Mid Term

-- QS 1: Write the difference between Primary Key and Composite Primary Key.

- A **Primary Key** uses one column to uniquely identify records.
- A **Composite Primary Key** uses a combination of columns to uniquely identify records when one column is not enough.

-- QS 2: Write the difference between using JOIN Query and not using JOIN query.

- Using **JOIN** lets retrieve related data from multiple tables in one query, reducing redundancy and improving data organization. whereas **not using JOIN** limits querying a single table, which may lead to data duplication and less efficient data management.

-- QS 3:

- **Create a table of Employees which has the following fields**
  – First Name, Last Name, Date of Birth, Department Id, Salary

```
        create table Employees (
        employee_id INT primary key,
        first_name VARCHAR(50) not null,
        last_name VARCHAR(50) not null,
        date_of_birth DATE not null,
        department_id INT,
        salary DECIMAL(10, 2) not null,
        foreign key(department_id) references  Departments
(department_id)
);
```

- **Create a table of Departments which has the following fields**
  – Department Id, Department Name

```
        create table Departments (
                department_id INT primary key,
                department_name VARCHAR(50) not null
        );
```

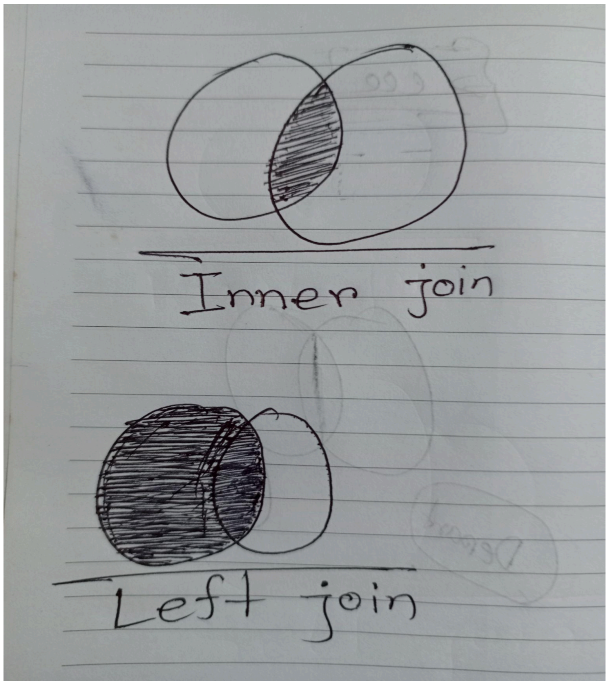
-- QS 4: Write SQL Query to get the second max salary

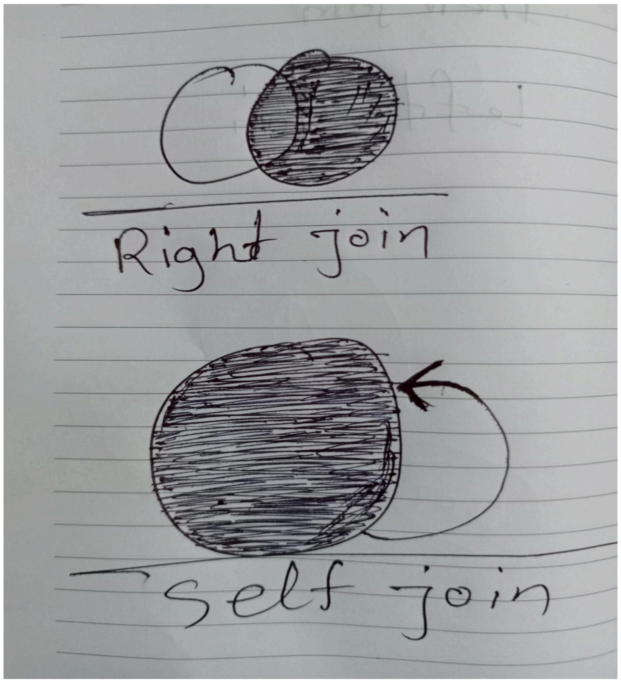
```
with max_salary as (select max(salary) from employees)

select max(salary) as second_max_salary from employees
where salary < (select * from max_salary);
```

-- QS 5: Write SQL Query to show the department names and the average salary of the departments.

```
select departments.department_name, avg(salary)
from employees join departments
        on departments.department_id = employees.department_id
        group by employees.department_id;
```

-- QS 6: Illustrate the INNER, LEFT, RIGHT, SELF Joins.





-- QS 7: What is a subquery? Write with an example.

- A subquery is a SQL query that is embedded within another SQL query. Example:

  ```
  -- this is main Query,
  select max(salary) as second_max_salary from employees
  where salary < (
                      select max(salary) from employees);  -- and this is subQuery.
  ```

-- QS 8: Show the names of the employees who get less salary than Steven.

```
with steven_salary as (select max(salary) from employees where first_name = "Steven")
select * from employees where salary < (select * from steven_salary);
– There were TWO  Steven in DummyDB. That's why I use MAX(Salary) all of Steven.
```

-- QS 9: Count the number of employees of each job type.

```
select employees.job_id, count(*)
from employees join jobs on employees.job_id = jobs.job_id
group by employees.job_id;
```

-- QS 10: Show the names of Departments which doesn’t have any employees.

```
select department_name from departments
left join employees on departments.department_id = employees.department_id
where employees.employee_id is null;
```