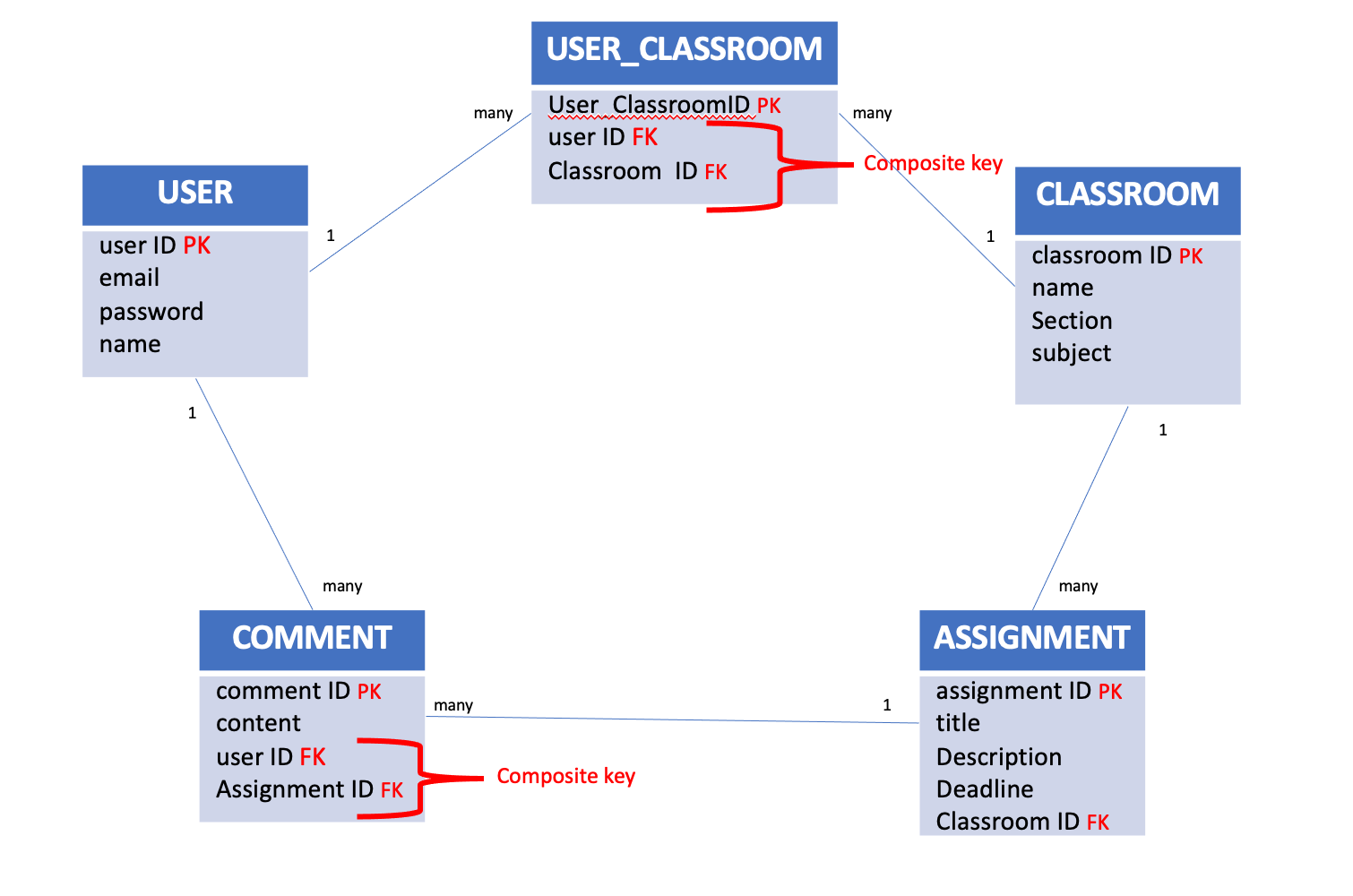
# C2- S5 - PRACTICE

*NOTE: check your* ***THEORY slides*** *to answer those questions!*

# EXERCISE 1 – GOOGLE CLASSROOM DATABASE



Here is the Entity Relation Diagram of the Google Classroom Database you designed in Chapter 1. You are now going to put it in MySQL!

**Q1 –** Write a statement to create the google classroom database, and to tell MySQL you are now working with it.

CREATE DATABASE IF NOT EXISTS GOOGLE\_CLASSROOM;

use GOOGLE\_CLASSROOM;

show databases;

**Q2** – For each table (USER, USER\_CLASSROOM, CLASSROOM, ASSIGNMENT, COMMENT), complete the following arrays, by specifying for each field:

* + The field type (SQL type) and size
  + Can be null or not?
  + Is a primary key or foreign keys?
* **USER TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type / size | Null? | Key |
| userId | int(10) | no | PRI |
| email | varchar(100) | No |  |
| password | varchar (100) | no |  |
| name | varchar (100) | no |  |

**USER\_CLASSROOM TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type / size | Null? | Key |
| userId | int(10) | No | CK |
| classId | int(10) | No | CK |
|  |  |  |  |

**CLASSROOM TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type / size | Null? | Key |
| classId | int(10) | No | PRI |
| name | varchar (100) | No |  |
| section | datetime | No |  |
| subject | varchar (100) | No |  |

**ASSIGNMENT TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type / size | Null? | Key |
| assignment\_Id | int(10) | no | PRI |
| title | varchar (100) | no |  |
| description | varchar (100) | yes |  |
| Deadline | Datetime | No |  |
| classroom\_Id | int(10) |  | MUL |

**COMMENT TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type / size | Null? | Key |
| comment\_Id | int(10) | No | PRI |
| content | varchar(100) | No |  |
| user\_Id | int (10) | No | MUL |
| assignment\_ID | inr(10) | no | MUL |

**Q3** – Write the SQL statement to create the 5 tables with appropriate properties.

1. User Table

create table if not EXISTS user(

    userID int (10) auto\_increment primary key,

    email varchar(100) not null,

    passowrd varchar(100) not null,

    name varchar (100) not null

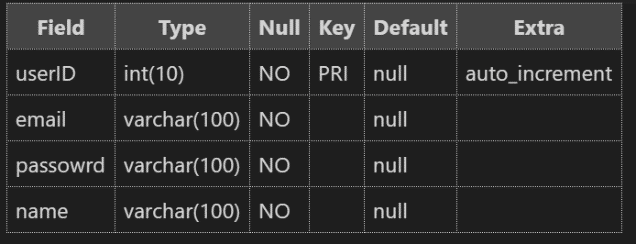
);

1. Classroom Table

create table if not EXISTS classroom (

    classroomID int(10) auto\_increment primary key,

    name varchar(100) not null,

    section datetime,

    subject varchar(100)

);

1. User\_Classroom Table

create table if not exists user\_classroom(

    userID int,

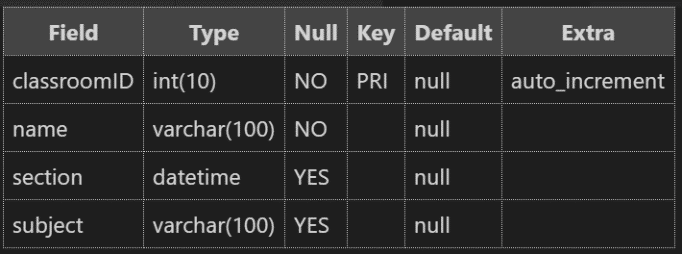
    classroomID int,

    foreign key (userID) references user (userID),

    foreign key(classroomID) references classroom (classroomID)

);



1. Assignment Table

create table if not exists assignment (

    assignmentID int(10) auto\_increment primary key,

    title varchar(100) not null,

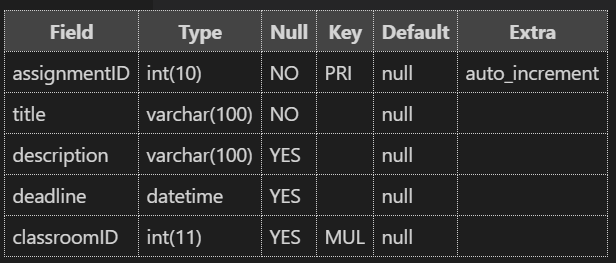
    description varchar(100),

    deadline datetime,

    classroomID int,

    foreign key (classroomID) references classroom(classroomID)

);



1. Comment Table

create table if not exists comment (

    commentID int(10) auto\_increment primary key,

    content varchar(100),

    userID int,

    assignmentID int,

    foreign key (userID) references user(userID),

    foreign key (assignmentID) references assignment(assignmentID)

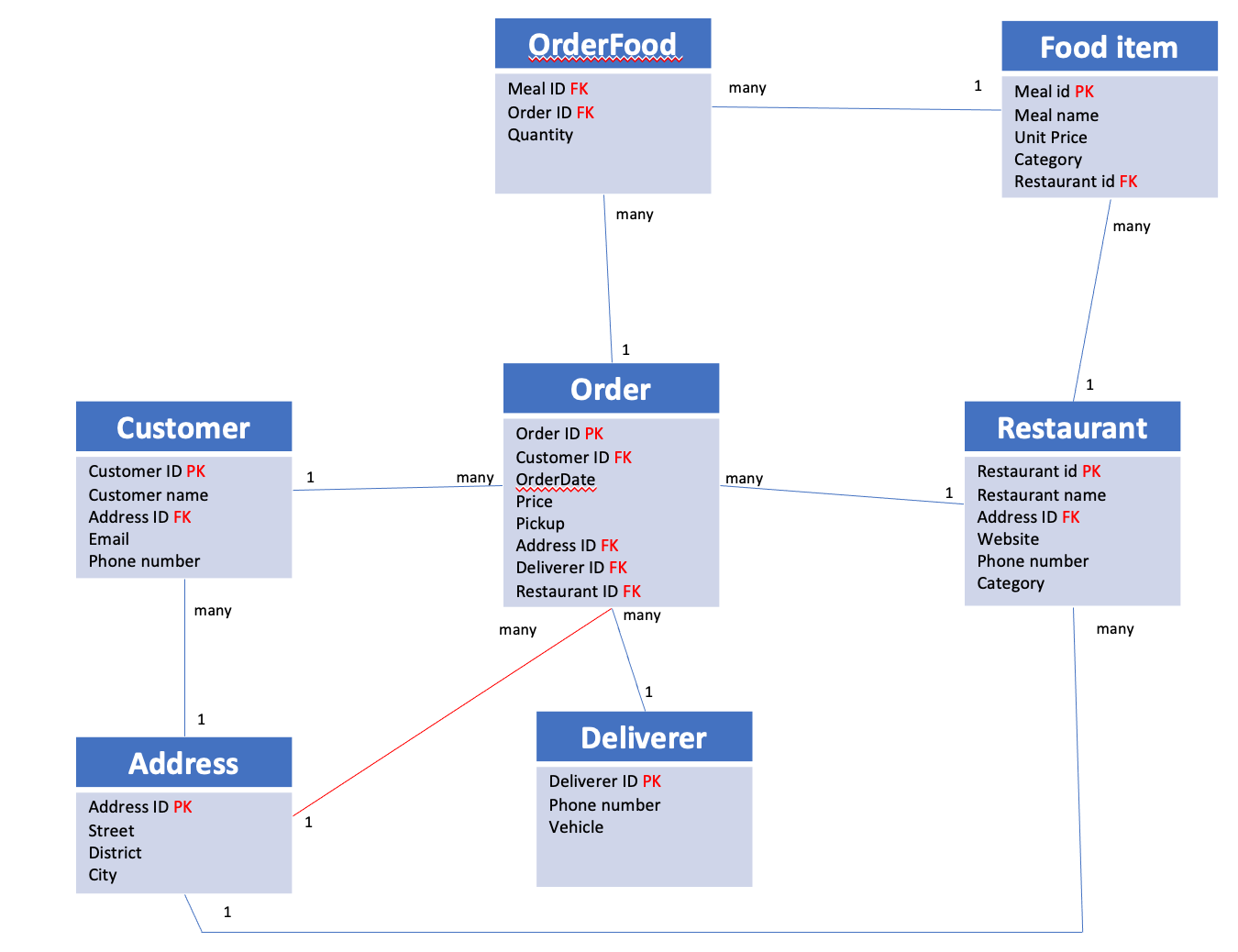
);



WARNING: Create the tables in the right order to respect the Foreign Key constraints.

**Q4 –** Write statements to insert at least 3 records in each table.

# EXERCISE 2 – FOODPANDA DATABASE

****

Here is the Entity Relation Diagram of the Foodpanda Database you designed in Chapter 1. You are now going to put it in MySQL!

**Q1 –** Write a statement to create the Foodpanda classroom database, and to tell MySQL you are now working with it.

**Q2** – For each table of the database, complete the following array, by specifying for each field:

* + The field type (SQL type) and size
  + Can be null or not?
  + Is a primary key or foreign keys?

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type / size | Null? | Key |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Q3** – Write the SQL statement to create the tables with appropriate properties.

WARNING: Create the tables in the right order to respect the Foreign Key constraints.

**Q4 –** Write statements to insert between 2 and 4 records in each table.