**SQL EXERCISE**

**#TASK 1**

**Creating Employee DB**

*1. create employee(emp\_id,employee\_name,department\_name,dept\_id,location\_id,salary)department(dept\_id,department\_name),locations (location\_id, location\_name) tables with relevant attributes.*

Ans:-

CREATE TABLE employee(

emp\_id int,

employee\_name varchar(30),

department\_name varchar(30),

location varchar(30),

salary int

);

CREATE TABLE department(

dept\_id int,

department\_name char

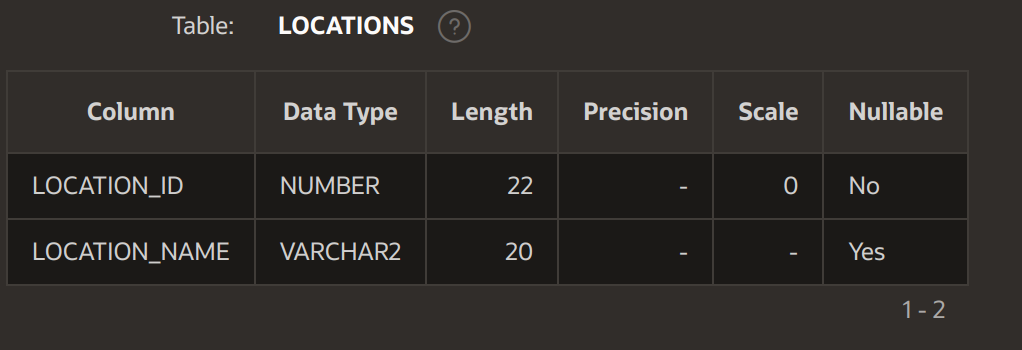
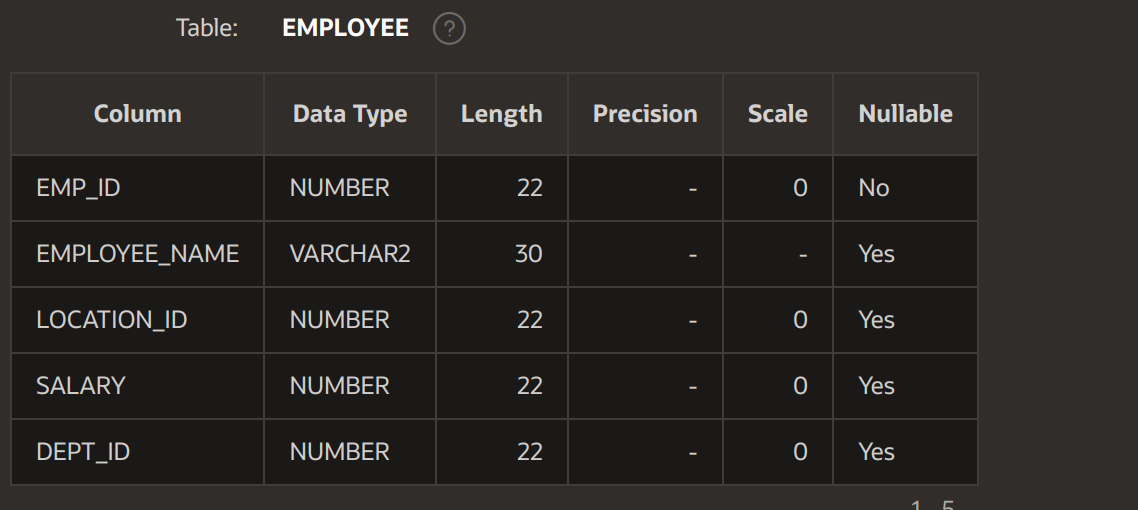
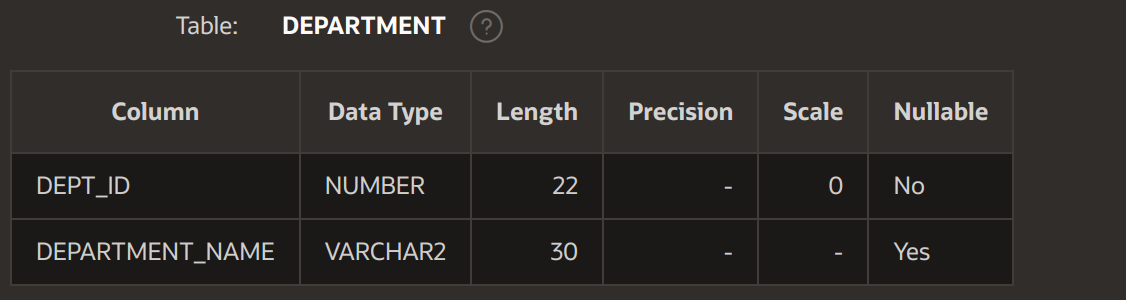
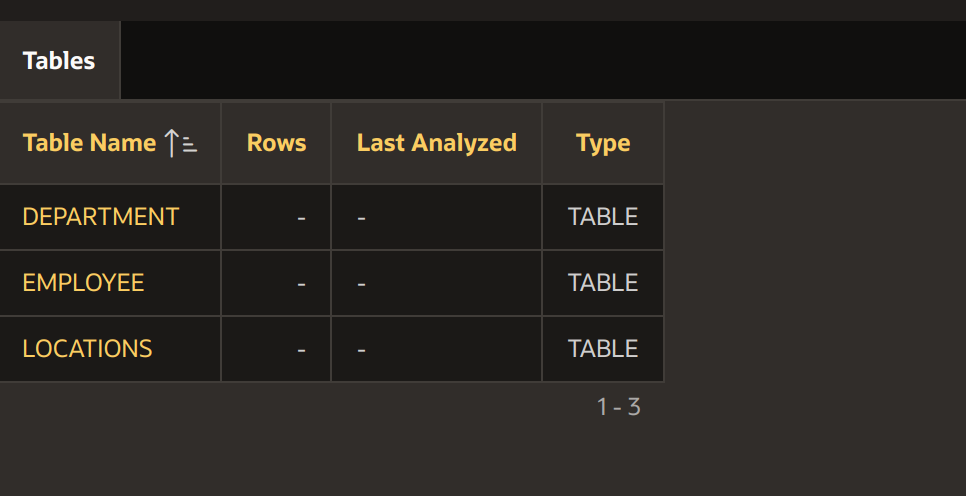
);

CREATE TABLE locations(

location\_id int,

location\_name char

);



*2. create primary key on each table and foreign keys (location->department, department->employee)*

Ans:-

ALTER TABLE employee

ADD CONSTRAINT emp\_id

PRIMARY KEY(emp\_id);

ALTER TABLE department

ADD CONSTRAINT dept\_id

PRIMARY KEY(dept\_id);

ALTER TABLE locations

ADD CONSTRAINT location\_id

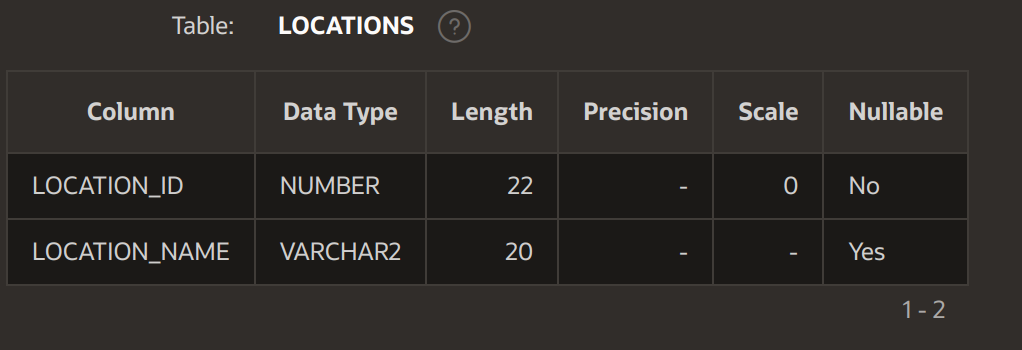
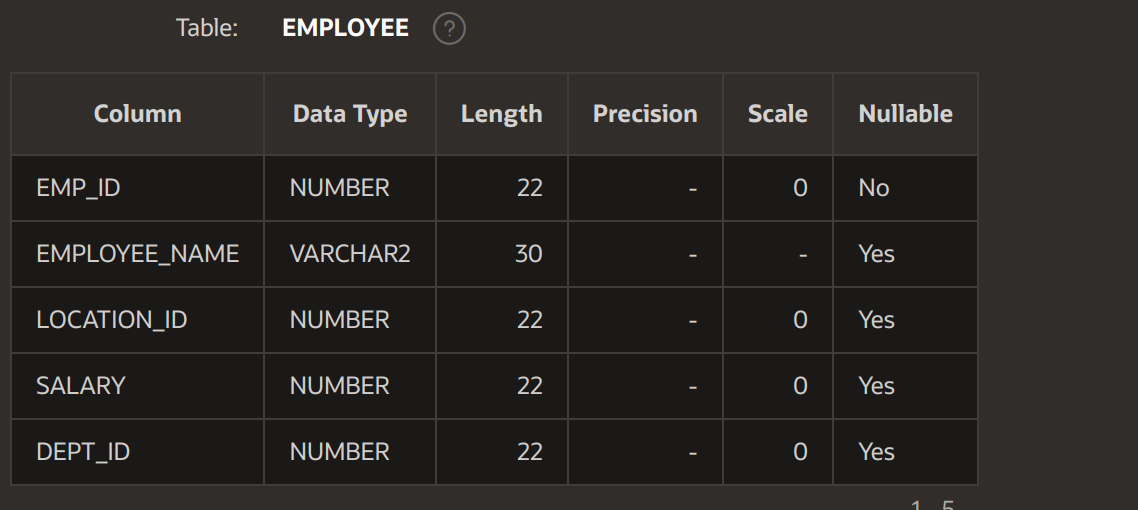
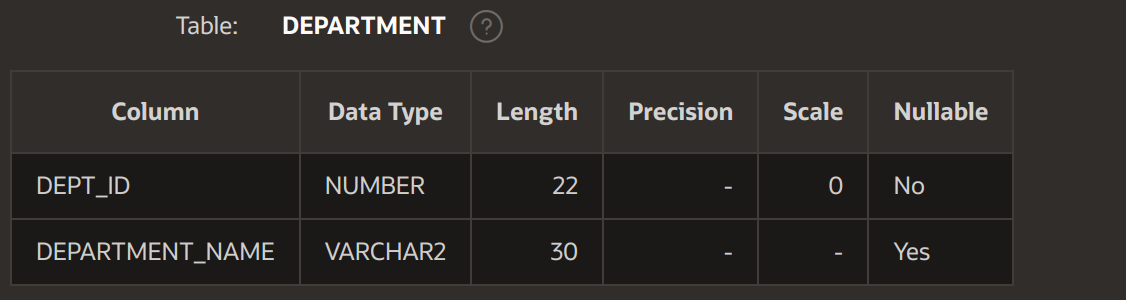
PRIMARY KEY(location\_id);

ALTER TABLE employee

ADD FOREIGN KEY (dept\_id) REFERENCES department(dept\_id);

ALTER TABLE employee

ADD FOREIGN KEY (location\_id) REFERENCES locations(location\_id);



*3. insert 20 employees data, 4 departments data, 2 locations data.*

Ans:-

INSERT INTO locations

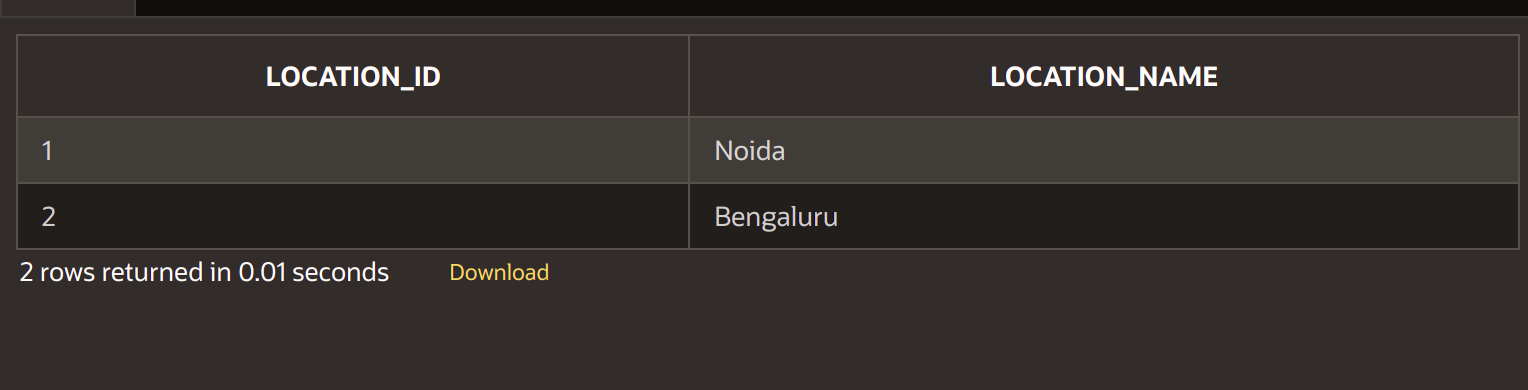
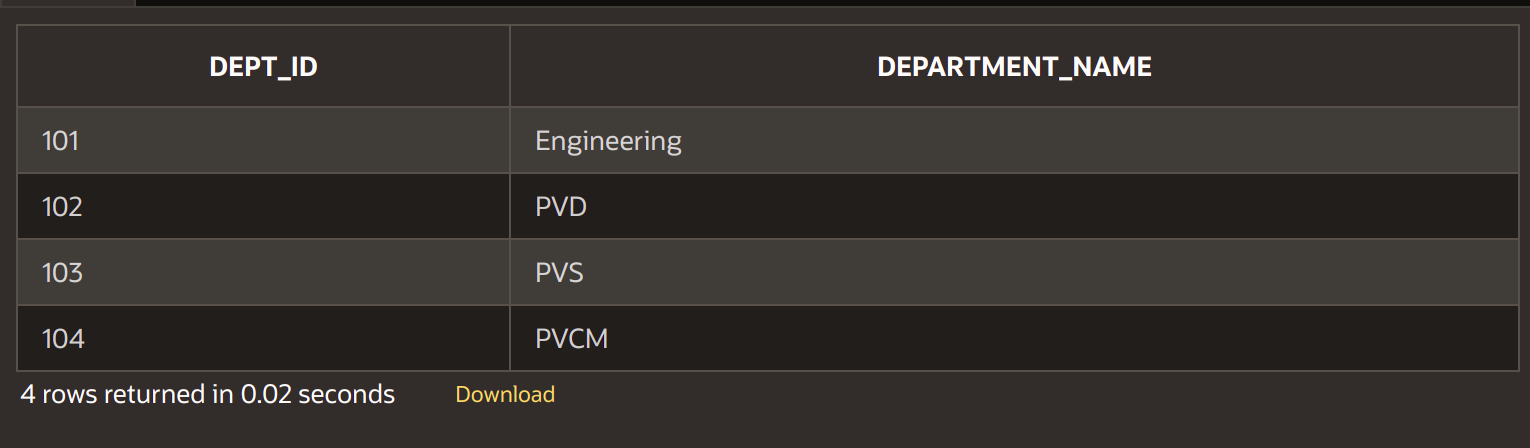
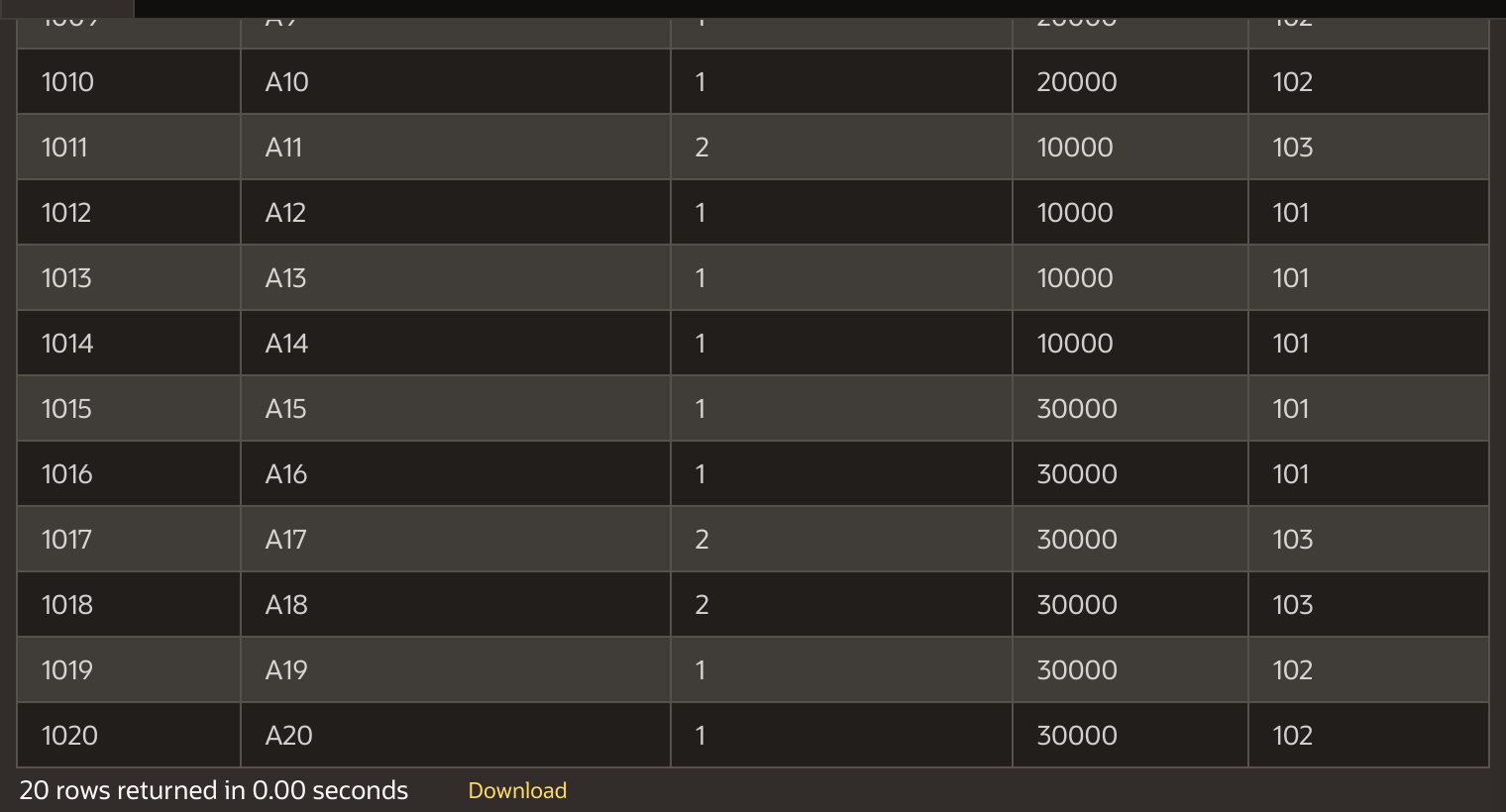
VALUES (1, 'Noida');

INSERT INTO Department

VALUES (101, 'Engineering');

INSERT INTO employee

VALUES (1001 , 'A1' , 'Engineeering' , 1 , 20000 , 101) ;



*4. write below queries:*

*a) display all employees names and their department names*

Ans:-

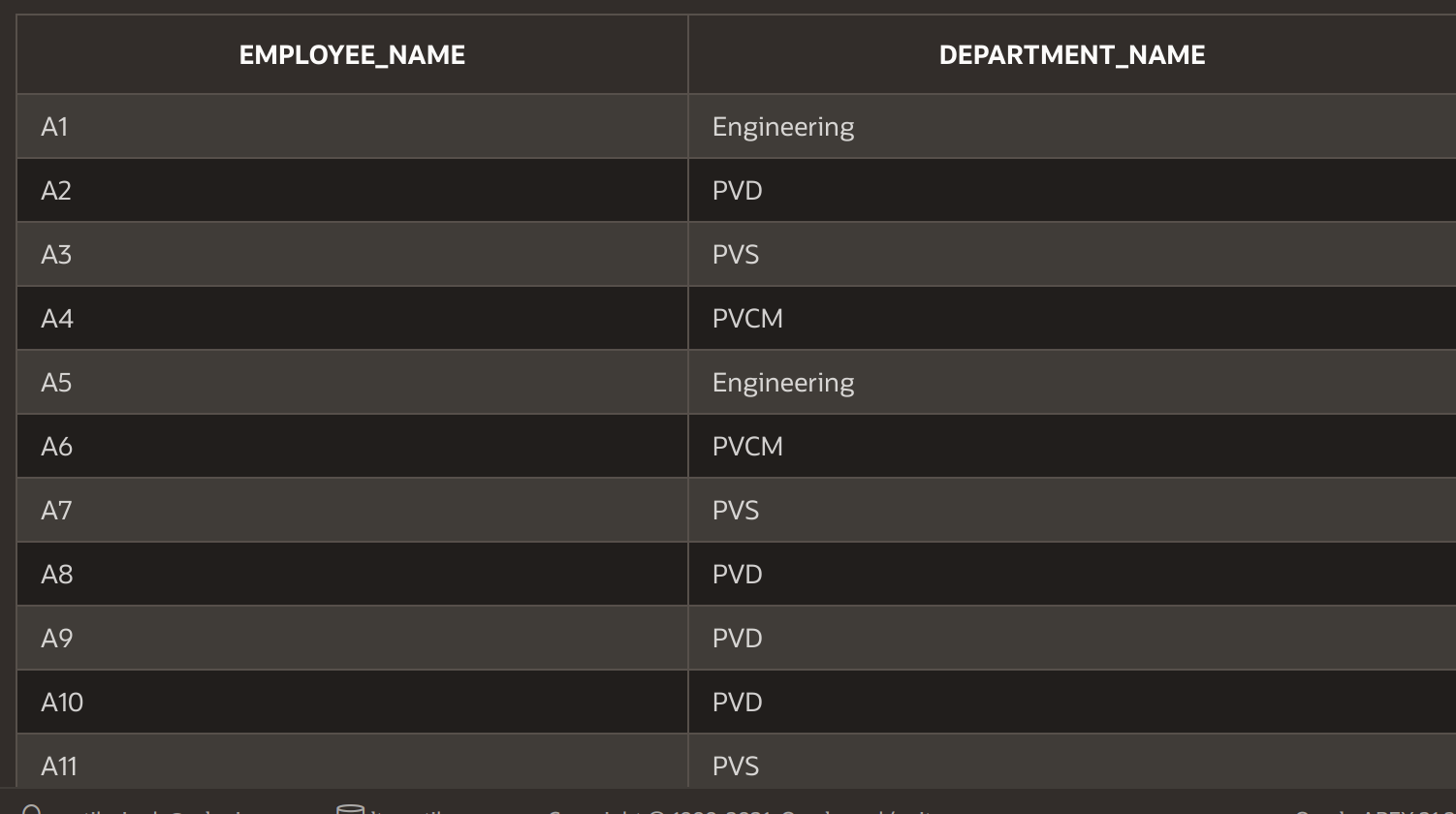
SELECT employee.employee\_name, department.department\_name

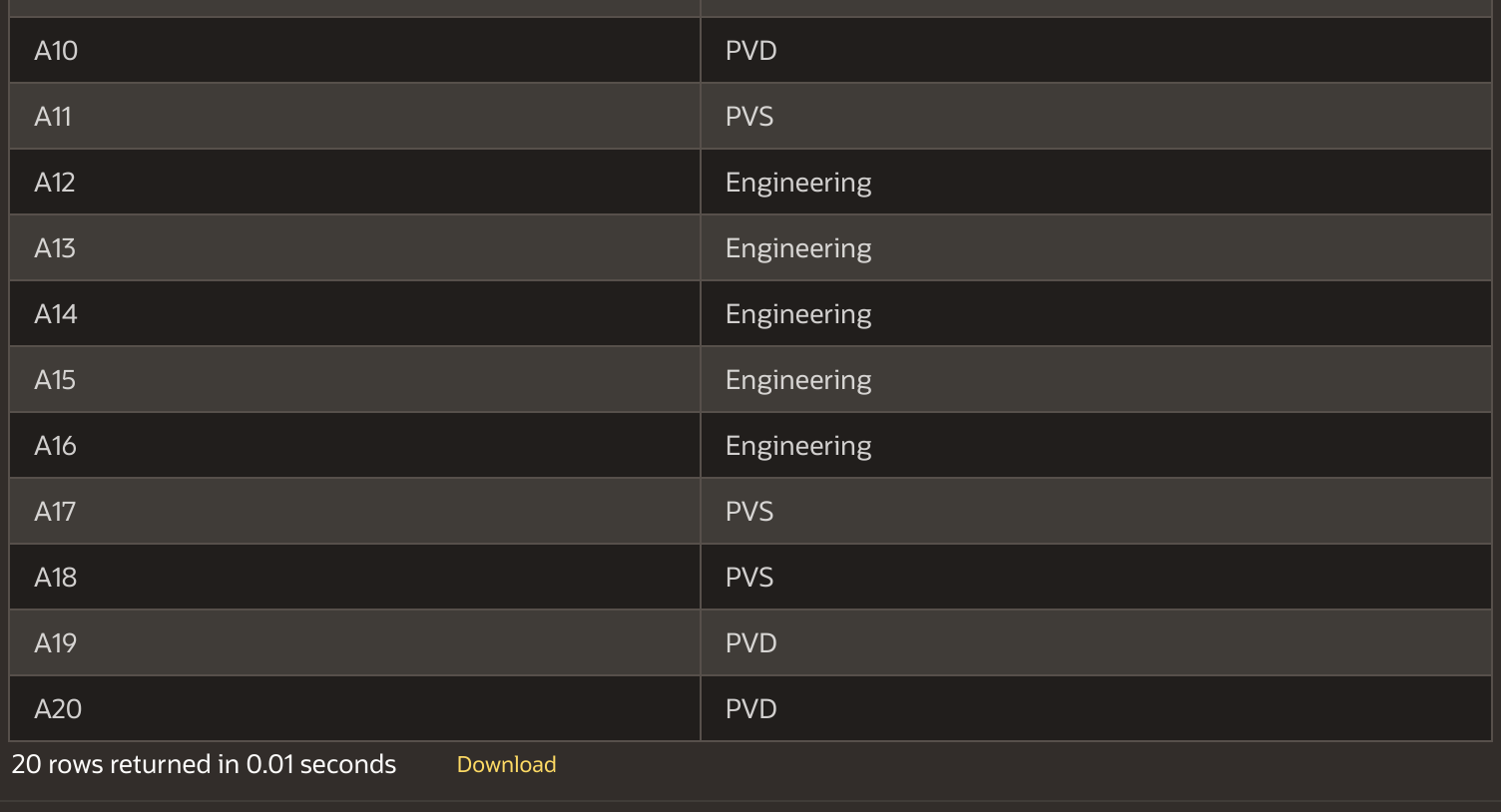
FROM employee

INNER JOIN department

ON employee.dept\_id = department.dept\_id

ORDER BY employee.employee\_name;





*b) display all location\_name, department\_name, employee\_name, salary for all matching rows from 3 tables.*

Ans:-

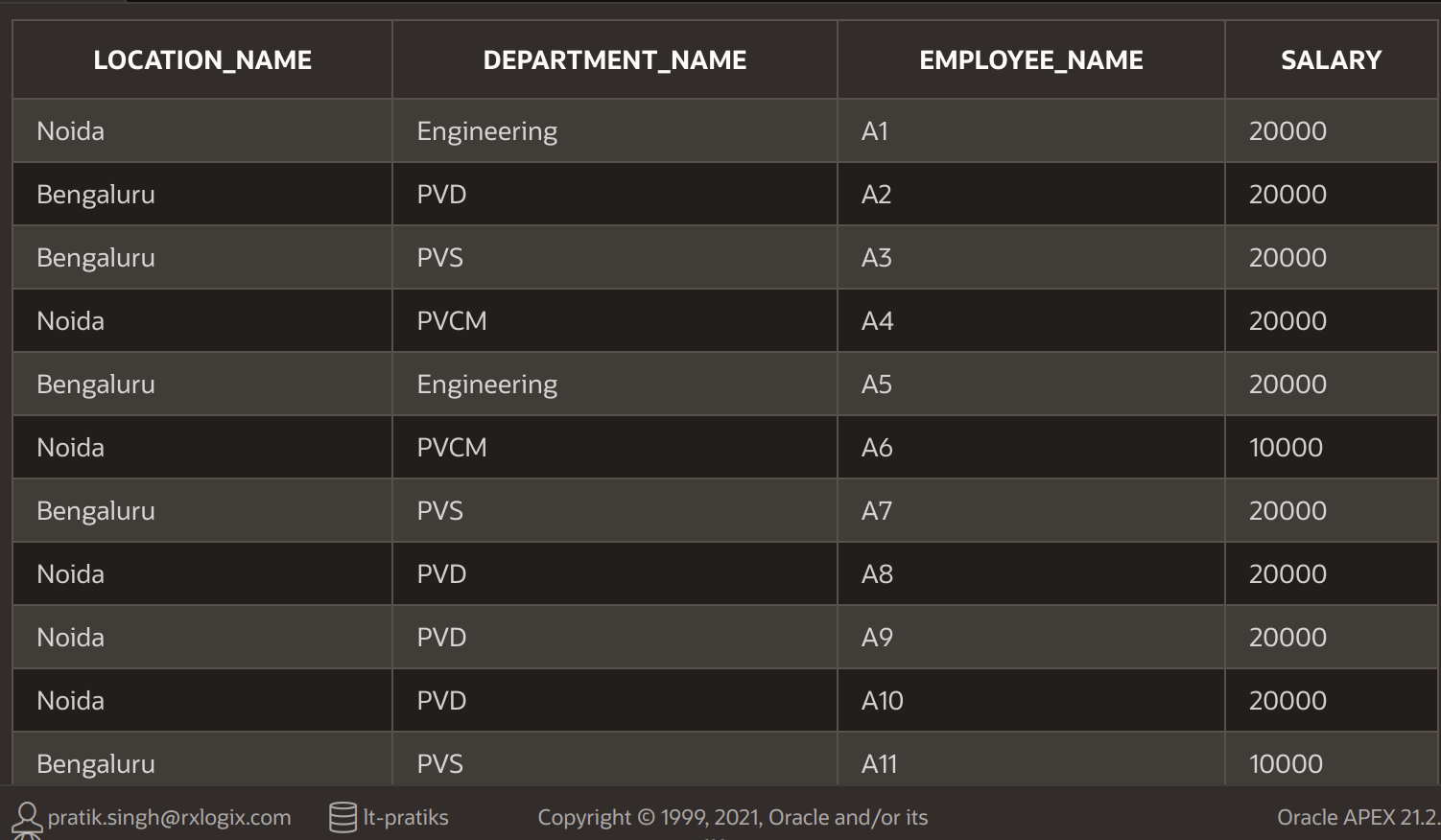
SELECT locations.location\_name, department.department\_name, employee.employee\_name, employee.salary

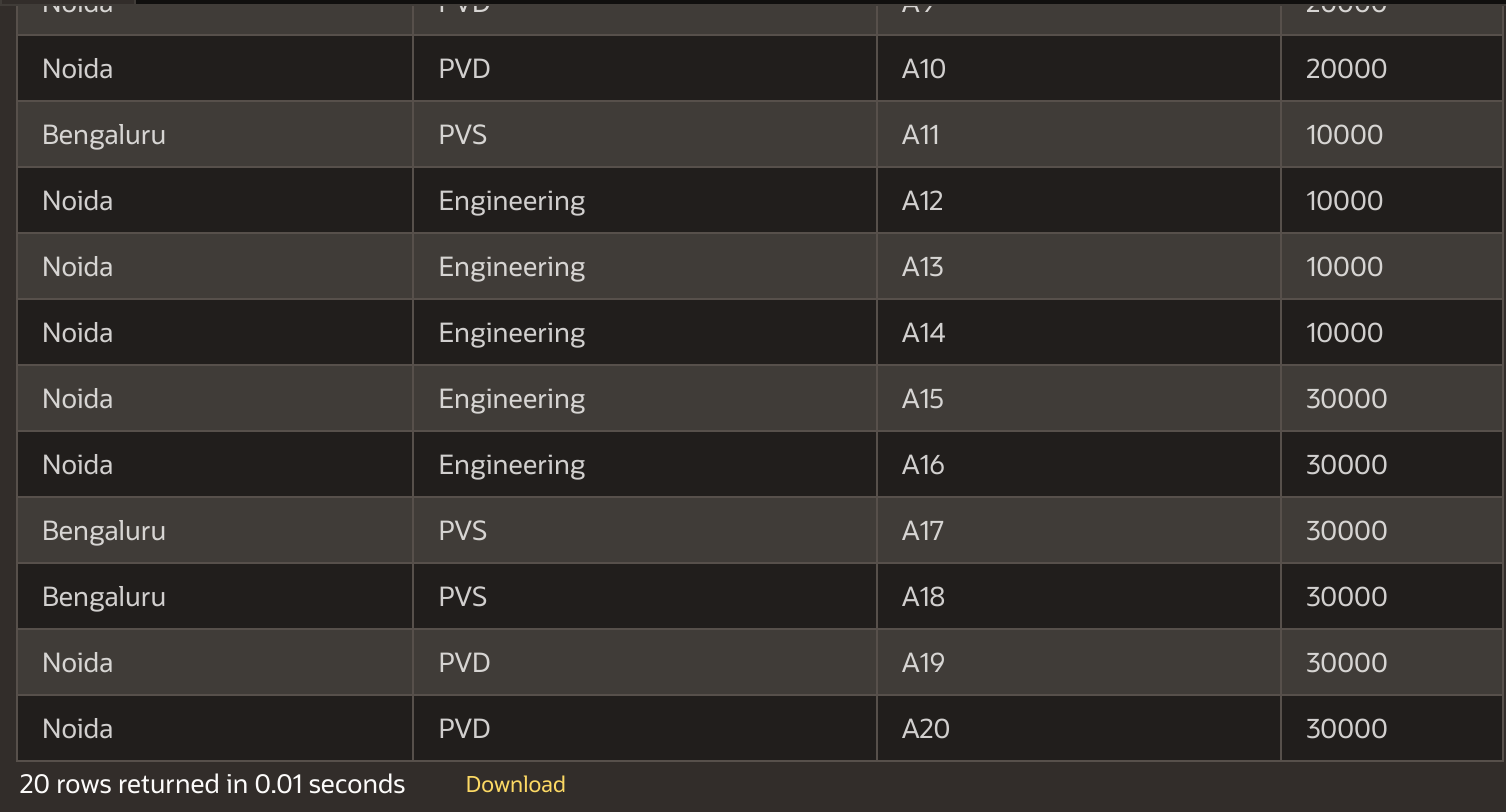
FROM ((employee

INNER JOIN department ON employee.dept\_id = department.dept\_id )

INNER JOIN locations ON employee.location\_id = locations.location\_id)

ORDER BY emp\_id;





*c) select maximum salary earned from each department*

Ans:-

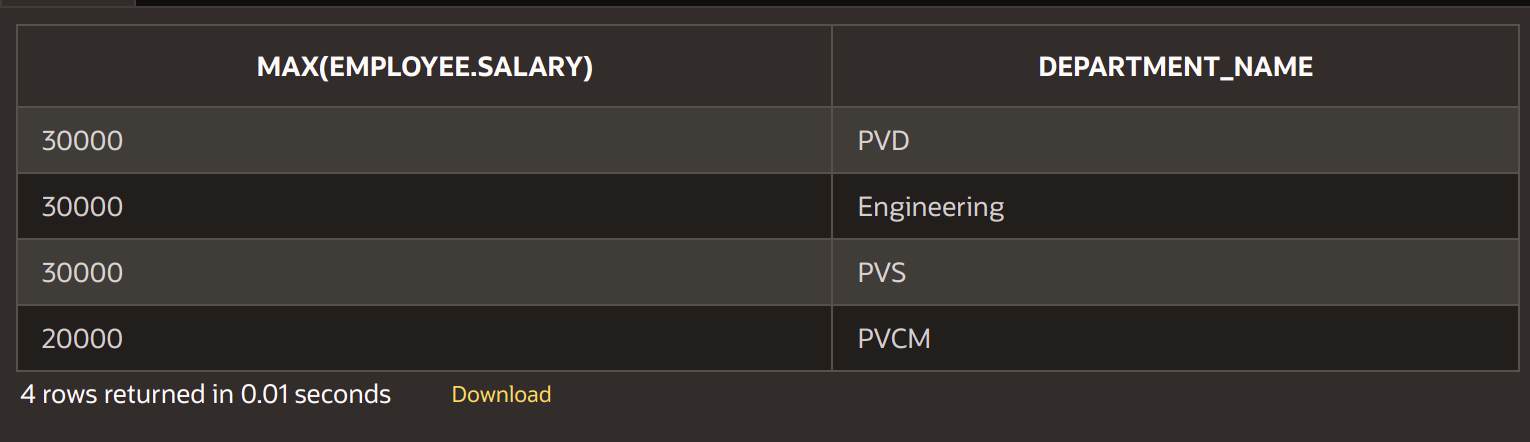
SELECT max(employee.salary), department.department\_name

FROM employee

INNER JOIN department

ON employee.dept\_id = department.dept\_id

GROUP BY department.department\_name;



*d) select 2nd highest salary from each department.*

SELECT max(salary), department.department\_name

FROM employee a

Join department

ON a.dept\_id = department.dept\_id

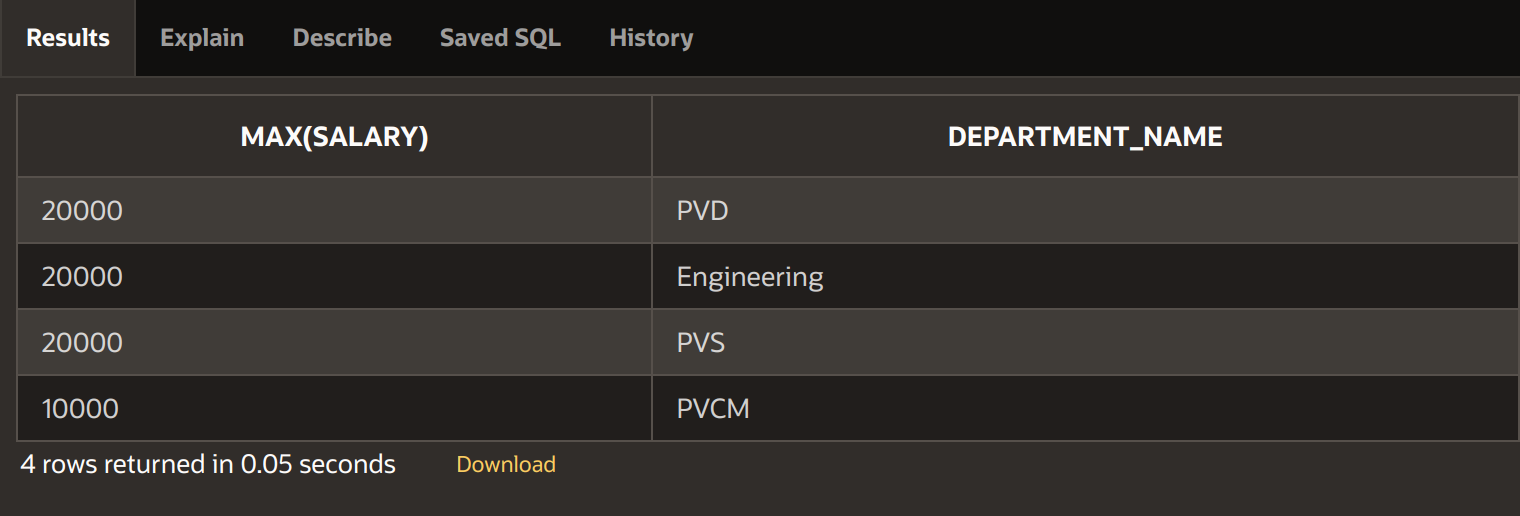
WHERE 1 = (SELECT count(distinct salary)

FROM employee b

where b.salary > a.salary and a.dept\_id = b.dept\_id

group by dept\_id)

group by department.department\_name;



*e) select location\_name, department\_name, average\_salary(of each location)*

Ans:

SELECT location\_name,department\_name,avg(salary)

FROM employee

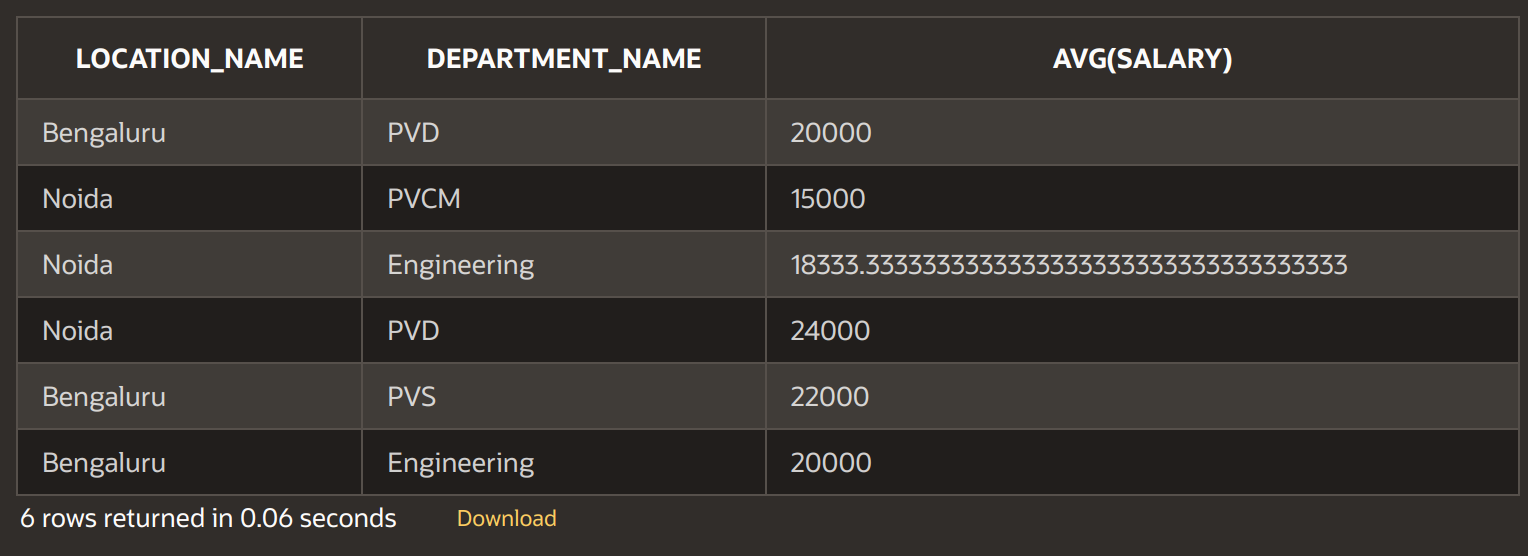
JOIN department

ON employee.dept\_id = department.dept\_id

JOIN locations

ON employee.location\_id = locations.location\_id

GROUP BY location\_name,department\_name;



*f) Show locations with no of department where no of department is 2*

Ans:-

SELECT location\_name,count(DISTINCT employee.dept\_id)

FROM employee

JOIN locations

ON employee.location\_id = locations.location\_id

HAVING count(DISTINCT employee.dept\_id) = 2

GROUP BY locations.location\_name;

