1.a) Data Definition Language helps you to define the database structure or schema. DDL commands help you to create the structure of the database and the other database objects. Its commands are auto-committed so, the changes are saved in the database permanently. The full form of DDL is Data Definition Language.

Command 1)

CREATE

```
CREATE TABLE TABLE_NAME (COLUMN_NAME DATATYPES[,....]);

Create database university;
Create table students;
Create view for_students;
```

Command 2)

DROP

```
DROP TABLE ;

Drop object_type object_name;
Drop database university;
Drop table student;
```

Command 3)

ALTER

```
ALTER TABLE table_name ADD column_name COLUMN-definition;

Alter table guru99 add subject varchar;
```

1.b)

DML commands it to allow you to manage the data stored in the database, although DML commands are not auto-committed. Moreover, they are not permanent. So, It is possible to roll back the operation. The full form of DML is Data Manipulation Language.

Command 1)

INSERT

```
INSERT INTO students (RollNo, FIrstName, LastName) VALUES ('60', 'Tom',
'Erichsen');
```

Command 2)

UPDATE

```
UPDATE students
SET FirstName = 'Jhon', LastName=' Wick'
WHERE StudID = 3;
```

Command 3)
DELETE

```
DELETE FROM table_name [WHERE condition];
```

Command 4)
SELECT

```
SELECT expression
FROM tableName
[WHERE condition];
```

2.

```
create table products(
    id varchar unique primary key,
    name varchar unique ,
    description text,
    price double precision
                     );
```

```
create table customers(
    id integer unique ,
    full_name varchar(50),
    timestamp timestamp,
    delivery_address text,
    primary key(id)
);
```

```
create table orders(
    code integer unique ,
    customer_id integer,
    total_sum double precision,
    is_paid boolean,
    primary key(code),
    foreign key (customer_id) references customers(id)
);
```

```
create table order_items(
    order_code integer unique ,
    product_id varchar unique ,
    quantity integer ,
    primary key (order_code, product_id),
    foreign key (product_id) references products(id),
    foreign key (order_code) references  orders(code));
```

3.

A.

```
create table  student(
    full_name varchar(100) not null ,
    id int not null primary key ,
    age int not null,
    birth_date date not null,
    gender text default 'unknown',
    average_grade double precision,
    information_ab_yourself text null,
    address varchar(50)
);
```

B.

```
create table instructors(
    full_name varchar(50) not null primary key,
    speak_language text default 'unknown',
    work_exp int,
    online boolean default false
);
```

C.

```
create table lecture_participants(
    lesson_title text default 'not defined',
    instructor_name varchar(50) not null ,
    students int not null ,
    room_number int,
    foreign key (instructor_name) references Instructors(full_name)

);
```

4.

1.Insert

```
insert into customers  (full_name, id)
VALUES ('Baigazy Iliyas', 77784);
```

2.Update

```
update customers set full_name = 'Alikhan Didar', id = 778
where full_name = 'Baigazy Iliyas';
```

3.Delete

```
delete from customers where full_name = 'Alikhan Didar';
```