SQL ASSIGNMENT

DAY 1

Insert Records - Department

Insert the following records into the department table
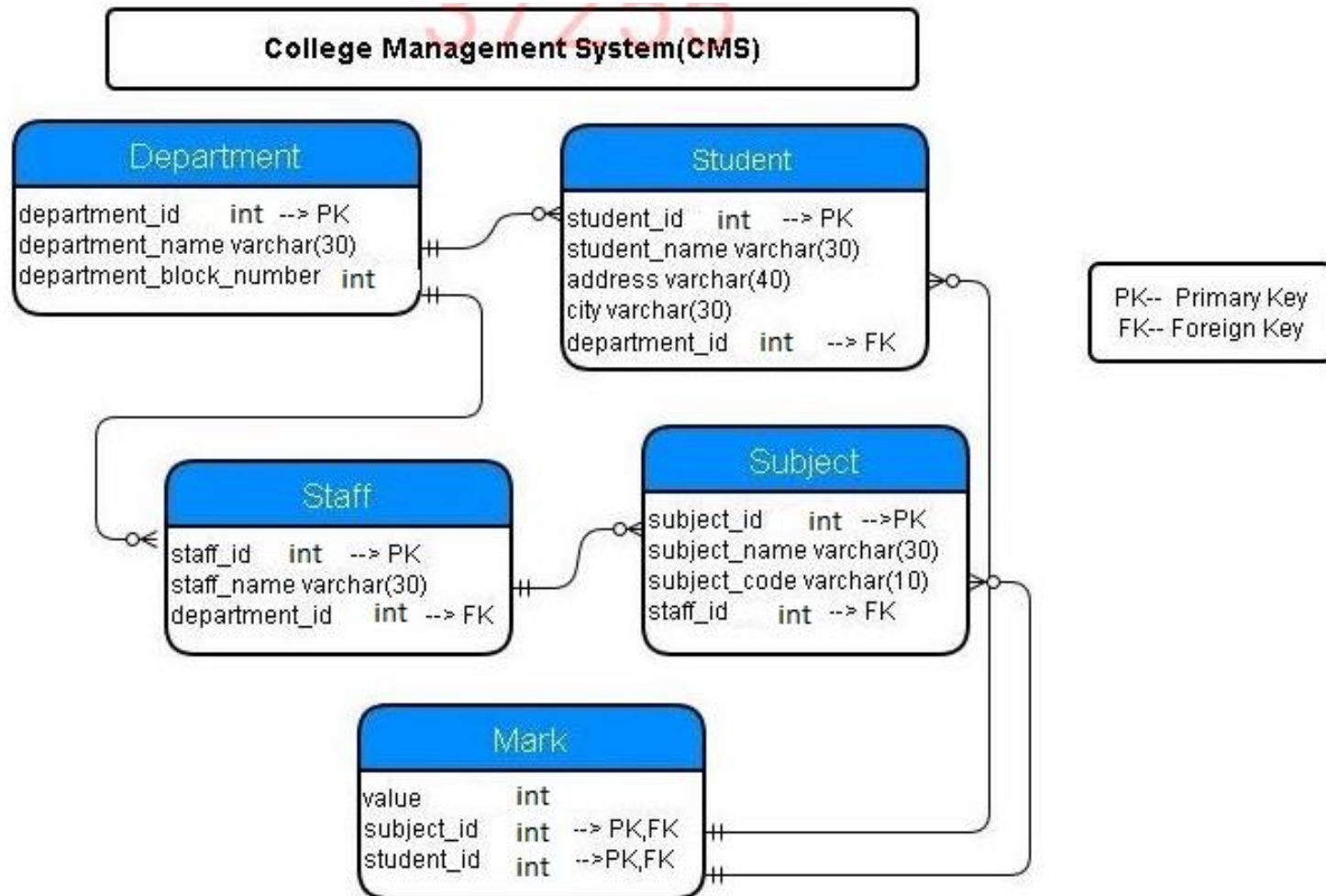
Department_id	Department_name	department_block_number
1	CSE	3
2	IT	3
3	SE	3

Department name based on block number

Write a query to display the names of the departments in block number 3. Sort the records in ascending order.
NOTE: Maintain the same sequence of column order, as specified in the question description

Refer the below schema diagram:

College Management System(CMS)

Department
- department_id   int --> PK
- department_name varchar(30)
- department_block_number   int

Student
- student_id   int   --> PK
- student_name varchar(30)
- address varchar(40)
- city varchar(30)
- department_id   int   --> FK

Staff
- staff_id   int   --> PK
- staff_name varchar(30)
- department_id   int --> FK

Subject
- subject_id   int --> PK
- subject_name varchar(30)
- subject_code varchar(10)
- staff_id   int --> FK

Mark
- value   int
- subject_id   int --> PK,FK
- student_id   int -->PK,FK

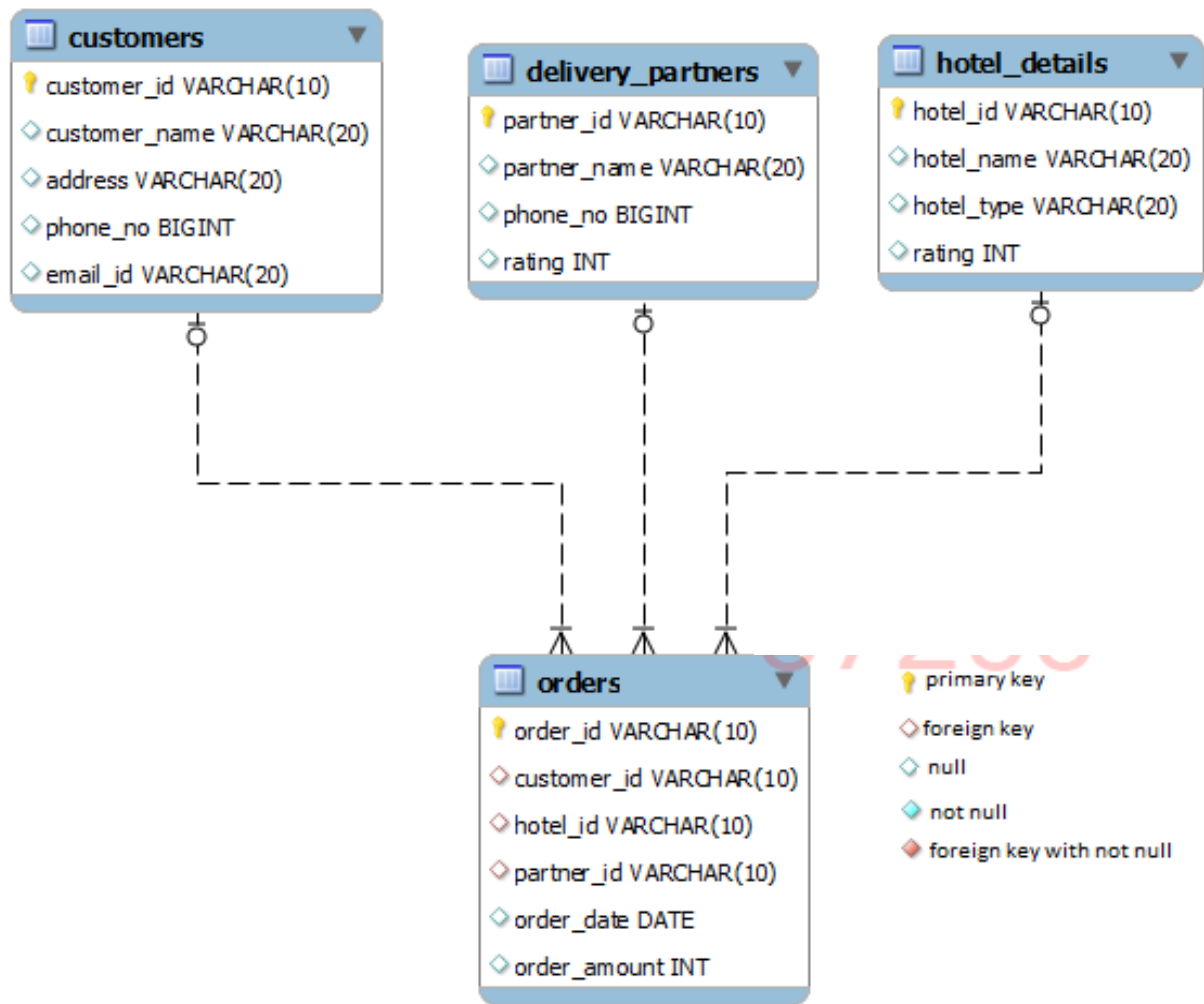PK-- Primary Key
FK-- Foreign Key

NOTE: Maintain the same sequence of column order, as specified in the question description

Delivery Partner details based on rating

Write a query to display partner id, partner name, phone number of delivery partners whose rating is between 3 to 5, sort the result based on partner id.

(Hint: Use **Delivery_partners** table to retrieve records.)

NOTE: Maintain the same sequence of column order, as specified in the question description

customers
- 🔑 customer_id VARCHAR(10)
- ◇ customer_name VARCHAR(20)
- ◇ address VARCHAR(20)
- ◇ phone_no BIGINT
- ◇ email_id VARCHAR(20)

delivery_partners
- 🔑 partner_id VARCHAR(10)
- ◇ partner_name VARCHAR(20)
- ◇ phone_no BIGINT
- ◇ rating INT

hotel_details
- 🔑 hotel_id VARCHAR(10)
- ◇ hotel_name VARCHAR(20)
- ◇ hotel_type VARCHAR(20)
- ◇ rating INT

orders
- 🔑 order_id VARCHAR(10)
- ◇ customer_id VARCHAR(10)
- ◇ hotel_id VARCHAR(10)
- ◇ partner_id VARCHAR(10)
- ◇ order_date DATE
- ◇ order_amount INT

- 🔑 primary key
- ◇ foreign key
- ◇ null
- ◇ not null
- 🔶 foreign key with not null

Car & owner details based on car type

Write a query to display car id, car name and owner id of all the cars whose car type is 'Hatchback' or 'SUV'. Sort the result based on car id.

(Hint: Use **CARS** tables to retrieve records. Data is **case-sensitive**. E.g: Car_type='Hatchback'. Use IN operator)

NOTE: Maintain the same sequence of column order, as specified in the question description

Car details based on type and name

Write a query to display car id, car name and car type of Maruthi company 'Sedan' type cars. Sort the result based on car id.

(HINT : Use **Cars** table to retrieve records.car name='Maruthi Swift'.car type='Sedan'.**Data is case sensitive.**)

NOTE: Maintain the same sequence of column order, as specified in the question description

owners

- 🔑 owner_id VARCHAR(10)
- ◇ owner_name VARCHAR(20)
- ◇ address VARCHAR(20)
- ◇ phone_no BIGINT
- ◇ email_id VARCHAR(20)

customers

- 🔑 customer_id VARCHAR(10)
- ◇ customer_name VARCHAR(20)
- ◇ address VARCHAR(20)
- ◇ phone_no BIGINT
- ◇ email_id VARCHAR(20)

Legend:

- 🔑 primary key
- ◇ foreign key
- ◇ null
- ◇ not null
- 🔴 foreign key with not null

cars

- 🔑 car_id VARCHAR(10)
- ◇ car_name VARCHAR(20)
- ◇ car_type VARCHAR(20)
- ◇ owner_id VARCHAR(10)

rentals

- 🔑 rental_id VARCHAR(10)
- ◇ customer_id VARCHAR(10)
- ◇ car_id VARCHAR(10)
- ◇ pickup_date DATE
- ◇ return_date DATE
- ◇ km_driven INT
- ◇ fare_amount DOUBLE(10,2)