

## Laboratory work 2

Student: Almat Begaidarov

Instructor: Aibek Kuralbaev

Assistant: Bermagambet Duissek

1.Explain the difference between DDL and DML, give the following examples:

a.at least 3 DDL commands;

b.at least 4 DML commands.

DDL (Data Definition Language) – used for defining database structures (database, tables, constraints and so on), it defines attributes of the table, its statements work on the whole object, doesn't use "WHERE" conditions.

Commands:

**CREATE** - to create a new database or table.

**ALTER** - command is used to alter the content in the table.

**DROP** - to delete some content in the database or table.

**TRUNCATE** - to delete all the content from the table.

**RENAME** - to rename the content in the database.

Examples:

CREATE DATABASE/TABLE students;

DROP DATABASE/TABLE students;

ALTER TABLE students

RENAME COLUMN grades TO gpa;

TRUNCATE TABLE students;

DML (Data Manipulation Language) – used to manage the data (add, retrieve or update tuples), works on selected data of table, "WHERE" conditions can be used to filter and work on selected dataset.

Commands:

**SELECT** - to retrieve the data from the table.

**INSERT** - to push the data in the table.

**UPDATE** - to reform the data in the table.

**DELETE** - to delete the data from the table.

Examples:

```
SELECT * from students;
```

```
INSERT INTO students(name, birth_date)
```

```
VALUES('Manat', '29-05-1997')
```

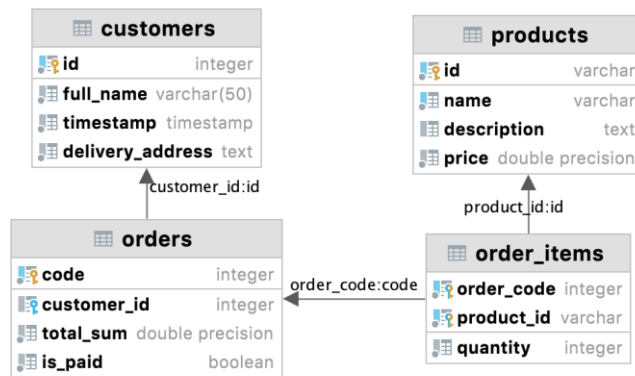
```
UPDATE students SET gpa = 3.61
```

```
WHERE name = 'Manat';
```

```
DELETE FROM students
```

```
WHERE name = 'Manat';
```

2. Write SQL statements to create tables in the figure below:



```
CREATE TABLE customers(
    id int primary key,
    full_name varchar(50) NOT NULL,
    timestamp timestamp NOT NULL,
    delivery_address text NOT NULL
);

CREATE TABLE orders(
    code int primary key,
    customer_id int references customers,
    total_sum double precision NOT NULL CHECK ( total_sum > 0 ),
    is_paid boolean NOT NULL
);

CREATE TABLE products(
    id varchar primary key,
    name varchar NOT NULL UNIQUE,
    description text,
    price double precision NOT NULL CHECK ( price > 0 )
);

CREATE TABLE order_items(
    order_code int references orders,
    product_id varchar references products,
    quantity int NOT NULL CHECK ( quantity > 0 ),
    primary key (order_code, product_id)
);
```

3. Write SQL statements describing tables with appropriate **data types** and **constraints** satisfying the following conditions (*maybe you need additional tables to store data **atomically** and **efficiently***):

- a. a students table storing data such as full name, age, birth date, gender, average grade, information about yourself, the need for a dormitory, additional info.
- b. an instructors table storing data such as full name, speaking languages, work experience, the possibility of having remote lessons.
- c. a lesson participants table storing data such as lesson title, teaching instructor, studying students, room number.

a)

```
CREATE TABLE students(  
    full_name varchar(50) primary key,  
    age int NOT NULL CHECK ( age >= 5 ),  
    birth_date date NOT NULL,  
    gender varchar(10) NOT NULL,  
    average_grade numeric(3, 2) NOT NULL CHECK ( average_grade > 0 AND average_grade <= 4.00  
) ,  
    personal_info text NOT NULL,  
    dormitory_need boolean NOT NULL,  
    extra_info text  
);
```

b)

```
CREATE TABLE instructors(  
    full_name varchar(50) primary key,  
    remote_lessons boolean NOT NULL  
);  
  
CREATE TABLE speaking_languages(  
    instructor_name varchar(50) NOT NULL references instructors,  
    language varchar NOT NULL,  
    primary key (instructor_name, language)  
);  
  
CREATE TABLE work_experience(  
    instructor_name varchar(50) NOT NULL references instructors,  
    company_name varchar NOT NULL,  
    primary key (instructor_name, company_name)  
);
```

c)

```
CREATE TABLE lesson_participants(  
    lesson_title varchar(50),  
    instructor_name varchar(50) references instructors,  
    room_number int NOT NULL,  
    primary key (lesson_title, instructor_name)  
);  
  
CREATE TABLE participating_student(  
    full_name varchar(50) references students,  
    participated_lesson varchar(50),  
    instructor varchar(50),  
    primary key (full_name, participated_lesson),  
    foreign key (participated_lesson, instructor) references lesson_participants  
);
```

4. Give examples of insertion, update and deletion of data on tables from exercise 2.

INSERT INTO customers

VALUES (123, 'Begaidarov Almat', '2021-09-22 17:19:55', '7 microdistrict, 21');

INSERT INTO orders

VALUES (1, 123, 1540, true);

INSERT INTO products

VALUES (17, 'Voda', 'fresh water 0.5l', 150);

INSERT INTO order\_items

VALUES (1, 17, 3);

UPDATE customers

SET full\_name = 'Begaidarov Manat' WHERE id = 123;

UPDATE orders

SET is\_paid = false WHERE customer\_id = 123;

UPDATE products

SET description = 'non-carbonated water' WHERE id = 1;

UPDATE order\_items

SET quantity = 5 WHERE order\_code = 1;

DELETE FROM customers

WHERE id = 123;

DELETE FROM orders

WHERE is\_paid = false;

DELETE FROM products

WHERE price > 150 AND price < 200;

DELETE FROM order\_items

WHERE order\_code = 1;