

Laboratory work №2

1. Explain the difference between DDL and DML, give the following examples:
 - a. at least 3 DDL commands;
 - b. at least 4 DML commands.

Answer:

DDL – it is used to create database scheme and can use be used to define some constraints as well. It basically defines the column (attributes) of the table. It doesn't have any further classifications. Basic command present in DDL are CREATE, DROP, RENAME, ALTER.

```
create table person (  
    person_id int primary key,  
    first_name varchar(50) not null,  
    last_name varchar(50) not null,  
    age int not null,  
    nationality varchar(40) not null  
);
```

```
alter table person  
add country varchar(50);
```

```
drop table person
```

DML – it is used to add, retrieve or update the data. It add or update the row of the table. These rows are called as tuple. It is further classified into Procedural and Non-Procedural DML. Basic commands present in DML update, insert, merge.

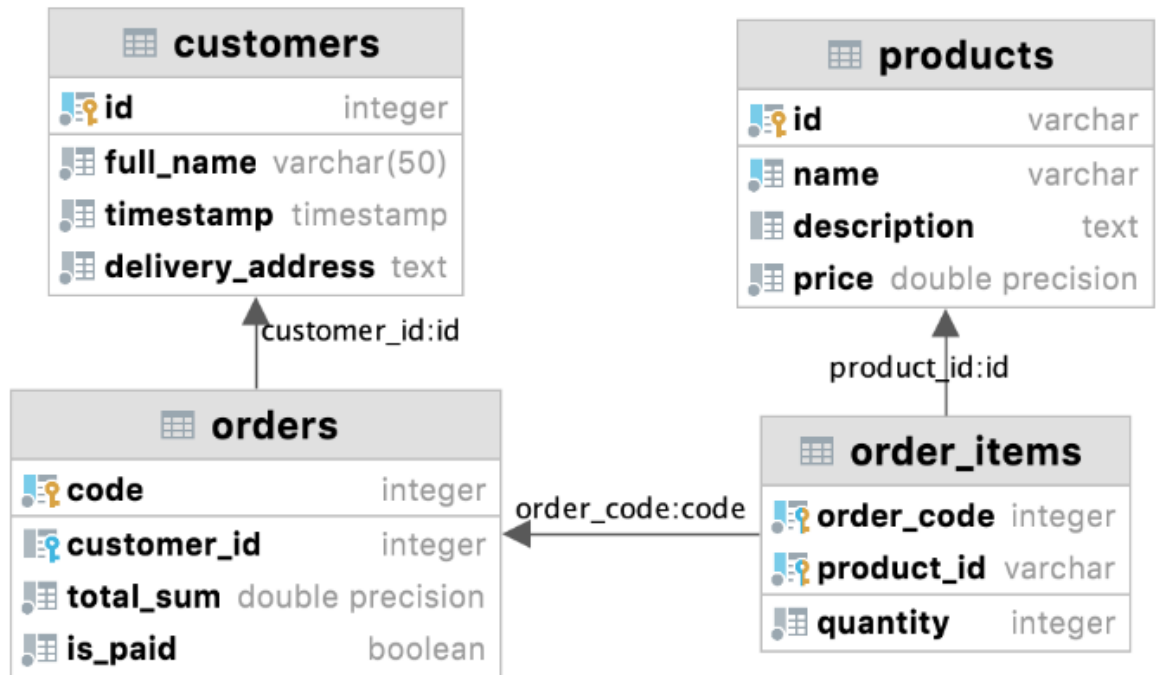
```
insert into person values (1234, 'Madina', 'Bolatova', 18, 'kazakh', 'Kazakhstan');
```

```
update person  
set first_name = 'Temirlan', last_name = 'Bolatov', age = 13, nationality = 'kazakh', country = 'KZ'  
where person_id = 1234;
```

```
delete from person where person_id = 25;
```

```
TRUNCATE TABLE person;
```

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



grey circle - not null, blue column - unique; quantity, total_sum, price > 0

Answer:

```
create table customers (  
  customer_id int,  
  full_name varchar(50) not null,  
  timestamp timestamp not null,  
  delivery_address text not null,  
  primary key (customer_id)  
);
```

```
create table products (  
  product_id varchar,  
  name varchar unique not null,  
  description text,  
  price double precision not null check ( price > 0 ),  
  primary key (product_id)  
);
```

```

create table orders (
    order_code int,
    customer_id int,
    total_sum double precision not null check ( total_sum > 0 ),
    is_paid boolean not null,
    primary key (order_code),
    foreign key (customer_id) references customers
);

```

```

create table order_items (
    order_code int not null,
    product_id varchar not null,
    quantity integer not null check ( quantity > 0),
    primary key (order_code, product_id),
    foreign key (order_code) references orders,
    foreign key (product_id) references products
);

```

3. Write SQL statements describing tables with appropriate **data types** of **constraints** satisfying the following conditions:
- 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.
 - An instructors table storing data such as full name, speaking languages, work experience, the possibility of having remote lessons.
 - A lessons participants table storing data such as lesson title, teaching instructor, studying students, room number.

Answer:

```

CREATE TABLE students (
    full_name varchar PRIMARY KEY,
    age integer NOT NULL,
    birth_date date NOT NULL,
    gender varchar NOT NULL check ( gender = 'Male' or gender = 'Female' ),
    average_grade double precision NOT NULL check ( average_grade >= .0 and average_grade <= 4.0),
    information text NOT NULL,
    need_for_dormitory boolean NOT NULL,
    additional_info text
);

```

```
CREATE TABLE instructors (  
    full_name varchar PRIMARY KEY,  
    remote_lessons boolean NOT NULL  
);  
  
CREATE TABLE work_exp (  
    instructor_name varchar NOT NULL,  
    company_name varchar NOT NULL,  
    PRIMARY KEY(instructor_name, company_name),  
    FOREIGN KEY(instructor_name) REFERENCES instructors(full_name)  
);  
  
CREATE TABLE languages (  
    instructor_name varchar NOT NULL,  
    language_name varchar NOT NULL,  
    PRIMARY KEY(instructor_name, language_name),  
    FOREIGN KEY(instructor_name) REFERENCES instructors(full_name)  
);
```

```
CREATE TABLE lesson_participants (  
    lesson_title varchar,  
    instructor varchar,  
    room_number integer NOT NULL,  
    PRIMARY KEY(lesson_title, instructor),  
    FOREIGN KEY(instructor) REFERENCES instructors(full_name)  
);  
  
CREATE TABLE studying_students (  
    full_name varchar NOT NULL,  
    studying_lesson varchar NOT NULL,  
    teaching_instructor varchar NOT NULL,  
    PRIMARY KEY(full_name, studying_lesson),  
    FOREIGN KEY(studying_lesson, teaching_instructor) REFERENCES lesson_participants,  
    FOREIGN KEY(full_name) REFERENCES students(full_name)  
);
```

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

Answer:

```
INSERT INTO customers values('1', 'Madina Bolatova', '2021-09-23 16:46:30', 'Sholpan 16');
INSERT INTO orders values(1, 1, 1700, true);
INSERT INTO products values(1, 'burger', 'chicken burger with sauce', 1500.00);
INSERT INTO order_items values(1, 1, 10);

UPDATE customers SET delivery_address = 'Telzhan Shonanuly 128' WHERE delivery_address = 'Sholpan 16';
UPDATE orders SET total_sum = 2000 where total_sum = 1700;
UPDATE products SET price = 1750.00 where name = 'burger';
UPDATE order_items SET quantity = 8 where quantity = 10;

DELETE FROM customers WHERE delivery_address = 'Telzhan Shonanuly 128';
DELETE FROM orders WHERE order_code = 1;
DELETE FROM products WHERE name = 'burger';
DELETE FROM order_items WHERE order_code = 1;
```