Tutorial 4

Eng. Ahmed Sherif

Postgres

1. Create the Employee table using Postgres database. The table should have the following columns:

- ID
- firstName
- middleName
- lastName
- country
- salary
- birthDate

Employees should be uniquely identified by their IDs. No null entries are allowed in the table.

2. Insert seven records into the Employee table.

3. Write sql statement that delete table Employee

Solution
Ex 4.1 no 1

```
create table Employee(

ID serial primary key,
firstName text not null,
middleName text not null,
lastName text not null,
country text not null,
salary integer not null,
birthDate date not null -- yyyy-mm-dd
);
```

date : must be entered as yyyy-mm-dd
text : datatype is given to string
not null : user must enter a value (required in google form)

Postgres

1. Create the Employee table using Postgres database. The table should have the following columns:

- ID
- firstName
- middleName
- lastName
- country
- salary
- birthDate

Employees should be uniquely identified by their IDs. No null entries are allowed in the table.

2. Insert seven records into the Employee table.

3. Write sql statement that delete table Employee

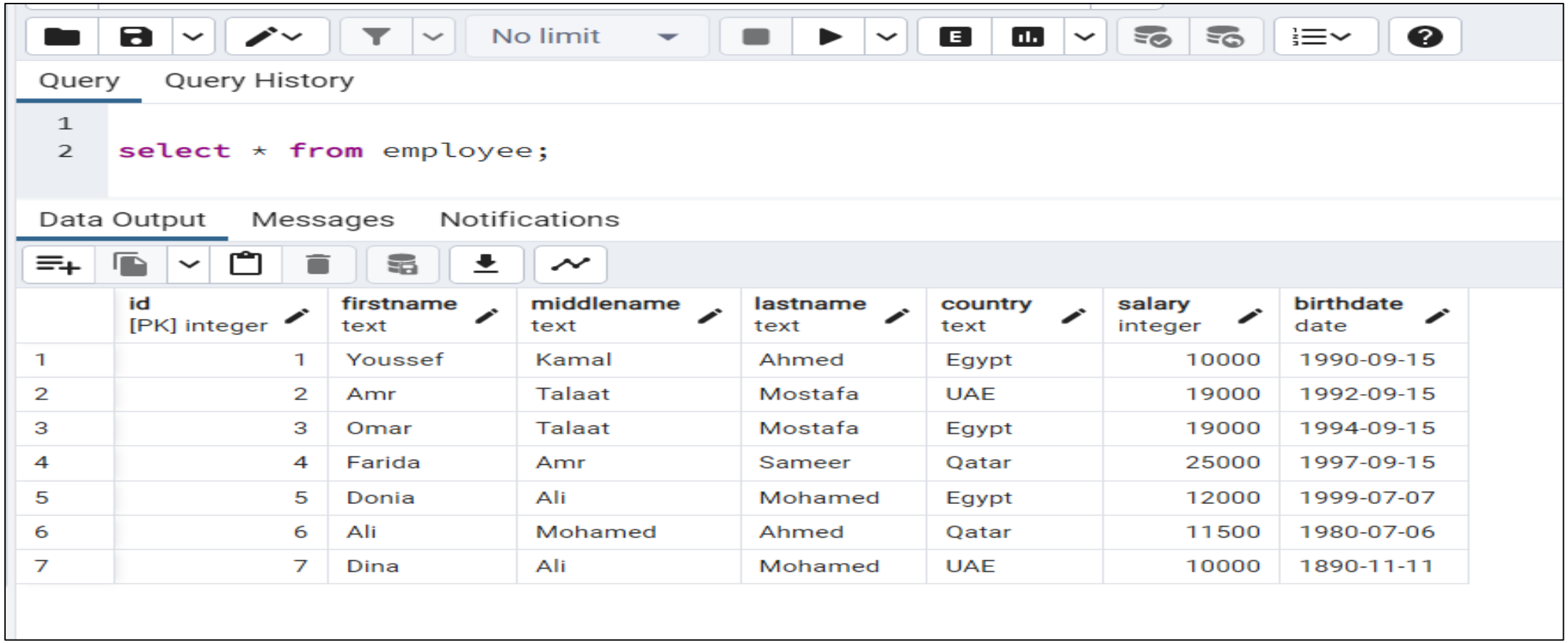
Solution : Ex 4.1 no 2

```
insert into Employee(firstName, middleName, lastName, country, salary , birthDate)
values('Youssef', 'Kamal' , 'Ahmed', 'Egypt' , 10000 , '1990-09-15');

insert into Employee(firstName, middleName, lastName, country, salary , birthDate)
values('Amr', 'Talaat' , 'Mostafa', 'UAE' , 19000, '1992-09-15');

insert into Employee(firstName, middleName, lastName, country, salary , birthDate)
values('Omar', 'Talaat' , 'Mostafa', 'Egypt' , 19000, '1994-09-15');

insert into Employee(firstName, middleName, lastName, country, salary , birthDate)
values('Farida', 'Amr' , 'Sameer', 'Qatar' , 25000, '1997-09-15');

insert into Employee(firstName, middleName, lastName, country, salary , birthDate)
values('Donia', 'Ali' , 'Mohamed', 'Egypt' , 12000, '1999-07-07');

insert into Employee(firstName, middleName, lastName, country, salary , birthDate)
values('Ali', 'Mohamed' , 'Ahmed', 'Qatar' , 11500, '1980-07-06');

insert into Employee(firstName, middleName, lastName, country, salary , birthDate)
values('Dina', 'Ali' , 'Mohamed', 'UAE' , 10000, '1890-11-11');
```

3. drop table Employee;

Postgres

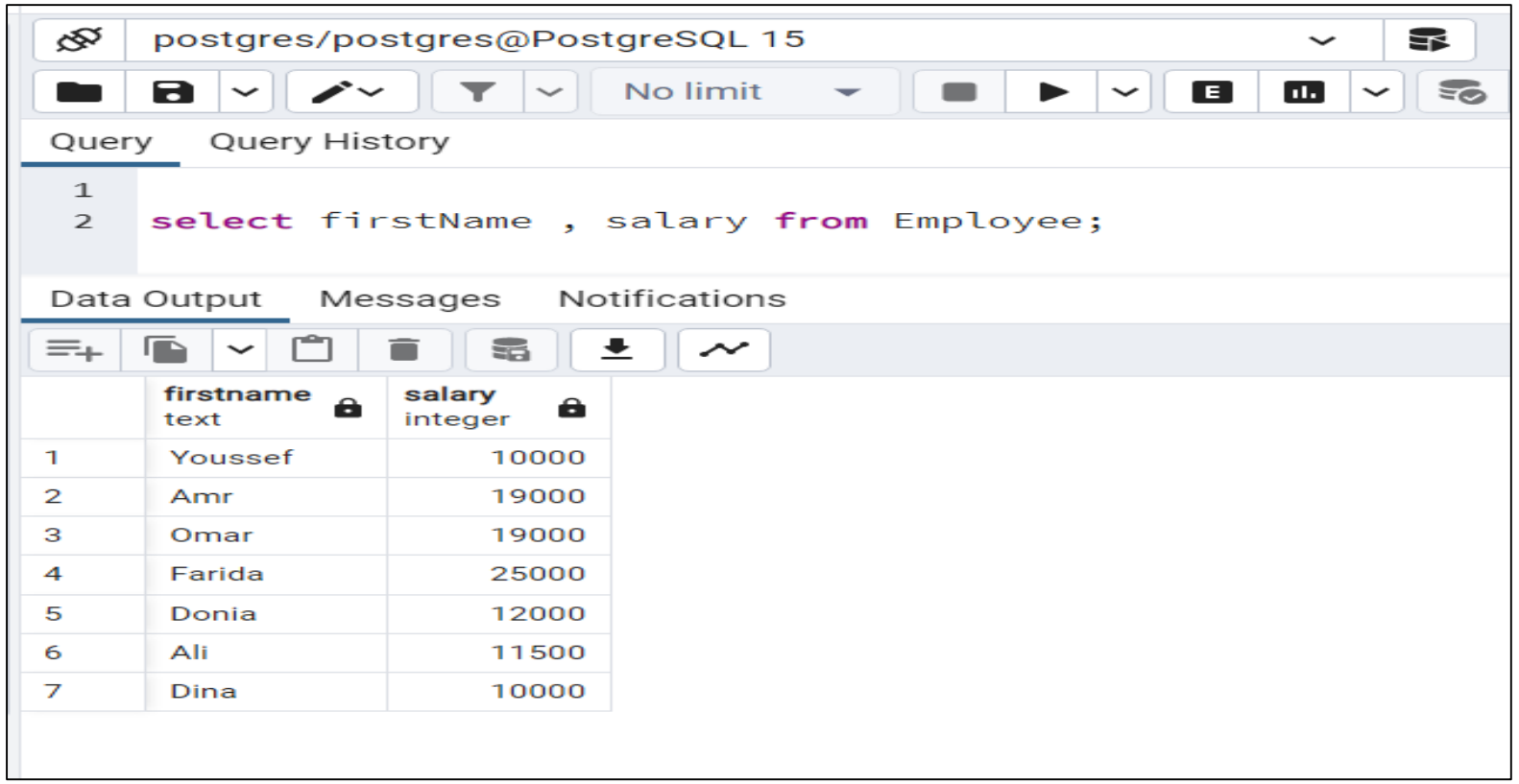
Exercise 4-2
Using the employee table from Ex 4.1
1. Show all the details about the employees



	id [PK] integer	firstname text	middlename text	lastname text	country text	salary integer	birthdate date
1	1	Youssef	Kamal	Ahmed	Egypt	10000	1990-09-15
2	2	Amr	Talaat	Mostafa	UAE	19000	1992-09-15
3	3	Omar	Talaat	Mostafa	Egypt	19000	1994-09-15
4	4	Farida	Amr	Sameer	Qatar	25000	1997-09-15
5	5	Donia	Ali	Mohamed	Egypt	12000	1999-07-07
6	6	Ali	Mohamed	Ahmed	Qatar	11500	1980-07-06
7	7	Dina	Ali	Mohamed	UAE	10000	1890-11-11

Postgres

Exercise 4-2
Using the employee table from Ex 4.1
2. Show the first name and salary of all the employees



	firstname text	salary integer
1	Youssef	10000
2	Amr	19000
3	Omar	19000
4	Farida	25000
5	Donia	12000
6	Ali	11500
7	Dina	10000

Postgres

Exercise 4-2
Using the employee table from Ex 4.1
3. Show distinct countries for the employees and sort them by country.

Query | Query History

```
1
2  select distinct country from Employee order by country;
```

Data Output | Messages | Notifications

	country text
1	Egypt
2	Qatar
3	UAE

Postgres

Exercise 4-2
Using the employee table from Ex 4.1
4. Show all the details about the employees who have a salary between 10000 and 15000.

Query | Query History

```
1
2  select * from Employee where salary >= 10000 and salary <= 15000;
```

Data Output | Messages | Notifications

	id [PK] integer	firstname text	middlename text	lastname text	country text	salary integer	birthdate date
1	1	Youssef	Kamal	Ahmed	Egypt	10000	1990-09-15
2	5	Donia	Ali	Mohamed	Egypt	12000	1999-07-07
3	6	Ali	Mohamed	Ahmed	Qatar	11500	1980-07-06
4	7	Dina	Ali	Mohamed	UAE	10000	1890-11-11

Postgres

Exercise 4-2
Using the employee table from Ex 4.1
5. Show the details of all employees who are having a salary more than 10000 and country is not Egypt.

Query | Query History

```
1
2  select * from Employee where country <> 'Egypt' and salary > 10000;
```

Data Output | Messages | Notifications

	id [PK] integer	firstname text	middlename text	lastname text	country text	salary integer	birthdate date
1	2	Amr	Talaat	Mostafa	UAE	19000	1992-09-15
2	4	Farida	Amr	Sameer	Qatar	25000	1997-09-15
3	6	Ali	Mohamed	Ahmed	Qatar	11500	1980-07-06

Postgres

Exercise 4-2
Using the employee table from Ex 4.1
6. Show all the details about the employees who were born after 11/2/1980 sorted by their country descendingly and within each country sorted by their salary ascendingly.

Query | Query History

```
1
2  select * from Employee where birthDate> '1980-2-11' order by country desc ,salary ;
```

Data Output | Messages | Notifications

	id [PK] integer	firstname text	middlename text	lastname text	country text	salary integer	birthdate date
1	2	Amr	Talaat	Mostafa	UAE	19000	1992-09-15
2	6	Ali	Mohamed	Ahmed	Qatar	11500	1980-07-06
3	4	Farida	Amr	Sameer	Qatar	25000	1997-09-15
4	1	Youssef	Kamal	Ahmed	Egypt	10000	1990-09-15
5	5	Donia	Ali	Mohamed	Egypt	12000	1999-07-07
6	3	Omar	Talaat	Mostafa	Egypt	19000	1994-09-15

Postgres

Exercise 4-2
Using the employee table from Ex 4.1
11. For all the Employees who earn between 10000 and 12000 update their first name to Ali, their last name to Mohammed and their birth date to 7/7/1999.

Query | Query History

```
1  UPDATE Employee
2  SET firstName = 'Ali', lastName = 'Mohammed', birthDate = '1999-7-7'
3  WHERE salary BETWEEN 10000 AND 12000;
4
5  select * from Employee;
```

Data Output | Messages | Notifications

	id [PK] integer	firstname text	middlename text	lastname text	country text	salary integer	birthdate date
1	2	Amr	Talaat	Mostafa	UAE	19000	1992-09-15
2	3	Omar	Talaat	Mostafa	Egypt	19000	1994-09-15
3	4	Farida	Amr	Sameer	Qatar	25000	1997-09-15
4	1	Ali	Kamal	Mohammed	Egypt	10000	1999-07-07
5	5	Ali	Ali	Mohammed	Egypt	12000	1999-07-07
6	6	Ali	Mohamed	Mohammed	Qatar	11500	1999-07-07
7	7	Ali	Ali	Mohammed	UAE	10000	1999-07-07

Postgres

Exercise 4-2
Using the employee table from Ex 4.1
12. Delete the record of Employee with id of 1.

Query Query History

```
1  delete from employee where id = 1;
2  select * from employee;
```

Data Output Messages Notifications

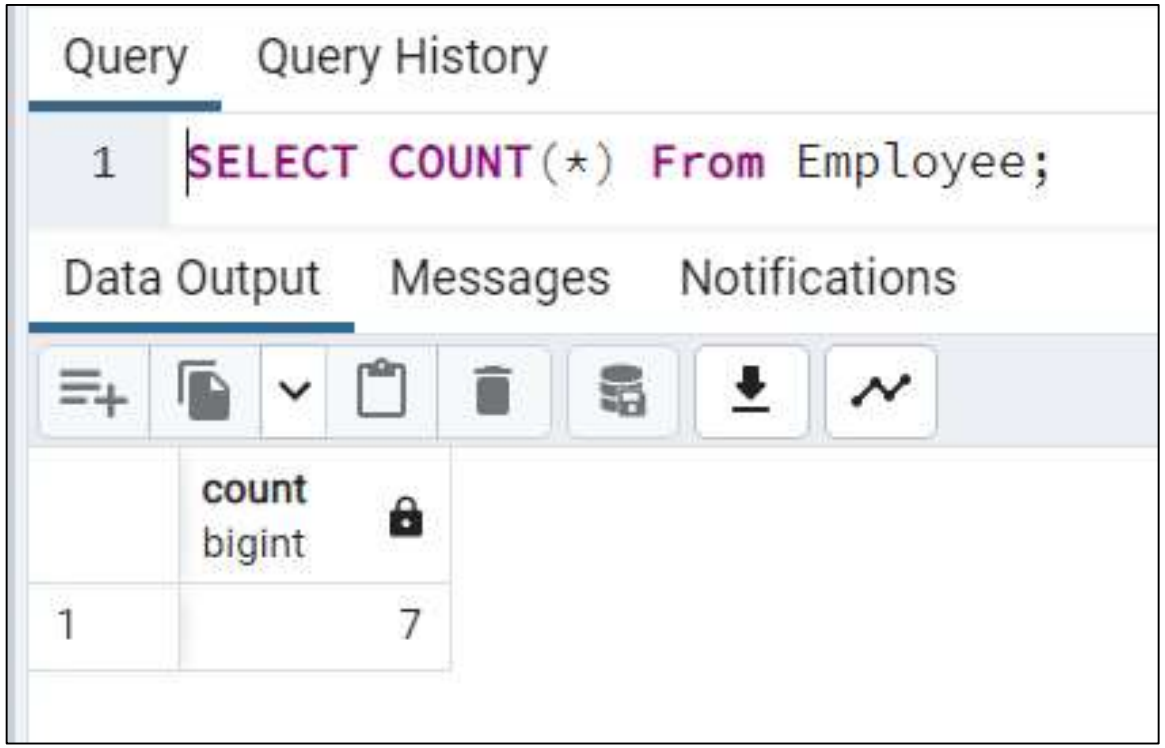
	id [PK] integer	firstname text	middlename text	lastname text	country text	salary integer	birthdate date
1	2	Amr	Talaat	Mostafa	UAE	19000	1992-09-15
2	3	Omar	Talaat	Mostafa	Egypt	19000	1994-09-15
3	4	Farida	Amr	Sameer	Qatar	25000	1997-09-15
4	5	Donia	Ali	Mohamed	Egypt	12000	1999-07-07
5	6	Ali	Mohamed	Ahmed	Qatar	11500	1980-07-06
6	7	Dina	Ali	Mohamed	UAE	10000	1890-11-11

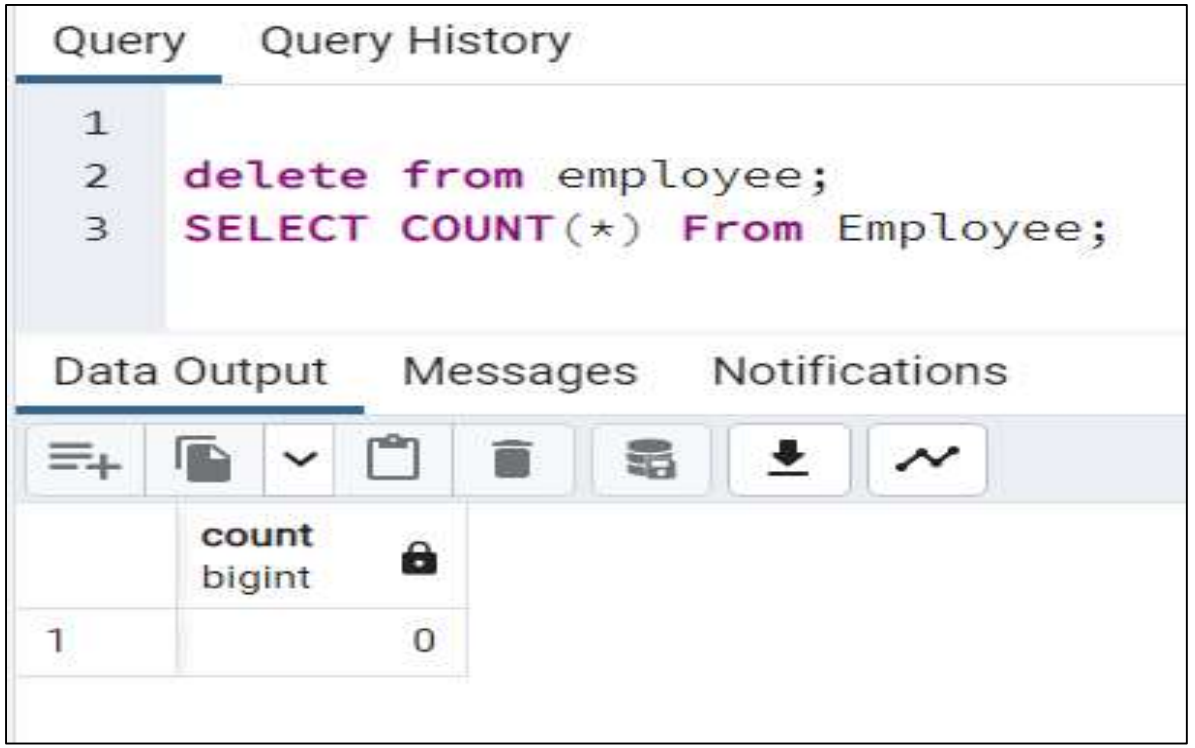
Postgres

Exercise 4-2
Using the employee table from Ex 4.1
13. Delete all the entries in table Employee.

Postgres

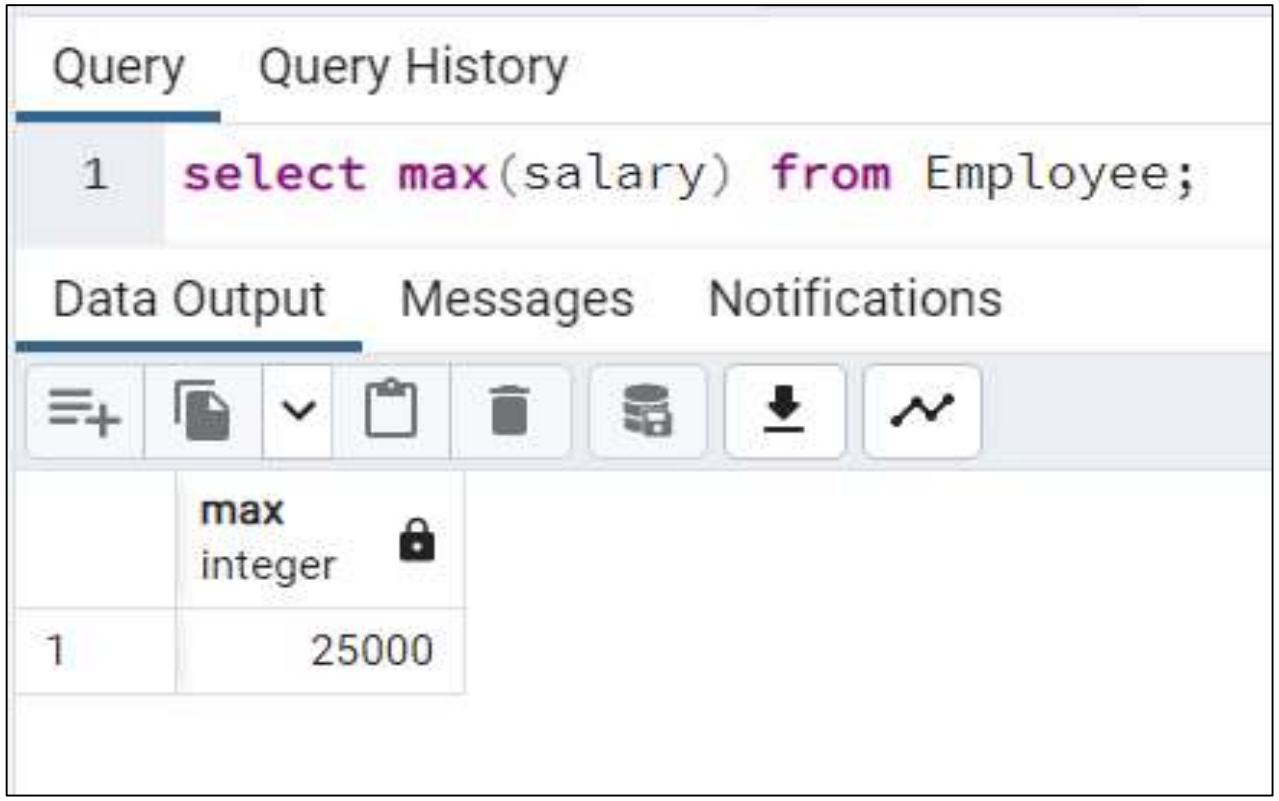
Exercise 4-3
Using the employee table from Ex 4.1
1. Show the number of records in the employee table.





Postgres

Exercise 4-3
Using the employee table from Ex 4.1
2. Show the highest salary of an Employee.



Postgres

Exercise 4-3

Using the employee table from Ex 4.1

3. show the average salary of the Egyptian Employees.

Query Query History

```
1  select avg(salary) from Employee where country = 'Egypt'
```

Data Output Messages Notifications

	avg numeric
1	13666.666666666667

Query Query History

```
1  select * from Employee;
```

Data Output Messages Notifications

	id [PK] integer	firstname text	middlename text	lastname text	country text	salary integer	birthdate date
1	15	Youssef	Kamal	Ahmed	Egypt	10000	1990-09-15
2	16	Amr	Talaat	Mostafa	UAE	19000	1992-09-15
3	17	Omar	Talaat	Mostafa	Egypt	19000	1994-09-15
4	18	Farida	Amr	Sameer	Qatar	25000	1997-09-15
5	19	Donia	Ali	Mohamed	Egypt	12000	1999-07-07
6	20	Ali	Mohamed	Ahmed	Qatar	11500	1980-07-06
7	21	Dina	Ali	Mohamed	UAE	10000	1890-11-11

Postgres

Exercise 4-3
Using the employee table from Ex 4.1
4. Show the average salary for every country and sorted by countries

Query | Query History

```
1  SELECT country , AVG(salary) FROM Employee GROUP BY country order by country;
```

Data Output | Messages | Notifications

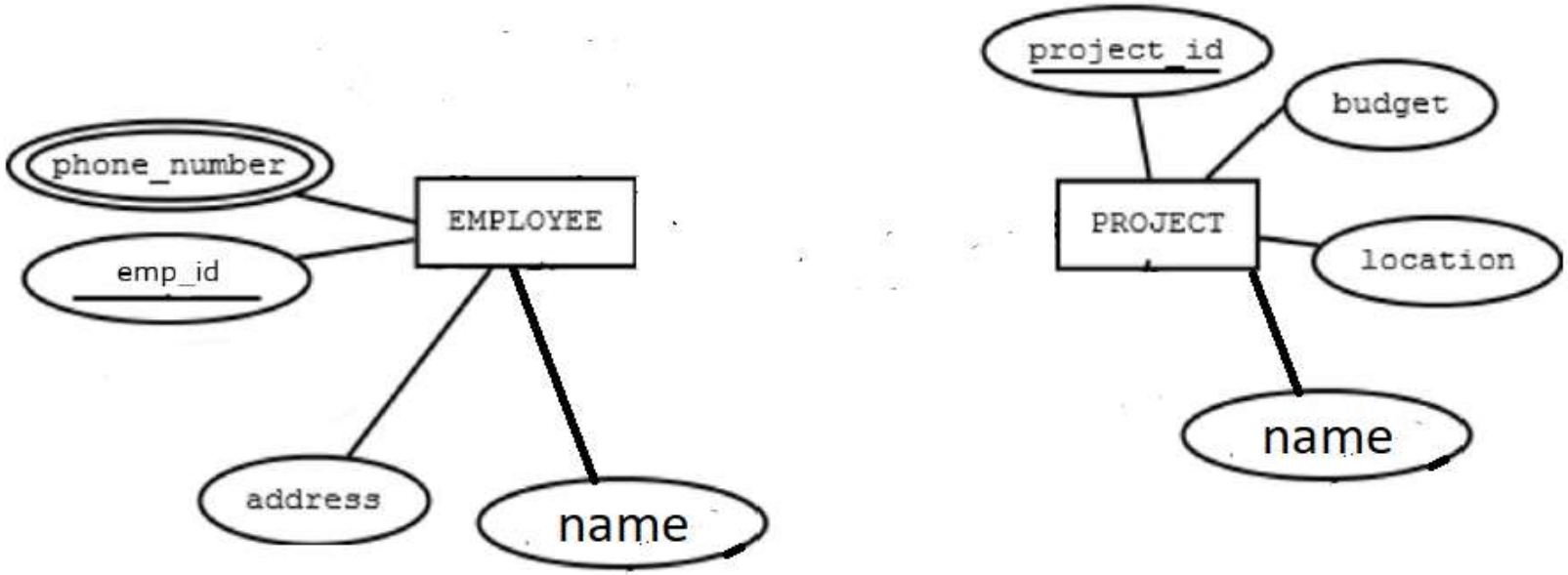
	country text	avg numeric
1	Egypt	13666.666666666667
2	Qatar	18250.000000000000
3	UAE	14500.0000000000000000

Postgres

Ex 4.5

We need to store some information about a company. It contains different employees. Each employee has name. They also have an address and phone number(s). The company has different projects. Each project has a name, location and budget. Employees are involved on at least one project. Each project has a manager. Design an ERD

Solution

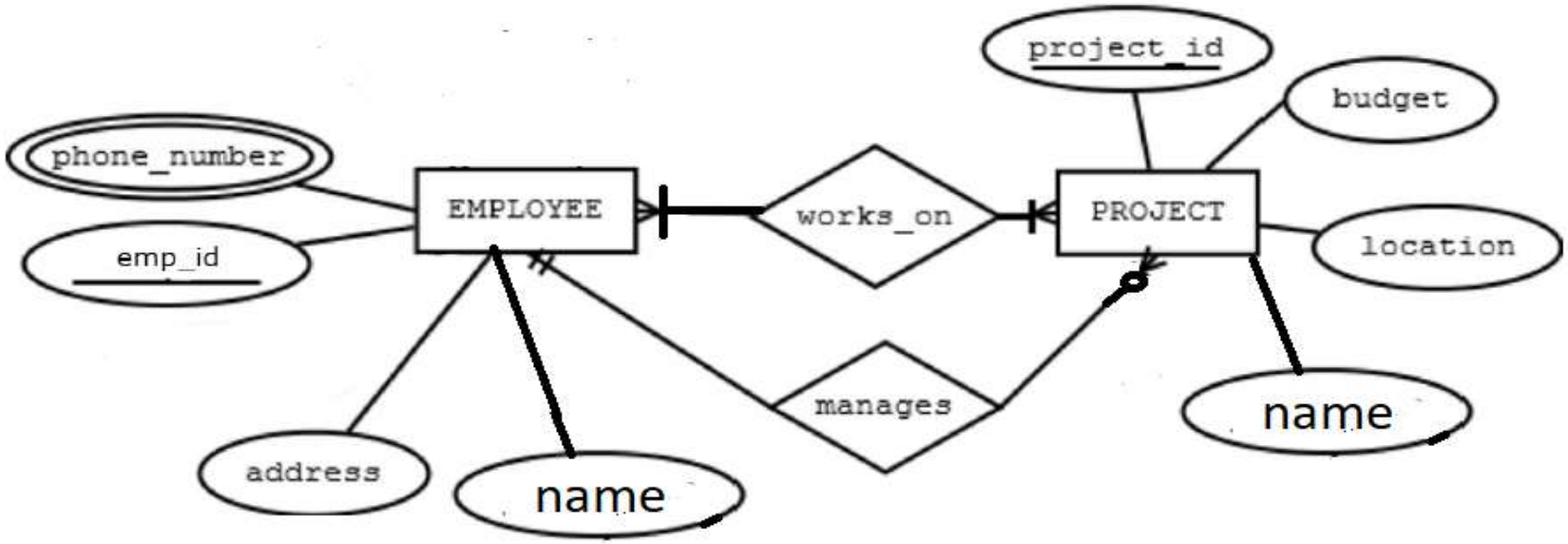


Postgres

Ex 4.5

We need to store some information about a company. It contains different employees. Each employee has first name and last name. They also have an address and phone number(s). The company has different projects. Each project has a name, location and budget. Employees are involved on at least one project. Each project has a manager.

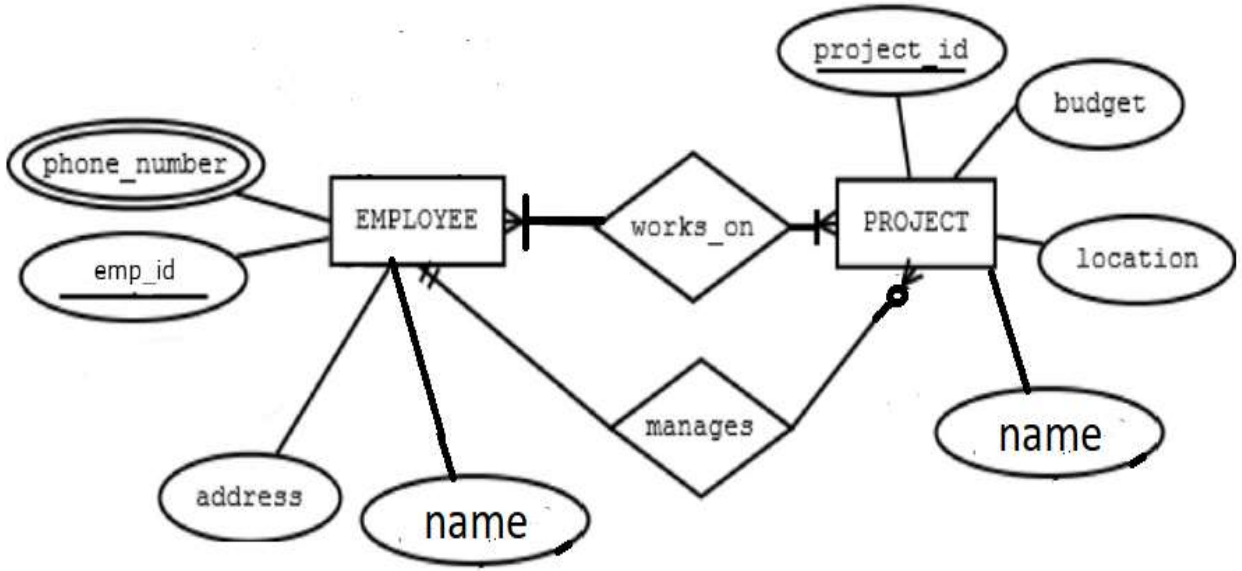
Solution
Design an ERD



Postgres

Solution
create tables schema from your ERD diagram

```
1. create table Employee(
emp_id serial primary key,
name text not null,
address text not null,
);

2. create table project(
project_id serial primary key,
name text not null,
budjet integer not null,
location text not null,
emp_id integer not null,
foreign key(emp_id) references employee(emp_id)
);
```



In one to many relation

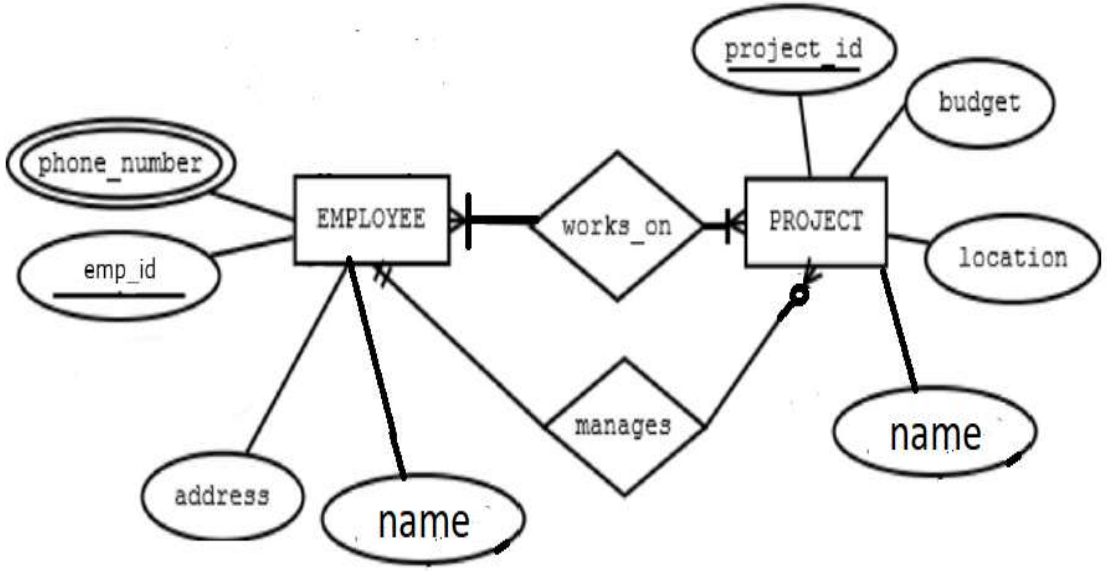
1. Take primary key from one relation (Employee)
2. Add new column in many relation (Project)
3. This new column is foreign key and reference the primary key of one relation (Employee)

Postgres

Solution

create tables schema from your ERD diagram

3. create table works_on(
emp_id integer,
project_id integer,
foreign key(emp_id) references employee(emp_id),
foreign key(project_id) references project(project_id),
primary key(emp_id , project_id)
);



In many to many relation

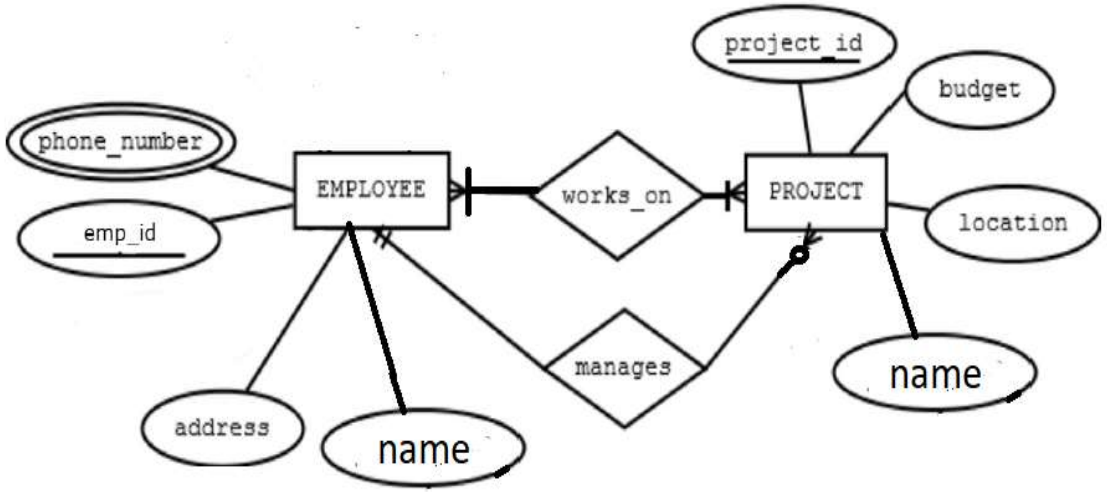
1. **Create new table**
2. **Take primary key of both tables**
3. **The primary key of new table consists of primary key of both tables**

Postgres

Solution

create tables schema from your ERD diagram

```
4. create table employee_phone(
emp_id integer,
number text,
foreign key(emp_id) references employee(emp_id),
primary key(emp_id , number)
);
```



In multivariant Attribute (donated by double circle)
Ex 6.5 shows that employee has more than one phone number

1. **Create new table**
2. **Take primary key of table (Employee)**
3. **Added it in new table as a foreign key and reference the primary key of Employee**
4. **The primary key of table consists of foreign key and multivariant attribute**

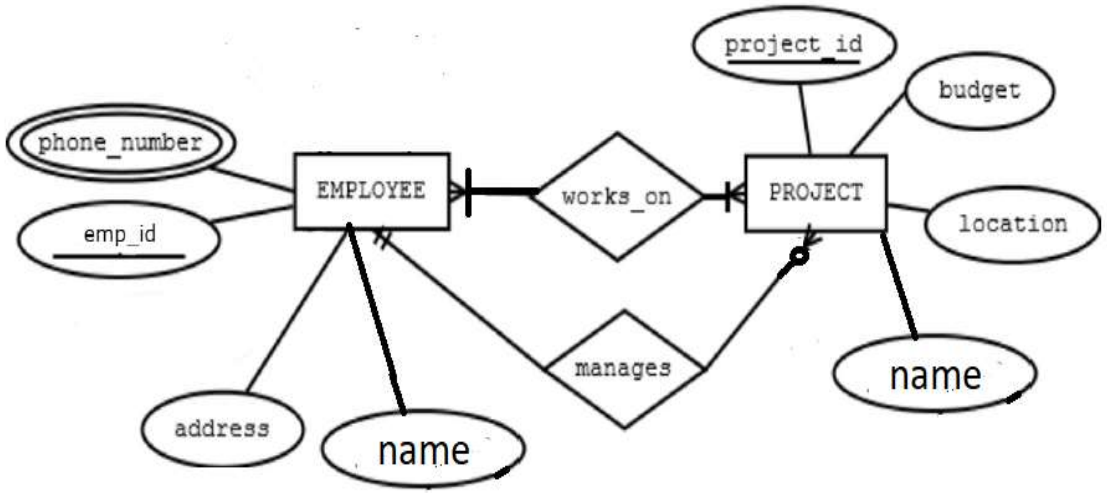
Postgres

Query | Query History

```
1  select * from employee_phone;
```

Data Output | Messages | Notifications

	emp_id [PK] integer	number [PK] text
1	1	010155
2	1	010160



Postgres

Query Query History

```
11  select * from employee;
12
```

Data Output Messages Notifications

	emp_id [PK] integer	name text	address text
1	1	Karim	Cairo
2	2	Adham	Giza
3	3	Zeyad	Cairo

Query Query History

```
34  select project_id,name,location,budjet,emp_id from project;
35
```

Data Output Messages Notifications

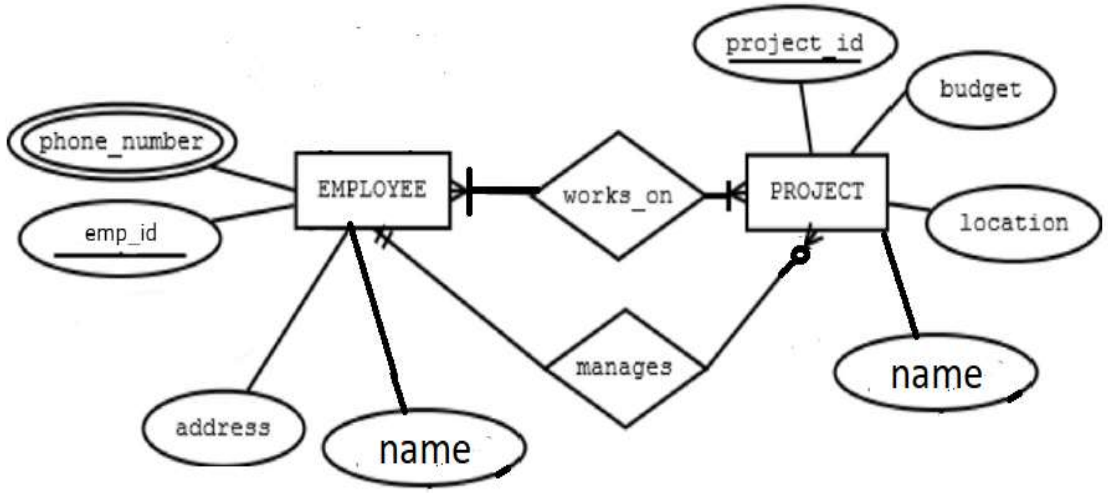
	project_id [PK] integer	name text	location text	budjet integer	emp_id integer
1	1001	P1	Luxor	10000	3
2	1002	P2	Giza	10000	3
3	1003	P3	Alexandria	10000	2

Query Query History

```
59
60  select * from works_on;
61
```

Data Output Messages Notifications

	emp_id [PK] integer	project_id [PK] integer
1	1	1001
2	2	1001
3	3	1001
4	3	1002
5	1	1002
6	2	1003



Postgres

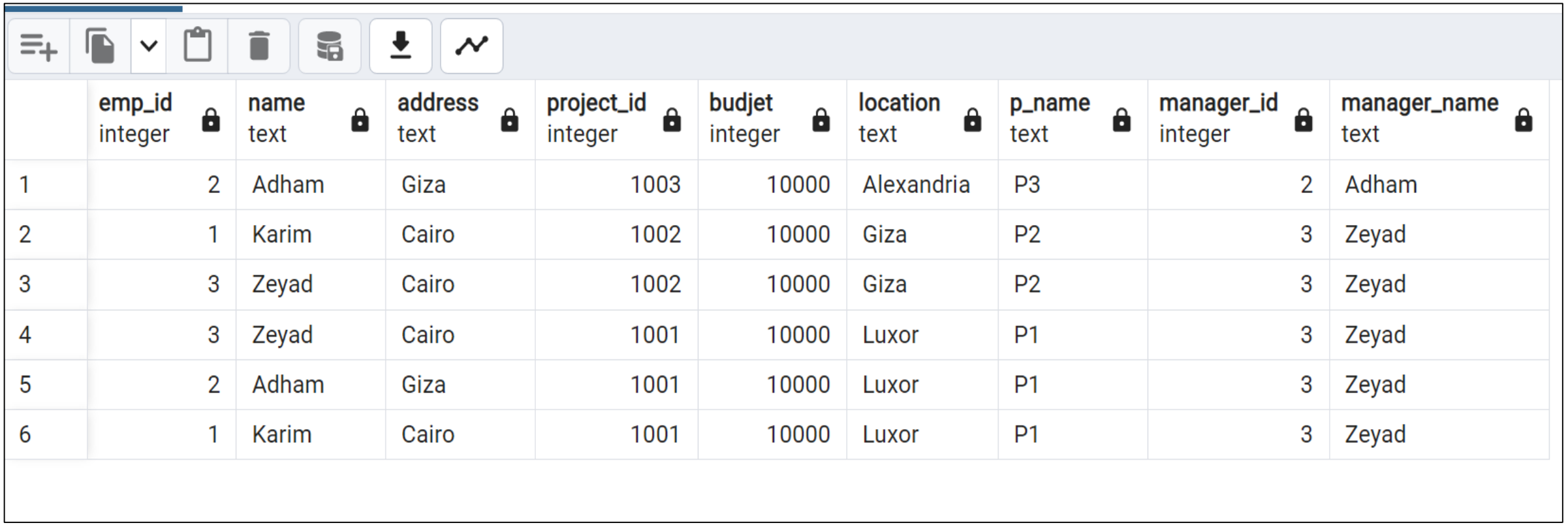
Show all details of employees that are working on project and show all information about projects and their manager.

Query | Query History

```
1  select e.* , p.project_id , p.budjet , p.location ,
2  p.name as p_name , p.emp_id as manager_id,
3  en.name as manager_name from works_on wo
4  inner join project p on p.project_id = wo.project_id
5  inner join employee e on e.emp_id = wo.emp_id
6  inner join employee en on en.emp_id = p.emp_id;
```

Postgres

Show all details of employees that are working on project and show all information about projects and their manager.



	emp_id integer	name text	address text	project_id integer	budjet integer	location text	p_name text	manager_id integer	manager_name text
1	2	Adham	Giza	1003	10000	Alexandria	P3	2	Adham
2	1	Karim	Cairo	1002	10000	Giza	P2	3	Zeyad
3	3	Zeyad	Cairo	1002	10000	Giza	P2	3	Zeyad
4	3	Zeyad	Cairo	1001	10000	Luxor	P1	3	Zeyad
5	2	Adham	Giza	1001	10000	Luxor	P1	3	Zeyad
6	1	Karim	Cairo	1001	10000	Luxor	P1	3	Zeyad