIMARY KEY,
73     "name" varchar(128) NOT NULL
74 );
75
76 CREATE TABLE "damage_car" (
77     "damage_car" integer REFERENCES dtp (damage_car) NOT NULL,
78     "damage" integer UNIQUE
79 );
80
81 CREATE TABLE "damage" (
82     "damage_id" integer REFERENCES details (id) NOT NULL,
83     "damage" integer REFERENCES damage_car (damage) NOT NULL
84 );
85
86 CREATE TABLE "injured" (
87     "injured" integer REFERENCES dtp (injured) NOT NULL,
88     "injured_id" integer REFERENCES car (id) NOT NULL
89 );
90
91 CREATE TABLE "guilty" (
92     "guilty" integer REFERENCES dtp (guilty) NOT NULL,
93     "guilty_id" integer REFERENCES car (id) NOT NULL
94 );
95
96 CREATE TABLE "osago" (
97     "id" integer PRIMARY KEY,
98     "data_and_time_of_issue" TIMESTAMP NOT NULL,
99     "end_date_and_time" TIMESTAMP NOT NULL,
100     "car" integer REFERENCES car (id) NOT NULL
101 );

```

4.6. Скрипт заполнения таблиц тестовыми данными после изменения

```

1 INSERT INTO people ( id, first_name, last_name, middle_name)
2 VALUES (1, 'Александр', 'Сидоров', 'Александрович'),
3 (2, 'Иван', 'Иванов', 'Иванович'),
4 (3, 'Петров', 'Григорий', 'Антонович');
5
6 INSERT INTO driver_license ( id, numer, categories, data_and_time_of_issue,
7                             end_date_and_time, unit_gipdd, people_id)
8 VALUES (1, 782212, 1, '2010-05-01', '2020-05-01', 'ГИБДД 7816', 1),
9 (2, 782245, 2, '2012-08-01', '2022-08-01', 'ГИБДД 7816', 2);
10
11 INSERT INTO inspector ( id, police_certificate, rank, people_id)

```



```

12 VALUES (1, 12345, 'Лейтенант', 1),
13 (2, 45678, 'Лейтенант', 2);
14
15 INSERT INTO machine_directory ( id, brand, model)
16 VALUES (1, 'LADA', 'Vesta'),
17 (2, 'LADA', 'Xray'),
18 (3, 'Reno', 'Lagan');
19
20 INSERT INTO violation ( id, title, punishment)
21 VALUES (1, 1000, 'Превышение скорости'),
22 (2, 500, 'Неправильная парковка');
23
24 INSERT INTO dir_categories ( id, name)
25 VALUES (1, 'A'),
26 (2, 'B'),
27 (3, 'C'),
28 (4, 'D');
29
30 INSERT INTO categories ( categories, id_categories)
31 VALUES (1, 1),
32 (1, 1);
33
34 INSERT INTO car ( id, registration_plate, brand_and_monel, categories, people_id)
35 VALUES (1, 'B123KX178', 1, 2, 1),
36 (2, 'A777AA64', 2, 2, 2);
37
38 INSERT INTO fine ( id, driver_license, police_certificate, data_and_time, id_violation,
39 people_id)
40 VALUES (1, 1, 2, '2018-12-10', 1, 1);
41
42 INSERT INTO dtp ( id, injured, guilty, damage_car)
43 VALUES (1, 1, 1, 1),
44 (2, 2, 2, 2);
45
46 INSERT INTO details ( id, name)
47 VALUES (1, 'Левое крыло'),
48 (2, 'Правое крыло'),
49 (3, 'Передний бампер');
50
51 INSERT INTO damage_car (damage_car, damage)
52 VALUES (1, 1),
53 (2, 2);
54
55 INSERT INTO damage (damage_id, damage)
56 VALUES (1, 1),
57 (2, 2);
58
59 INSERT INTO injured (injured, injured_id)
60 VALUES (1, 1),
61 (2, 2);
62
63 INSERT INTO guilty (guilty, guilty_id)
64 VALUES (2, 1),
65 (1, 2);
66
67 INSERT INTO osago (id, data_and_time_of_issue, end_date_and_time, drivers)
68 VALUES (1, '2018-05-01', '2019-05-01', 1),
69 VALUES (2, '2017-12-05', '2019-12-05', 2);
70
71 INSERT INTO drivers (drivers, drivers_id)

```

```
71 VALUES (1, 1),
72 VALUES (2, 2);
```

Во время выполнения 3 лабораторной работы в качестве первичных лю
При выполнении 3 лабораторной работы первичные ключи были типа Integer.
В последствие типы первичных ключей были изменены а serial.

4.7. Скрипт создания структуры базы данны первичный ключ serial

```
1 CREATE TABLE "people" (  
2     "id" serial PRIMARY KEY,  
3     "first_name" varchar(128) NOT NULL,  
4     "last_name" varchar(128) NOT NULL,  
5     "middle_name" varchar(128)  
6 );  
7  
8 CREATE TABLE "driver_license" (  
9     "id" serial PRIMARY KEY,  
10    "number" integer NOT NULL,  
11    "categories" serial UNIQUE,  
12    "data_and_time_of_issue" TIMESTAMP NOT NULL,  
13    "end_date_and_time" TIMESTAMP NOT NULL,  
14    "unit_gipdd" varchar(128) NOT NULL,  
15    "people_id" integer REFERENCES people (Id) NOT NULL  
16 );  
17  
18 CREATE TABLE "inspector" (  
19     "id" serial PRIMARY KEY,  
20     "police_certificate" integer NOT NULL,  
21     "rank" varchar(128) NOT NULL,  
22     "people_id" integer REFERENCES people (Id) NOT NULL  
23 );  
24  
25 CREATE TABLE "machine_directory" (  
26     "id" serial PRIMARY KEY,  
27     "brand" varchar(128) NOT NULL,  
28     "model" varchar(128)  
29 );  
30  
31 CREATE TABLE "violation" (  
32     "id" serial PRIMARY KEY,  
33     "title" varchar(128) NOT NULL,  
34     "punishment" varchar(128) NOT NULL  
35 );  
36  
37 CREATE TABLE "dir_categories" (  
38     "id" serial PRIMARY KEY,  
39     "name" varchar(2) NOT NULL  
40 );  
41  
42 CREATE TABLE "categories" (  
43     "categories" integer REFERENCES driver_license (categories) NOT NULL,  
44     "id_categories" integer REFERENCES dir_categories (id) NOT NULL  
45 );  
46  
47 CREATE TABLE "car" (  
48     "id" serial PRIMARY KEY,
```

```

49     "registration_plate" varchar(12) NOT NULL UNIQUE,
50     "brand_and_monel" integer NOT NULL,
51     "categories" integer REFERENCES dir_categories (id) NOT NULL,
52     "people_id" integer REFERENCES people (Id) NOT NULL
53 );
54
55 CREATE TABLE "fine" (
56     "id" serial PRIMARY KEY,
57     "driver_license" integer REFERENCES driver_license (Id) NOT NULL,
58     "police_certificate" integer REFERENCES inspector (Id) NOT NULL,
59     "data_and_time" TIMESTAMP NOT NULL,
60     "id_violation" integer NOT NULL
61 );
62
63 CREATE TABLE "dtp" (
64     "id" serial PRIMARY KEY,
65     "injured" integer UNIQUE,
66     "guilty" integer UNIQUE,
67     "damage_car" integer UNIQUE
68 );
69
70 CREATE TABLE "details" (
71     "id" serial PRIMARY KEY,
72     "name" varchar(128) NOT NULL
73 );
74
75 CREATE TABLE "damage_car" (
76     "damage_car" integer REFERENCES dtp (damage_car) NOT NULL,
77     "damage" integer NOT NULL UNIQUE
78 );
79
80 CREATE TABLE "damage" (
81     "damage_id" integer REFERENCES details (id) NOT NULL,
82     "damage" integer REFERENCES damage_car (damage) NOT NULL
83 );
84
85 CREATE TABLE "injured" (
86     "injured" integer REFERENCES dtp (injured) NOT NULL,
87     "injured_id" integer REFERENCES car (id) NOT NULL
88 );
89
90 CREATE TABLE "guilty" (
91     "guilty" integer REFERENCES dtp (guilty) NOT NULL,
92     "guilty_id" integer REFERENCES car (id) NOT NULL
93 );
94
95 CREATE TABLE "osago" (
96     "id" serial PRIMARY KEY,
97     "data_and_time_of_issue" TIMESTAMP NOT NULL,
98     "end_date_and_time" TIMESTAMP NOT NULL,
99     "car" varchar(12) REFERENCES car (registration_plate) NOT NULL
100 );

```

4.8. Скрипт заполнения таблиц тестовыми данными первичный ключ serial

```

1 INSERT INTO people (first_name, last_name, middle_name)
2 VALUES ('Александр', 'Сидоров', 'Александрович'),

```

```

3 ('Иван', 'Иванов', 'Иванович'),
4 ('Петров', 'Григорий', 'Антонович');
5
6 INSERT INTO driver_license (number, categories, data_and_time_of_issue,
7                             end_date_and_time, unit_gipdd, people_id)
8 VALUES (782212, 1, '2010-05-01', '2020-05-01', 'ГИБДД 7816', 1),
9 (782245, 2, '2012-08-01', '2022-08-01', 'ГИБДД 7816', 2);
10
11 INSERT INTO inspector (police_certificate, rank, people_id)
12 VALUES (12345, 'Лейтенант', 1),
13 (45678, 'Лейтенант', 2);
14
15 INSERT INTO machine_directory (brand, model)
16 VALUES ('LADA', 'Vesta'),
17 ('LADA', 'Xray'),
18 ('Reno', 'Lagan');
19
20 INSERT INTO violation (title, punishment)
21 VALUES (1000, 'Превышение скорости'),
22 ( 500, 'Неправильная парковка');
23
24 INSERT INTO dir_categories (name)
25 VALUES ('A'),
26 ('B'),
27 ('C'),
28 ('D');
29
30 INSERT INTO categories (id_categories)
31 VALUES (1),
32 (1);
33
34 INSERT INTO car (registration_plate, brand_and_monel, categories, people_id)
35 VALUES ('B123KX178', 1, 2, 1),
36 ('A777AA64', 2, 2, 2);
37
38 INSERT INTO fine (driver_license, police_certificate, data_and_time, id_violation)
39 VALUES (1, 2, '2018-12-10', 1);
40
41 INSERT INTO dtp ( injured, guilty, damage_car)
42 VALUES (1, 1, 1),
43 (2, 2, 2);
44
45 INSERT INTO details (name)
46 VALUES ('Левое крыло'),
47 ('Правое крыло'),
48 ('Передний бампер');
49
50 INSERT INTO damage_car (damage_car)
51 VALUES (1),
52 (2);
53
54 INSERT INTO damage (damage_id)
55 VALUES (1),
56 (2);
57
58 INSERT INTO injured ( injured_id)
59 VALUES (1),
60 (2);
61
62 INSERT INTO guilty (guilty_id)

```

```
63 VALUES (1),
64 (2);
65
66 INSERT INTO osago (data_and_time_of_issue, end_date_and_time, drivers)
67 VALUES ('2018-05-01', '2019-05-01', 1),
68 VALUES ('2017-12-05', '2019-12-05', 2);
69
70 INSERT INTO drivers (drivers, drivers_id)
71 VALUES (1),
72 VALUES (2);
```

5. Выводы

В ходе выполнения данной работы были изучены основы создания скриптов на языке SQL. С помощью SQL-DDL описаны структуры разрабатываемой схемы базы данных. С использованием SQL-DML созданные структуры заполнены тестовыми данными. Изучен синтаксис обновления структуры существующей таблицы.