

## Laboratory work 2

Please write your answers to the pdf file for defence:

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

a. at least 3 DDL commands;

b. at least 4 DML commands.

Data Definition Language - which is used to define data structures.

Data Manipulation Language - which is used to manipulate data itself.

DDL - create tables, alter table are instructions in SQL

DML - insert, update, delete are instructions in SQL

DDL - creating schemas and define some constraints

DML - add or update the row of the table

Basic commands of DDL - create, drop, rename, alter

Basic commands of DML - update, insert, merge

DDL commands

```
CREATE TABLE Customers
(
    Id INT PRIMARY KEY,
    Age INT,
    FirstName VARCHAR(20) NOT NULL,
    LastName VARCHAR(20) NOT NULL,
    Phone VARCHAR(20) NOT NULL UNIQUE
);
```

```
ALTER TABLE zxc
ADD address VARCHAR(20) NOT NULL;
```

```
drop table zxc;
```

```
ALTER TABLE zxc
RENAME TO custommerrz;
```

DML commands

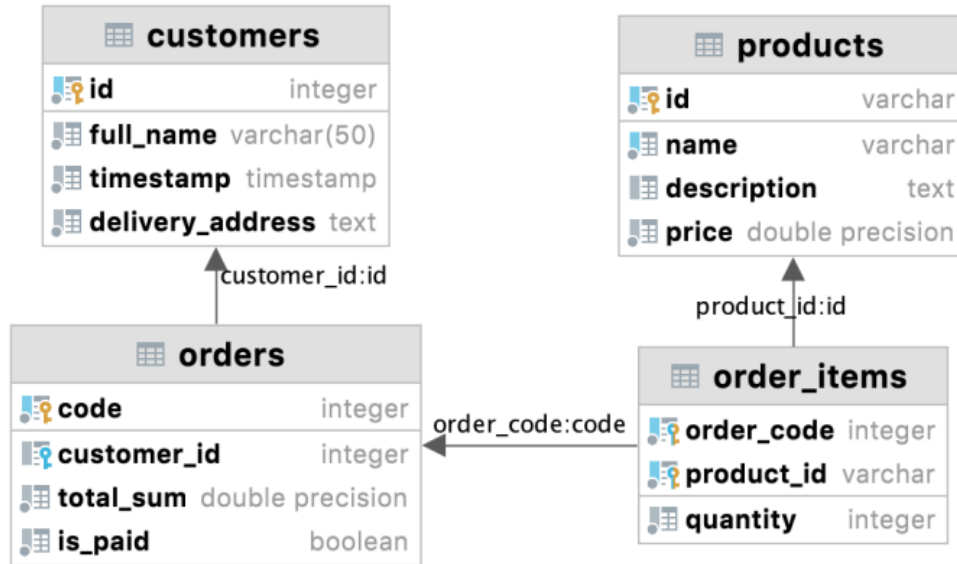
```
INSERT into custommerrz(id, age, FirstName, LastName, Phone, address) values
(555, 18, 'Torekeldi', 'Niyazbek', '+7705', 'Zenk');
```

```
SELECT * from custommerrz;
```

```
UPDATE custommerrz SET age=19 WHERE age = 18;
```

```
DELETE from custommerrz where id = 555;
```

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



grey circle - not null, blue column - unique; quantity, total\_sum, price > 0

```
CREATE TABLE order_items
(
    order_code INTEGER NOT NULL UNIQUE,
    product_id VARCHAR NOT NULL UNIQUE,
    quantity INTEGER NOT NULL CHECK(quantity > 0),
    PRIMARY KEY(order_code, product_id)
);

CREATE TABLE products(
    id VARCHAR PRIMARY KEY,
    name VARCHAR UNIQUE NOT NULL,
    description TEXT,
    price DOUBLE PRECISION NOT NULL CHECK(price > 0),
    FOREIGN KEY (ID) REFERENCES order_items(product_id)
);

CREATE TABLE orders
(
    code INTEGER PRIMARY KEY,
    customer_id INTEGER UNIQUE,
    total_sum DOUBLE PRECISION NOT NULL CHECK(total_sum >
0),
    is_paid BOOLEAN NOT NULL,
    FOREIGN KEY (code) REFERENCES order_items(order_code)
);

CREATE TABLE customers
(
    id INTEGER PRIMARY KEY,
    full_name varchar(50) NOT NULL,
    timestamp TIMESTAMP NOT NULL,
    delivery_address TEXT NOT NULL,
    FOREIGN KEY (ID) REFERENCES orders(customer_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 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 lesson participants table storing data such as lesson title, teaching instructor, studying students, room number.

```
CREATE TABLE students
(
    full_name VARCHAR PRIMARY KEY CHECK (char_length(full_name)>5),
    age INTEGER NOT NULL,
    birth_date DATE NOT NULL,
    gender VARCHAR NOT NULL,
    average_grade DOUBLE PRECISION NOT NULL,
    about_yourself TEXT NOT NULL CHECK (char_length(about_yourself)>5),
    need_dormitory boolean NOT NULL,
    additional_info TEXT
);
insert into students(full_name, age, birth_date, gender, average_grade, about_yourself, need_dormitory, additional_info)
values('Torekeldi', 18, '27.10.2002', 'Male', 4.0, 'Make your life more happy', False, NULL);
insert into students(full_name, age, birth_date, gender, average_grade, about_yourself, need_dormitory, additional_info)
values('Student 2', 19, '22.11.2001', 'Male', 3.44, 'Only positive vibes', False, NULL);
insert into students(full_name, age, birth_date, gender, average_grade, about_yourself, need_dormitory, additional_info)
values('Student 3', 17, '27.10.2003', 'Female', 3.67, 'Take care about yourself', False, NULL);

CREATE TABLE instructors(
    full_name VARCHAR UNIQUE CHECK (char_length(full_name)>5),
    languages TEXT NOT NULL,
    work_exp INTEGER NOT NULL,
    possibility_remote_lessons boolean NOT NULL,
    PRIMARY KEY(full_name)
);
insert into instructors(full_name, languages, work_exp, possibility_remote_lessons) values ('Paul Davis', 'English,
Mexican',3,True);
insert into instructors(full_name, languages, work_exp, possibility_remote_lessons) values ('Aibek Kuralbayev', 'English,
Russian, Kazakh',5,True);

CREATE TABLE room(
    id INTEGER,
    subject_name VARCHAR NOT NULL,
    student_name VARCHAR NOT NULL,
    instructor_name VARCHAR NOT NULL,
    date_time timestamp NOT NULL,
    attendance BOOLEAN NOT NULL,
    FOREIGN KEY(student_name) REFERENCES students(full_name),
    FOREIGN KEY(instructor_name) REFERENCES instructors(full_name),
    PRIMARY KEY(student_name,date_time)
);

insert into room(id, subject_name, student_name, instructor_name, date_time, attendance) values(111,'Discrete Structure',
'Torekeldi', 'Aibek Kuralbayev', '19.09.2021 15:00:00', True);
insert into room(id, subject_name, student_name, instructor_name, date_time, attendance) values(111,'Discrete Structure',
'Student 2', 'Aibek Kuralbayev', '19.09.2021 15:00:00', False);
insert into room(id, subject_name, student_name, instructor_name, date_time, attendance) values(111,'Discrete Structure',
'Student 3', 'Aibek Kuralbayev', '19.09.2021 15:00:00', True);

CREATE TABLE lesson_participants
(
    lesson_title VARCHAR NOT NULL,
    lesson_code VARCHAR NOT NULL,
    instructor_name VARCHAR NOT NULL,
    student_name VARCHAR NOT NULL,
    room_number INTEGER NOT NULL,
    date_time TIMESTAMP NOT NULL,
    PRIMARY KEY(lesson_title,student_name,date_time)
);
insert into lesson_participants(lesson_title, lesson_code, instructor_name, student_name, room_number, date_time)
select 'Discrete Structure', 'DISC201', instructor_name, student_name, id, date_time from room where attendance = True;
```

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

```
insert into order_items(order_code, product_id, quantity) values (111,'CSCI2020',
100);
insert into order_items(order_code, product_id, quantity) values (222,'CSCI2021',
100);
insert into products(id, name, description, price) values('CSCI2020','Make you
happy', 'Make your own decision', 555);
insert into orders(code, customer_id, total_sum, is_paid) values
(111,21,2000,False);
insert into customers(id, full_name, timestamp, delivery_address) values(21,
'21Savage', '2002-10-27', 'Zenkov');

UPDATE customers SET delivery_address = 'KBTU' where delivery_address = 'Zenkov';
UPDATE orders SET total_sum = 2222 where total_sum = 0;

delete from customers where id = 21;
delete from products where name = 'Make you happy';
```