desc employee -- this command will not work because schema name is missing

desc hr.employees

SELECT \* FROM TAB

-- HOW MANY TABLES EXIST IN THIS SCHEMA

SELECT COUNT(\*) FROM TAB;

SELECT 1+1 FROM DUAL; -- IF THE VALUE IS NOT COMING FROM THE TABLE WE USE DUAL TABLE

SELECT SYSDATE FROM DUAL;

SELECT CURRENT\_TIMESTAMP FROM DUAL;

SELECT COUNT(\*) AS "NUMBER OF EMPLOYEES" FROM HR.EMPLOYEES;

SELECT COUNT(\*) AS "NUMBER OF EMPLOYEES" FROM HR.EMPLOYEES WHERE DEPARTMENT\_ID = 20;

SELECT COUNT(\*) AS "NUMBER OF EMPLOYEES" FROM HR.EMPLOYEES WHERE DEPARTMENT\_ID = 20 AND SALARY > 5000;

SELECT FIRST\_NAME,SALARY FROM HR.EMPLOYEES WHERE DEPARTMENT\_ID = 20 AND SALARY > 5000;

SELECT FIRST\_NAME,SALARY FROM HR.EMPLOYEES WHERE DEPARTMENT\_ID IN (10,20,30,50) AND SALARY > 5000;

SELECT FIRST\_NAME,SALARY FROM HR.EMPLOYEES WHERE DEPARTMENT\_ID IN (10,20,30,50) AND SALARY > 5000 ORDER BY FIRST\_NAME;

-- SORT DESENDING

SELECT FIRST\_NAME,SALARY FROM HR.EMPLOYEES WHERE DEPARTMENT\_ID IN (10,20,30,50) AND SALARY > 5000 ORDER BY FIRST\_NAME DESC;

SELECT FIRST\_NAME,SALARY FROM HR.EMPLOYEES WHERE DEPARTMENT\_ID IN (10,20,30,50) AND SALARY > 5000 ORDER BY 1 DESC; -- ORDER BY COLUMN NUMBER

SELECT FIRST\_NAME,SALARY FROM HR.EMPLOYEES ORDER BY SALARY DESC ;

-- TOP N QUERY WE USE ROWNUM IS A PSEUDO COLUMN

SELECT FIRST\_NAME,SALARY FROM HR.EMPLOYEES WHERE ROWNUM <= 5 ORDER BY 1 DESC;

SELECT SUM(SALARY) FROM HR.EMPLOYEES;

SELECT DEPARTMENT\_ID,SUM(SALARY) FROM HR.EMPLOYEES; -- ORA-00937: not a single-group group function

SELECT DEPARTMENT\_ID,SUM(SALARY) FROM HR.EMPLOYEES GROUP BY DEPARTMENT\_ID ORDER BY DEPARTMENT\_ID;

SELECT DEPARTMENT\_ID,SUM(SALARY) FROM HR.EMPLOYEES GROUP BY DEPARTMENT\_ID HAVING SUM(SALARY)> 100000 ORDER BY DEPARTMENT\_ID;

desc hr.employees;

-- 1. The HR department needs a query to display all unique job codes from the EMPLOYEES table.

select job\_id from hr.employees GROUP BY JOB\_ID HAVING COUNT(JOB\_ID)=1;

-- 2. The HR department has requested a report of all employees and their job IDs. Display the

-- last name concatenated with the job ID (separated by a comma and space) and name the

-- column Employee and Title.

SELECT LAST\_NAME || ',' || JOB\_ID AS "EMPLOYEE NAME AND TITLE" FROM HR.EMPLOYEES;

-- 3. To familiarize yourself with the data in the EMPLOYEES table, create a query to display all

-- the data from that table. Separate each column output with a comma. Name the column

-- THE\_OUTPUT.

SELECT EMPLOYEE\_ID || ','|| FIRST\_NAME || ',' || LAST\_NAME || ',' || EMAIL || ',' || PHONE\_NUMBER || ',' || HIRE\_DATE || ',' || JOB\_ID || ',' || SALARY || ',' || COMMISSION\_PCT || ',' || MANAGER\_ID || ',' || DEPARTMENT\_ID AS "THE OUTPUT" FROM HR.EMPLOYEES;

-- 4. Because of budget issues, the HR department needs a report that displays the last name and

-- salary of employees who earn more than $12,000

SELECT LAST\_NAME,SALARY FROM HR.EMPLOYEES WHERE SALARY > 12000;

-- 5. Create a report that displays the last name and department number for employee 176.

SELECT LAST\_NAME,DEPARTMENT\_ID FROM HR.EMPLOYEES WHERE EMPLOYEE\_ID = 176;

-- 6. The HR department needs to find high-salary and low-salary employees to display the last

-- name and salary for any employee whose salary is not in the $5,000–$12,000 range.

SELECT LAST\_NAME,SALARY FROM HR.EMPLOYEES WHERE SALARY NOT BETWEEN 5000 AND 12000;

-- 7. Create a report to display the last name, job ID, and start date for the employees whose last

-- names are Matos and Taylor. Order the query in ascending order by start date.

SELECT LAST\_NAME,JOB\_ID,HIRE\_DATE FROM HR.EMPLOYEES WHERE LAST\_NAME IN ('Matos','Taylor') ORDER BY HIRE\_DATE;

-- 8. Display the last name and department number of all employees in departments 20 or 50 in

-- ascending alphabetical order by name.

SELECT LAST\_NAME,DEPARTMENT\_ID FROM HR.EMPLOYEES WHERE DEPARTMENT\_ID IN (20,50) ORDER BY LAST\_NAME;

-- 9. Display the last name and salary of employees who earn between $5,000 and $12,000, and

-- are in department 20 or 50. Label the columns Employee and Monthly Salary, respectively

SELECT LAST\_NAME AS EMPLOYEE, SALARY AS "MONTHLY SALARY" FROM HR.EMPLOYEES WHERE SALARY BETWEEN 5000 AND 12000 AND DEPARTMENT\_ID IN (20,50);

-- 10. The HR department needs a report that displays the last name and hire date for all

-- employees who were hired in whose commission amount is 20%.1994

SELECT LAST\_NAME,HIRE\_DATE FROM HR.EMPLOYEES WHERE COMMISSION\_PCT = 0.2 AND HIRE\_DATE LIKE '%1994';

-- 11. Create a report to display the last name, salary, and commission of all employees who earn

--commissions.

SELECT LAST\_NAME,SALARY,COMMISSION\_PCT FROM HR.EMPLOYEES WHERE COMMISSION\_PCT IS NOT NULL;

-- 12. Display all employee last names in which the third letter of the name is “a.”

SELECT LAST\_NAME FROM HR.EMPLOYEES WHERE LAST\_NAME LIKE '\_\_a%';

-- 13. Display the last names of all employees who have both an ‘a’ and an ‘e’ in their last name.

SELECT LAST\_NAME FROM HR.EMPLOYEES WHERE LAST\_NAME LIKE '%a%' AND LAST\_NAME LIKE '%e%';

-- 14. Display the last name, job, and salary for all employees whose jobs are either that of a sales

-- representative or a stock clerk, and whose salaries are not equal to $2,500, $3,500, or

-- $7,000

SELECT LAST\_NAME,JOB\_ID,SALARY FROM HR.EMPLOYEES WHERE JOB\_ID IN ('SA\_MAN','ST\_CLERK') AND SALARY NOT IN (2500,3500,7500);

-- 15. Display the last name, salary, and commission for all employees

SELECT LAST\_NAME,SALARY,COMMISSION\_PCT FROM HR.EMPLOYEES;

desc HR.EMPLOYEES;  
SELECT FIRST\_NAME,SALARY,COMMISSION\_PCT,SALARY+COMMISSION\_PCT FROM HR.EMPLOYEES;  
SELECT 100+0 FROM DUAL;  
SELECT 100/0 FROM DUAL;  
SELECT FIRST\_NAME,SALARY,COMMISSION\_PCT,SALARY+NVL(COMMISSION\_PCT,0) FROM HR.EMPLOYEES;  
SELECT MAX(COMMISSION\_PCT),MIN(COMMISSION\_PCT),SUM(COMMISSION\_PCT),AVG(COMMISSION\_PCT) FROM HR.EMPLOYEES;  
SELECT DEPARTMENT\_ID,SUM(COMMISSION\_PCT) FROM HR.EMPLOYEES GROUP BY DEPARTMENT\_ID;  
SELECT DEPARTMENT\_ID,AVG(COMMISSION\_PCT) FROM HR.EMPLOYEES GROUP BY DEPARTMENT\_ID;  
SELECT DEPARTMENT\_ID,EMPLOYEE\_ID,SUM(COMMISSION\_PCT),AVG(COMMISSION\_PCT) FROM HR.EMPLOYEES GROUP BY DEPARTMENT\_ID;  
SELECT DEPARTMENT\_ID,EMPLOYEE\_ID,SUM(COMMISSION\_PCT),AVG(COMMISSION\_PCT) FROM HR.EMPLOYEES GROUP BY DEPARTMENT\_ID,EMPLOYEE\_ID;  
SELECT SALARY FROM HR.EMPLOYEES ORDER BY SALARY DESC;  
SELECT DEPARTMENT\_ID,SUM(SALARY)FROM HR.EMPLOYEES GROUP BY DEPARTMENT\_ID HAVING SUM(SALARY)>80000;  
SELECT DEPARTMENT\_ID,SUM(SALARY)FROM HR.EMPLOYEES GROUP BY DEPARTMENT\_ID HAVING SUM(SALARY)>80000 ORDER BY SUM(SALARY);  
SELECT DEPARTMENT\_ID,DEPARTMENT\_NAME,SUM(SALARY)FROM HR.EMPLOYEES GROUP BY DEPARTMENT\_ID HAVING SUM(SALARY)>100000 ORDER BY 2;  
SELECT COUNT(\*) FROM HR.EMPLOYEES,HR.DEPARTMENTS;--2889  
SELECT COUNT(\*) FROM HR.EMPLOYEES,HR.DEPARTMENTS WHERE 1=1; --2889  
SELECT COUNT(\*) FROM HR.EMPLOYEES,HR.DEPARTMENTS WHERE 1=0;  
SELECT COUNT(\*) FROM HR.EMPLOYEES;  
SELECT COUNT(\*) FROM HR.DEPARTMENTS;

CREATE TABLE VOTERS (

N\_VOTER\_ID NUMBER,

C\_VOTER\_NAME VARCHAR2(25),

N\_AGE NUMBER,

C\_GENDER VARCHAR2(15),

N\_WARD\_NO NUMBER,

C\_WARD\_NAME VARCHAR2(25)

);

DESC VOTERS

ALTER TABLE VOTERS

ADD V\_STATUS VARCHAR2(1) DEFAULT 'N'

ALTER TABLE VOTERS ADD D\_LAST\_UPDATED DATE DEFAULT CURRENT\_TIMESTAMP

ALTER TABLE VOTERS MODIFY D\_LAST\_UPDATED DATE DEFAULT CURRENT\_TIMESTAMP

CREATE TABLE VOTERS (

N\_VOTER\_ID NUMBER,

C\_VOTER\_NAME VARCHAR2(25),

N\_AGE NUMBER,

C\_GENDER VARCHAR2(15),

N\_WARD\_NO NUMBER,

C\_WARD\_NAME VARCHAR2(25)

);

DESC VOTERS;

ALTER TABLE VOTERS ADD CONSTRAINT PK\_N\_VOTER\_ID PRIMARY KEY(N\_VOTER\_ID);

ALTER TABLE VOTERS ADD CONSTRAINT NN\_N\_WARD\_NO NOT NULL(N\_WARD\_NO);

ALTER TABLE VOTERS MODIFY

CREATE TABLE ELECTORS (

N\_ELECTOR\_ID NUMBER,

C\_ELECTOR\_NAME VARCHAR2(25),

N\_AGE NUMBER,

C\_GENDER VARCHAR2(15),

N\_WARD\_NO NUMBER,

C\_WARD\_NAME VARCHAR2(25),

C\_PARTY\_NAME VARCHAR2(25)

);

CREATE TABLE CUSTOMER\_DETAILS(N\_ACCOUNT NUMBER(7,2),C\_NAME VARCHAR2(20), C\_EMAIL VARCHAR2(20), C\_ADDRESS VARCHAR(50), C\_PHONE\_NO VARCHAR2(10) NOT NULL);

DESC CUSTOMER\_DETAILS;

ALTER TABLE CUSTOMER\_DETAILS ADD CONSTRAINT PK\_N\_ACCOUNT PRIMARY KEY(N\_ACCOUNT);

CREATE TABLE ORDER\_DETAIL

(N\_ORDER NUMBER(5),

D\_ORDER\_DATE DATE NOT NULL,

N\_ORDER\_TOTAL NUMBER(10),

C\_PAYMENT\_TYPE VARCHAR2(15),

N\_ACCOUNT NUMBER(10)

);

ALTER TABLE ORDER\_DETAIL

ADD CONSTRAINT FK\_N\_ACCOUNT

FOREIGN KEY (N\_ACCOUNT) REFERENCES CUSTOMER\_DETAILS(N\_ACCOUNT);

INSERT INTO CUSTOMER\_DETAILS(

N\_ACCOUNT, C\_NAME,C\_EMAIL,C\_ADDRESS,C\_PHONE\_NO)

VALUES (

12345,'NESAN','KARAN@GMAIL.COM','XXER','9080234559'

);

INSERT INTO ORDER\_DETAIL(N\_ORDER NUMBER(5),

D\_ORDER\_DATE,

N\_ORDER\_TOTAL,

C\_PAYMENT\_TYPE,

N\_ACCOUNT)

VALUES (12,'12/DEC/2022',500,'GPAY',1)

SELECT COUNT(\*) FROM HR.EMPLOYEES; -- 107

SELECT COUNT(\*) FROM HR.DEPARTMENTS; -- 27

SELECT COUNT(\*) FROM HR.EMPLOYEES, HR.DEPARTMENTS; -- 2889

DESC HR.EMPLOYEES;

SELECT FIRST\_NAME,HR.DEPARTMENTS.DEPARTMENT\_ID,HR.DEPARTMENTS.DEPARTMENT\_NAME FROM HR.EMPLOYEES INNER JOIN HR.DEPARTMENTS ON HR.DEPARTMENTS.DEPARTMENT\_ID = HR.EMPLOYEES.DEPARTMENT\_ID;

SELECT FIRST\_NAME,EMP.DEPARTMENT\_ID,DEPT.DEPARTMENT\_NAME FROM HR.EMPLOYEES EMP INNER JOIN HR.DEPARTMENTS DEPT ON DEPT.DEPARTMENT\_ID = EMP.DEPARTMENT\_ID;

RE C DE EM

SELECT EMP.FIRST\_NAME,DEPT.DEPARTMENT\_NAME,REGION.REGION\_NAME, COUNTRY.COUNTRY\_NAME

FROM HR.REGIONS REGION

INNER JOIN

HR.COUNTRIES COUNTRY

ON REGION.REGION\_ID = COUNTRY.REGION\_ID

INNER JOIN

HR.LOCATIONS LOC

ON COUNTRY.COUNTRY\_ID = LOC.COUNTRY\_ID

INNER JOIN

HR.DEPARTMENTS DEPT

ON LOC.LOCATION\_ID = DEPT.LOCATION\_ID

INNER JOIN

HR.EMPLOYEES EMP

ON DEPT.DEPARTMENT\_ID = EMP.DEPARTMENT\_ID;