1

DDL	DML
It stands for Data Definition Language.	It stands for Data Manipulation Language.
It is used to create database schema and can be used to define some constraints as well.	It is used to add, retrieve or update the data.
It basically defines the column (Attributes) of the table.	It add or update the row of the table. These rows are called as tuple.
It doesn't have any further classification.	It is further classified into Procedural and Non-Procedural DML.
Basic command present in DDL are CREATE, DROP, RENAME, ALTER etc.	BASIC command present in DML are UPDATE, INSERT, MERGE etc.
DDL does not use WHERE clause in its statement.	While DML uses WHERE clause in its statement.

Examples of DDL commands:

- CREATE – is used to create the database or its objects (like table, index, function, views, store procedure and triggers).
- DROP – is used to delete objects from the database.
- ALTER-is used to alter the structure of the database.

Examples of DML:

- INSERT – is used to insert data into a table.
- UPDATE – is used to update existing data within a table.
- DELETE – is used to delete records from a database table.
- SELECT – command to fetch data or values from the database

2

```
CREATE TABLE customers
  (
    Id int unique not null primary key,
    Full_name varchar(50) not null,
    Timestamp_adress timestamp not null,
    Delivery_adress text not null
  )


CREATE TABLE orders
   (
      code int not null unique primary key ,
      customer_id int,
      total_sum double precision not null,
      is_paid boolean not null,
      foreign key (customer_id) references customers(id)
   )

CREATE TABLE products
   (
      id varchar primary key unique not null,
      name varchar unique not null,
      description text,
      price double precision not null
   )

CREATE TABLE order_items
   (
      order_code int unique not null,
      product_id varchar unique not null,
      quantity int not null,
      primary key (order_code,product_id),
      foreign key (order_code) references orders(code),
      foreign key (product_id) references products(id)
   )
```

3

```
CREATE TABLE student
   (
      id int unique not null primary key,
      fullname varchar not null,
      age smallint not null,
      birthdate date not null,
      gender varchar not null,
      average_grade double precision not null,
      information text,
      need_for_dormitory boolean not null,
      additional_info text,
        check ( age >=0 or age <=200 ),
        check ( gender = 'Male' or gender = 'Female'),
        check ( average_grade>=0.0 and average_grade<=4.0 )
)

CREATE TABLE language
   (
        id int unique not null primary key,
        name varchar not null,
)

CREATE TABLE instructor
   (
        id int unique not null primary key,
        fullname varchar not null,
        work_experience smallint not null,
        possibility_of_remote_lessons boolean not null,
        check ( work_experience>=0 )

)

CREATE TABLE lesson_participants
   (
        id int not null unique primary key,
        instructor_id int not null,
        title varchar not null,
        room varchar not null ,
        foreign key (instructor_id) references instructor(id),
        check ( room>='001' and room<='900' )
)

CREATE TABLE studiyng
   (
        student_id int not null ,
        lesson_id int not null ,
        foreign key (student_id) references student(id),
        foreign key (lesson_id) references lesson_participants(id)
)
```

4

```
INSERT INTO customers VALUES

(2, 'Name2', TIMESTAMP '2021-09-09 17:12:23','Adress')
DELETE FROM customers
WHERE Full_name = ' Random Name'
```

```
INSERT INTO orders VALUES
(1234, 2, 999999.99, true)

UPDATE orders
SET total_sum = total_sum/2
WHERE code = 1234
```