### Example: Employee Database

#### EMPLOYEE

<table>
<thead>
<tr>
<th>emp_id</th>
<th>name</th>
<th>dept_id</th>
<th>mgr_id</th>
<th>salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>57</td>
<td>Alex</td>
<td>1</td>
<td>-</td>
<td>84000</td>
</tr>
<tr>
<td>23</td>
<td>Jen</td>
<td>1</td>
<td>57</td>
<td>72000</td>
</tr>
<tr>
<td>85</td>
<td>Max</td>
<td>3</td>
<td>19</td>
<td>51000</td>
</tr>
<tr>
<td>44</td>
<td>Tim</td>
<td>2</td>
<td>37</td>
<td>63000</td>
</tr>
</tbody>
</table>

#### DEPARTMENT

<table>
<thead>
<tr>
<th>dept_id</th>
<th>name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Sales</td>
</tr>
<tr>
<td>2</td>
<td>Accounting</td>
</tr>
<tr>
<td>3</td>
<td>HR</td>
</tr>
</tbody>
</table>
Query #1

- Find all employees in the Sales and HR departments.
Query #1

- Find all employees in the Sales and HR departments.

```
select e.name
from employee as e, department as d
where e.dept_id = d.dept_id
  and (d.name = 'Sales'
       or d.name = 'HR')
```

OR

```
select e.name
from employee as e
join department as d
  on d.dept_id = e.dept_id
where d.name = 'Sales'
  or d.name = 'HR'
```
Query #1

Find all employees in the Sales and HR departments.

```
select e.name
from employee as e
where e.dept_id in (
    select d.dept_id
    from department as d
    where d.name = 'Sales'
    or d.name = 'HR')
```

IN + SUBQUERY
(WHERE CLAUSE)
Query #2

- Find all employees with the maximum salary.
Query #2

- Find all employees with the maximum salary.

```sql
select e.name
from employee as e
where e.salary = (  
    select max(e2.salary)  
    from employee as e2)
```

**SCALAR SUBQUERY**
*(WHERE CLAUSE)*
Query #2

- Find all employees with the maximum salary.

```sql
select e.name
from employee as e
join (
    select max(e2.salary) as maxsal
    from employee as e2
) on e.salary = maxsal
```

SUBQUERY
(FROM CLAUSE)
Query #3

- Find the average salary for each department.
Query #3

Find the average salary for each department.

```sql
SELECT d.name,
       AVG(e.salary) AS avgsal
FROM department AS d
JOIN employee AS e
    ON e.dept_id = d.dept_id
GROUP BY d.name
```
Query #3

- Find the average salary for each department.

```sql
select d.name,
       (select avg(e.salary)
        from employee e
        where e.dept_id = d.dept_id)
       as avgsal
from department as d
```

SCALAR SUBQUERY
(SELECT CLAUSE)
Query #4

- Find the departments with at most one manager.
Find the departments with at most one manager.

```
select d.name
from department as d
join employee as e
  on e.dept_id = d.dept_id
where e.mgr_id is null
group by d.name
having count(e.emp_id) <= 1
```
Find the departments with at most one manager.

```
select d.name
from department as d
where unique (
    select e.dept_id
    from employee as e
    where e.dept_id = d.dept_id
    and e.mgr_id is null)

UNIQUE
```
Query #5

- Find all managers who supervise at least one employee.
Query #5

Find all managers who supervise at least one employee.

```sql
select e.name
from employee as e
where exists (  
    select *
    from employee as e2
    where e2.mgr_id = e.emp_id
  )
```
Query #6

- Find the names of all employees and their manager.
Query #6

- Find the names of all employees and their manager.

```sql
select e.name as emp,
       m.name as mgr
from employee as e
join employee as m
     on m.emp_id = e.mgr_id
```

PROBLEM?
Query #6

Find the names of all employees and their manager.

<table>
<thead>
<tr>
<th>emp_id</th>
<th>name</th>
<th>dept_id</th>
<th>mgr_id</th>
<th>salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>57</td>
<td>Alex</td>
<td>1</td>
<td>-</td>
<td>84000</td>
</tr>
<tr>
<td>23</td>
<td>Jen</td>
<td>1</td>
<td>57</td>
<td>72000</td>
</tr>
<tr>
<td>85</td>
<td>Max</td>
<td>3</td>
<td>19</td>
<td>51000</td>
</tr>
<tr>
<td>44</td>
<td>Tim</td>
<td>2</td>
<td>37</td>
<td>63000</td>
</tr>
</tbody>
</table>
Query #6

Find the names of all employees and their manager.

```
select e.name as emp,
       m.name as mgr
from employee as e
join employee as m
    on m.emp_id = e.mgr_id
union
select e.name,
       'n/a'
from employee as e
where e.mgr_id is null
```
Query #6

Find the names of all employees and their manager.

```
select e.name as emp,
       m.name as mgr
from employee as e
left outer join employee as m
       on m.emp_id = e.mgr_id
```

LEFT OUTER JOIN
Find all departments where the total salary is greater than the average total salary for all departments.
Query #7

Find all departments where the total salary is greater than the average total salary for all departments.

with dept_ttl as (  
    select dept_id, sum(salary) as dttl  
    from employee  
    group by dept_id),
with dept_avg as (  
    select avg(dttl) as davg  
    from dept_ttl)  
select dept_id  
from dept_ttl as t1,dept_avg as t2  
where t1.dttl > t2.davg

WITH CLAUSE
Modifying a Table

- **insert** – add new records
- **delete** – remove some (or all) records
- **update** – change some (or all) existing records
Add a new record to the EMPLOYEE table:

```sql
insert into employee(id,name,dept_id,mgr_id,salary)
values(99,'Amy',2,16,95000)
```
Add a new record to the EMPLOYEE table:

```
insert into employee
values(99,'Amy',2,16,95000)
```
Add a new record to the EMPLOYEE table:

```sql
insert into employee
values(99,'Amy',2,null,95000)
```
Insert

- Add all managers to the new MANAGER table:

  ```sql
  insert into manager
  select *
  from employee
  where mgr_id is null
  ```
Delete

Delete all employees from the EMPLOYEE table:

delete from employee
Delete employee #44:

    delete from employee
    where emp_id = 44
Delete

Delete all employees in the Sales department:

```sql
delete from employee
where dept_id = (  
    select dept_id
    from department
    where name = 'Sales'
)
```
Delete

Delete all employees whose salary is greater than the average salary:

delete from employee
where salary > (select avg(salary) from employee)

PROBLEM?
Make employee #23 a manager:

update employee
set mgr_id = null
where emp_id = 23
Update

- Give all employees a 5% raise:

  ```sql
  update employee
  set salary = salary * 1.05
  ```
Give all employees with a salary greater than $100,000 a 3% raise and all others a 5% raise:

```sql
update employee
set salary = salary * 1.05
where salary <= 100000

update employee
set salary = salary * 1.03
where salary > 100000
```

PROBLEM?
Update

Give all employees with a salary greater than $100,000 a 3% raise and all others a 5% raise:

```sql
update employee
set salary = case
  when salary > 100000
    then salary * 1.03
  else
    salary * 1.05
end
```

CASE STATEMENT
If you disliked the lecture, please forward all complaints to: