# **WHY WE NEED SQL?**

**IF WE WANT TO COMMUNICATE WITH THE DATA BASE THROUGH ANY PROGRAMMING LANGUAGE THEN WE SHOULD USE A QUARY LANGUAGE KNOWN AS STRUCTURE QUARY LANGUAGE.**

**THERE ARE FIVE STATEMENT OF SQL.**

**1-DDL—DATA DEFINITION LANGUAGE.**

**2-DML—DATA MANIPULATION LANGUAGE.**

**3-DQL—DATA QUARY LANGUAGE.**

**4-TCL—TRANSACTION CONTROL LANGUAGE.**

**5-DCL—DATA CONTROL LANGUAGE.**

## **DATA:-**

**🡪IT IS THE RAW FACT WHICH DESCRIBES THE PROPERTIES OF AN ENTITY.**

**🡪DATA BASE IS THE PLACE OR MEDIUM WHERE WE CAN STORE MULTIPLE NO. OF DATA.**

## **DBMS:-**

**🡪THE DBMS STANDS FOR DATA BASE MANAGEMENT SYSTEM. WHERE THE DATA WILL BE STORE IN AN ORGANISED AND SEQUENTIAL MANNER.**

**🡪WE ARE USING DBMS TO PERFORM CRUD OPERATIONS.**

**CRUD**

**CREATE, READ, UPDATE, DELETE**

**🡪IN DBMS THE DATA WILL BE STORE IN FORM OF FILES.**

**🡪RDBMS STANDS FOR RELATIONAL DATA BASE MANAGEMENT SYSTEM IN WHICH THE DATA WILL BE STORE IN THE FORM TABLES.**

**TABLES:-**

**🡪IT IS THE COMBINATION OF MULTIPLE ROWS AND COLUMNS.**

**🡪E.F CODD THE SCIENTIST WHO DEVELOP RDBMS.**

**🡪ACCORDING TO THE E.F CODD THE DATA WHICH WILL BE STORE INSIDE THE TABLE MUST BE OF SINGLE VALUED DATA.**

**🡪WE CAN STORE THE DATA IN MULTIPLE TABLES IF NEEDED WE CAN ESTABLISH A CONNECTION BETWEEN THE TABLES AND RETRIEVE DATA SIMULTANEOUSLY FROM BOTH THE TABLES BY USING RDBMS.**

**DATA TYPE:-**

**🡪THE DATA TYPE IN SQL TELLS US WHICH TYPE OF DATA WE CAN STORE INTO A PARTICULAR COLUMN.**

**🡪THERE ARE FIVE TYPES OF DATA TYPE:-**

**(I)CHAR:-**

**🡪THE ALLOWED CHARACTERS ARE UPPERCASE A-Z , LOWERCASE A-Z, DIGITS 0-9 AND ALL THE SPECIAL CHARACTERS.**

**🡪IT IS MANDATORY TO ENCLOSE THE CHAR TYPE VALUES WITHIN SINGLE COATS.**

**🡪AT THE TIME OF DECLARATION OF CHAR DATA TYPE IT IS MANDATORY TO MENTION THE SIZE.**

**🡪THE MAX SIZE OF CHAR DATA TYPE FOLLOWS FIXED LENGTH MEMORY ALLOCATION.**

**(II)VARCHAR:-**

**🡪IT IS EXACTLY SAME AS CHAR DATA TYPE EXCEPT THE FOLLOWING DIFFERENCE LIKE IT FOLLOWS VARIABLE LENGTH MEMORY ALLOCATION.**

**🡪THE MAX SIZE IS 2000.**

**(III)VARCHAR2:-**

**-🡪T IS EXACTLY SAME AS VARCHAR TYPE EXCEPT THE SIZE.**

**🡪THE MAX SIZE OF VARCHAR 2 IS 4000.**

**(IV)NUMBER:-**

**🡪THE NUMBER DATA TYPES IS USED TO STORE THE INTEGRAL VALUE.**

**🡪IN NUMBER DATA TYPES IT IS MANDATORY TO MENTION THE SIZE.**

**🡪THERE ARE TWO TYPES**

**NUMBER(SIZE)- EX: NUMBER(5)**

**NUMBER (PRECISION, SCALE) //PRECISION = WHOLE NUMBER, SCALE = DECIMAL POINT**

**EXAMPLES:-**

**NUMBER (5,3) =66.666**

**NUMBER (3,5) =.00666**

**NUMBER (3,3) =.666**

**-SCALE CANNOT BE -VE**

**(V)DATE:-**

**DD– MMM–YYYYY = 17-AUGUST-2024/17-08-2024**

**DD—MMM—YY = 17–AUGUST-24/17-08-24**

**LARGE OBJECT(LOB)**

**🡪THERE ARE TWO TYPES OF LARGE OBJECT:-**

**1.BINARY LOB: -**

**🡪IT IS USED TO STORE THE BINARY VALUES LIKE PHOTOS, VIDEOS, MP3, MP4, ETC.**

**🡪MAX SIZE: - 4 GB**

**2. CHARACTER LOB: -**

**🡪IT IS USED TO STORE THE CHARACTERS UP TO 4 GB SIZE**

**FORMAT THE LINE**

**SET PAGESIZE 1111**

**SET LINESIZE 1111;**

**CL SCR; //SYNTAX TO CLEAR SCREEN;**

**CREATE TABLE STD1(SNAME VARCHAR(5),SID NUMBER(3)); // SYNTAX FOR CREATING TABLE**

**SELECT \* FROM TAB;// TO CHECK THE NUMBER OF TABLES PRESENT**

**TNAME TABTYPE CLUSTERID**

**DEPT TABLE**

**EMP TABLE**

**BONUS TABLE**

**SALGRADE TABLE**

**STD1 TABLE**

**DESC STUDENT1; // SYNTAX TO DESCRIBE THE TABLE**

**DQL**

**THE DQL STATEMENT OF SQL IS USED TO RETRIEVE THE DATA FROM THE DATA BASE.**

**🡪THERE ARE FOUR STATEMENTS OF DQL: -**

**1. SELECT**

**2. PROJECTION**

**3. SELECTION**

**4.JOINS**

# **WHY WE NEED SQL?**

**🡪IF WE WANT TO COMMUNICATE WITH THE DATA BASE THROUGH ANY PROGRAMMING LANGUAGE THEN WE SHOULD USE A QUARY LANGUAGE KNOWN AS STRUCTURE QUARY LANGUAGE.**

**🡪THERE ARE FIVE STATEMENT OF SQL.**

**1-DDL—DATA DEFINITION LANGUAGE.**

**2-DML—DATA MANIPULATION LANGUAGE.**

**3-DQL—DATA QUERY LANGUAGE.**

**4-TCL—TRANSACTION CONTROL LANGUAGE.**

**5-DCL—DATA CONTROL LANGUAGE.**

**DATA:-**

**🡪IT IS THE RAW FACT WHICH DESCRIBES THE PROPERTIES OF AN ENTITY.**

**🡪DATA BASE IS THE PLACE OR MEDIUM WHERE WE CAN STORE MULTIPLE NO. OF DATA.**

**DBMS:-**

**🡪THE DBMS STANDS FOR DATA BASE MANAGEMENT SYSTEM. WHERE THE DATA WILL BE STORE IN AN ORGANISED AND SEQUENTIAL MANNER.**

**🡪WE ARE USING DBMS TO PERFORM CRUD OPERATIONS.**

**CRUD-- CREATE, READ, UPDATE, DELETE**

**🡪IN DBMS THE DATA WILL BE STORE IN FORM OF FILES.**

**🡪RDBMS STANDS FOR RELATIONAL DATA BASE MANAGEMENT SYSTEM IN WHICH THE DATA WILL BE STORE IN THE FORM TABLES.**

**TABLES:-IT IS THE COMBINATION OF MULTIPLE ROWS AND COLUMNS.**

**🡪E.F CODD WAS THE SCIENTIST WHO DEVELOP RDBMS.**

**🡪ACCORDING TO E.F CODD THE DATA WHICH WILL BE STORED INSIDE THE TABLE MUST BE OF SINGLE VALUED DATA.**

**🡪WE CAN STORE THE DATA IN MULTIPLE TABLES IF NEEDED WE CAN ESTABLISH A CONNECTION BETWEEN THE TABLES AND RETRIEVE DATA SIMULTANEOUSLY FROM BOTH THE TABLES BY USING RDBMS.**

**DQL**

**🡪THE DQL STATEMENT OF SQL IS USED TO RETRIVE THE DATA FROM THE DATABASE .**

**THERE ARE 4 STATEMENTS OF DQL:-**

**1.SELECT**

**2.PROJECTION**

**3.SELECTION**

**4.JOIN**

**1.SELECT**

**🡪THE SELECT STATEMENT IN SQL IS USED TO SELECT THE DATA FROM THE GIVEN TABLE AND DISPLAY IT INTO THE CONSOLE .**

**2.PROJECTION**

**🡪IF WE WANT TO RETRIVE THE DATA FROM A GIVEN TABLE BY USING ONLY COLUMNS THEN WE SHOULD GO FOR PROJECTION .**

**🡪IF WE ARE USING PROJECTION THEN ALL THE DATA WILL FETCH BYDEFAULTLY WHICH IS PRESENT INSIDE THAT COLUMN .**

**3.SELECTION**

**🡪IF WE WANT TO RETRIVE THE DATA FROM THE DATABASE BY SELECTION BOTH ROWS AND COLUMN THEN WE SHOULD GO FOR SELECTION .**

**4.JOIN**

**🡪IF WE WANT TO RETRIVE THE DATA FROM THE DATABASE BY USING MULTIPLE TABLE SAMULTANEOUSLY THEN WE SHOULD GO FOR JOIN .**

**SELECT**

**SYNTAX :- SELECT COLUMN\_NAME FROM TABLE\_NAME ;**

**SELECT – EXECUTE SECOND**

**FROM – EXECUTE FIRST**

**NOTE:**

**SELECT EMP.\* FROM EMP; 🡪 FROM RETRIVE ALL COLUMN IN A TABLE**

**SELECT \* FROM EMP; 🡪 FROM RETRIVE ALL COLUMN IN A TABLE**

**🡪 IT IS USED TO DISPLAY ALL THE EMP TABLE COLUMNS ALONG WITH A GIVEN COLUMN NAME IN THE QUERY.**

**SELECT EMP.\*,ENAME FROM EMP;**

**Q.WRITE A QUERY TO DISPLAY THE NAME OF THE EMPLOYEES**

**SELECT ENAME FROM EMP;**

**Q.WRITE A QUERY TO DISPLAY THE DESIGNATION OF EMPLOYEES**

**SELECT JOB FROM EMP;**

**JOB**

**CLERK**

**SALESMAN**

**SALESMAN**

**MANAGER**

**SALESMAN**

**MANAGER**

**MANAGER**

**ANALYST**

**Q.WRITE A QUERY TO DISPLAY THE DATE OF JOINING OF THE EMPLOYEES**

**SELECT HIREDATE FROM EMP;**

**HIREDATE**

**17-DEC-80**

**20-FEB-81**

**22-FEB-81**

**02-APR-81**

**28-SEP-81**

**01-MAY-81**

**09-JUN-81**

**19-APR-87**

**Q.WRITE A QUERY TO DISPLAY THE NAME AND SALARY OF A EMPLOYEES**

**SELECT ENAME, SAL FROM EMP;**

**ENAME SAL**

**SMITH 800**

**ALLEN 1600**

**WARD 1250**

**JONES 2975**

**MARTIN 1250**

**BLAKE 2850**

**CLARK 2450**

**Q.WRITE A QUERY TO DISPLAY ALL THE DETAILS OF A EMPLOYEES**

**SELECT \* FROM EMP;**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**7369 SMITH CLERK 7902 17-DEC-80 800 20**

**7499 ALLEN SALESMAN 7698 20-FEB-81 1600 300 30**

**7521 WARD SALESMAN 7698 22-FEB-81 1250 500 30**

**7566 JONES MANAGER 7839 02-APR-81 2975 20**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMPLOYEES ALONG WITH THEIR NAMES**

**SELECT EMP.\*, ENAME FROM EMP;**

**DISTINCT**

**🡪THE DISTINCT CLAUSE IS USED TO REMOVE THE DUPLICATE VALUES OF A GIVEN COLUMN .**

**🡪IF WE USE DISTINCT CLAUSE WITH MULTIPLE COLUMN THEN IT FIRST COMBINE ALL THE COLUMNS THEN GIVE THE DISTINCT VALUE .**

**Q.WRITE A QUERY TO DISPLAY THE DEPARTMENT NUMBER OF THE EMPLOYEES BY REMOVING DUPLICATE.**

**SELECT DISTINCT DEPTNO FROM EMP;**

**DEPTNO**

**30**

**20**

**10**

**Q.WRITE A QUERY TO DISPLAY THE JOB AND DEPARTMENT NUMBER OF EMPLOYEES BY REMOVING DUPLICATES.**

**SELECT DISTINCT JOB, DEPTNO FROM EMP;**

**JOB DEPTNO**

**MANAGER 20**

**PRESIDENT 10**

**CLERK 10**

**SALESMAN 30**

**ANALYST 20**

**MANAGER 30**

**MANAGER 10**

**CLERK 30**

**CLERK 20**

**Q.WRITE A QUERY TO DISPLAY THE SALARY OF EMPLOYEES**

**SELECT SAL FROM EMP;**

**Q.WRITE A QUERY TO DISPLAY ANNUAL SALARY OF EMPLOYEES**

**SELECT SAL\*12 FROM EMP;**

**SAL\*12**

**9600**

**19200**

**15000**

**35700**

**15000**

**34200**

**29400**

**36000**

**Q.WRITE A QUERY TO DISPLAY NAME AND SALARY OF EMPLOYEES WITH A HIKE OF 20%;**

**SELECT ENAME, SAL+(20\*SAL/100) FROM EMP;**

**ALIAS**

**🡪IT IS USED TO PROVIDE THE ALTERNATIVE NAME TO A GIVEN COLUMN IN THE RESULTANT TABLE;**

**🡪 WE CAN ACHIEVE ALIAS WITH OR WITHOUT USING AS KEYWORD**

**SELECT SAL\*12 AS "ANNUAL SAL" FROM EMP; // WITH USING AS KEYWORD**

**SELECT SAL\*12 "ANNUAL SAL" FROM EMP; // WITHOUT USING AS KEYWORD**

**🡪IF THE ALIAS NAME IS SINGLE WORD THEN DOUBLE QUATATION IS NOT REQUIRED .**

**Q.WRITE A QUERY TO DSIPLAY TO DETAILS OF THE EMPLOYEE ALONG WITH THEIR ANNUAL SALARY;**

**SELECT EMP.\*, SAL\*12 AS ANNUALSAL FROM EMP;**

**SELECTION**

**🡪IF WE WANT TO RETRIEVE THE DATA FROM THE DATABASE BY USING BOTH COLUMN AND ROWS THEN WE SHOULD GO FOR SELECTION**

**🡪WE CAN ACHIEVE BY USING WHERE CLAUSE**

**Q.WRITE A QUERY TO DISPLAY THE NAME,JOB AND DEPARTMENT NUMBER OF THE EMPLOYEES THOSE ARE WORKING IN DEPARTMENT NUMBER 10;**

**SELECT ENAME,JOB, DEPTNO FROM EMP WHERE DEPTNO=10;**

**Q.WRITE A QUERY TO DISPLAY THE NAME AND JOB OF EMPLOYEE WORKING AS MANAGER**

**SELECT ENAME, JOB FROM EMP WHERE JOB='MANAGER';**

**// THE DATA INSIDE THE TABLE ARE CASE-SENSITIVE BUT THE STRUCTURE ROWS AND COLUMN NAME ARE CASE SENSITIVE;**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF EMPLOYEES THOSE ARE EARNING MORE THAM 1500;**

**SELECT \* FROM EMP WHERE SAL>1500;**

**Q.WRITE A QUERY TO DISPLAY THE NAME AND JOB OF EMPLOYEE WHOSE NAME IS 'MILLER';**

**SELECT ENAME, JOB FROM EMP WHERE ENAME='MILLER';**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF EMPLOYEES THOSE ARE HIRED BEFORE 1987;**

**SELECT \* FROM EMP WHERE HIREDATE<'01-JAN-1987';**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF EMPLOYEES ALONG WITH THE HIREDATE THOSE ARE WORKING AS SALESMAN;**

**SELECT EMP.\*, HIREDATE FROM EMP WHERE JOB='SALESMAN';**

**WHERE CLAUSE**

**🡪THE WHERE CLAUSE IS USED TO FILTER THE RECORDS OR ROWS AND IT WILL GET EXECUTED ROW BY ROW**

**🡪THE WHERE CLAUSE EXECUTES AFTER THE FROM CLAUSE**

**🡪WE CAN'T USE MULTI-ROW FUNCTION WITH WHERE CLAUSE**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF EMPLOYEES EXCEPT THE EMPLOYEE OF DEPT NO-30;**

**SELECT \* FROM EMP WHERE DEPTNO!=30;**

**CONCATENATION OPERATOR**

**🡪IT IS USED TO MERGE THE STRINGS ,**

**🡪SYMBOL OF CONCATENATION OPERATOR IS || .**

**SELECT 'HII ' || DISTINCT JOB FROM EMP; (WRONG)**

**SELECT 'HII ' || DISTINCT JOB FROM EMP**

**\***

**ERROR AT LINE 1:**

**ORA-00936: MISSING EXPRESSION**

**SELECT DISTINCT JOB||' HII' FROM EMP; (CORRECT)**

**JOB||'HII'**

**SALESMAN HII**

**MANAGER HII**

**PRESIDENT HII**

**ANALYST HII**

**CLERK HII**

**SELECT DISTINCT 'HII ' ||JOB FROM EMP;**

**'HII'||JOB**

**HII CLERK**

**HII SALESMAN**

**HII MANAGER**

**HII PRESIDENT**

**HII ANALYST**

**LOGICAL OPERATOR**

**LOGICAL AND SYMBOL-'AND'**

**LOGICAL OR SYMBOL-'OR'**

**🡪WRITE A QUERY TO DISPLAY THE JOB AND DEPTNO OF THE EMPLOYEES THOSE ARE WORKING AS 'MANAGER' IN DEPTNO=30;**

**SELECT JOB, DEPTNO FROM EMP WHERE JOB='MANAGER' AND DEPTNO=30;**

**JOB DEPTNO**

**MANAGER 30**

**Q.WRITE A QUERY TO DISPLAY THE NAME,DEPTNO AND SALARY OF THE EMPLOYEES THOSE ARE WORKING IN DEPTNO=20 AND EARNING LESS THAN 3000;**

**SELECT ENAME,DEPTNO,SAL FROM EMP WHERE DEPTNO=20 AND SAL<3000;**

**ENAME DEPTNO SAL**

**SMITH 20 800**

**JONES 20 2975**

**ADAMS 20 1100**

**Q.WRITE A QUERY TO DISPLAY THE ANNUAL SALARY THOSE ARE EARNING MORE THAN 1250 BUT LESS THAN 4000;**

**SELECT SAL\*12 FROM EMP WHERE SAL>1250 AND SAL<4000;**

**SAL\*12**

**19200**

**35700**

**34200**

**29400**

**SELECT SAL\*12 AS "ANNUAL SALARY" FROM EMP WHERE SAL>1250 AND SAL<4000;**

**ANNUAL SALARY**

**19200**

**35700**

**34200**

**29400**

**Q.WRITE A QUERY TO DISPLAY THE NAME AND DEPTNO OF EMPLOYEES THOSE ARE WORKING IN DEPTNO 10 OR 30;**

**SELECT ENAME,DEPTNO FROM EMP WHERE DEPTNO=10 OR DEPTNO=30;**

**ENAME DEPTNO**

**ALLEN 30**

**WARD 30**

**MARTIN 30**

**BLAKE 30**

**CLARK 10**

**KING 10**

**Q.WRITE A QUERY TO DISPLAY THE NAME, JOB AND DEPTNO OF THE EMPLOYEES WORKING AS 'CLERK' IN DEPTNO=10 OR 20;**

**SELECT ENAME,JOB,DEPTNO FROM EMP WHERE JOB='CLERK' AND DEPTNO=10 OR JOB='CLERK' AND DEPTNO=20;**

**OR**

**SELECT ENAME,JOB,DEPTNO FROM EMP WHERE JOB='CLERK'AND (DEPTNO=10 OR DEPTNO=20);**

**ENAME JOB DEPTNO**

**SMITH CLERK 20**

**ADAMS CLERK 20**

**MILLER CLERK 10**

**Q.WRITE A QUERY TO DISPLAY THE NAME, JOB AND DEPTNO OF THE EMPLOYEE WORKING AS CLERK OR MANAGER IN DEPTNO=10;**

**SELECT ENAME,JOB,DEPTNO FROM EMP WHERE (JOB='CLERK' OR JOB='MANAGER') AND DEPTNO=10;**

**ENAME JOB DEPTNO**

**CLARK MANAGER 10**

**MILLER CLERK 10**

**IN OPERATOR**

**🡪WHENEVER WE HAVE MULTIPLE CONDITIONS AT RHS THEN WE SHOULD GO FOR IN OPERATOR. AS ALTERNATIVE OF 'OR' OPERATOR WE GO FOR 'IN' OPERATOR.**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMPLOYEES THOSE ARE WORKING IN DEPTNO 10 OR 20;**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**7369 SMITH CLERK 7902 17-DEC-80 800 20**

**7566 JONES MANAGER 7839 02-APR-81 2975 20**

**7782 CLARK MANAGER 7839 09-JUN-81 2450 10**

**7788 SCOTT ANALYST 7566 19-APR-87 3000 20**

**Q.WRITE A QUERY TO DISPLAY THE NAME AND JOB OF EMPLOYEES THOSE ARE WORKING AS CLERK, MANAGER OR SALESMAN**

**SELECT ENAME,JOB FROM EMP WHERE JOB IN('CLERK','MANAGER','SALESMAN');**

**ENAME JOB**

**SMITH CLERK**

**ALLEN SALESMAN**

**WARD SALESMAN**

**JONES MANAGER**

**MARTIN SALESMAN**

**Q.WRITE A QUERY TO DISPLAY THE EMPLOYEE NUMBER, ENAME AND SALARY OF EMPLOYEE WHOSE EMPLOYEE NUMBER IS 7902 OR 7839 BUT EARNING SALARY MORE THAN 3500;**

**SELECT EMPNO,ENAME,SAL FROM EMP WHERE (EMPNO IN(7902,7839)) AND SAL>3500;**

**EMPNO ENAME SAL**

**7839 KING 5000**

**Q.WRITE A QUERY TO DISPLAY THE NAME AND DEPTNO OF EMPLOYEES THOSE ARE NOT WORKING IN DEPTNO 10 OR 20;**

**SELECT ENAME,DEPTNO FROM EMP WHERE DEPTNO NOT IN(10,20);//NOT IN OPERATOR**

**ENAME DEPTNO**

**ALLEN 30**

**WARD 30**

**MARTIN 30**

**BLAKE 30**

**BETWEEN OPERATOR**

**🡪IT IS USED TO FIND THE VALUE IN A PERTICULAR RANGE .**

**Q.WRITE A QUERY TO DISPLAY THE NAME AND SALARY OF EMPLOYEES THOSE ARE EARNING SALARY BETWEEN 1250 TO 3000;**

**SELECT ENAME,SAL FROM EMP WHERE SAL BETWEEN 1250 AND 3000;//STARTING AND ENDING RANGE WILL BE INCLUDED**

**ENAME SAL**

**ALLEN 1600**

**WARD 1250**

**JONES 2975**

**MARTIN 1250**

**BLAKE 2850**

**Q.WRITE A QUERY TO DISPLAY THE NAME,DEPTNO AND HIREDATE OF EMPLOYEES WORKING IN DEPTNO 10 AND HIRED DURING 1981;**

**SELECT ENAME,DEPTNO,HIREDATE FROM EMP WHERE DEPTNO=10 AND HIREDATE BETWEEN '1-JAN-1981' AND '31-DEC-1981';**

**ENAME DEPTNO HIREDATE**

**CLARK 10 09-JUN-81**

**KING 10 17-NOV-81**

**Q.WRITE A QUERY TO DISPLAY THE NAME, SALARY AND HIREDATE OF EMPLOYEES THOSE ARE HIRED DURING 1987 IN DEPTNO 20 AND SALARY GREATER THAN 2000;**

**SELECT ENAME,SAL,HIREDATE FROM EMP WHERE (HIREDATE BETWEEN '1-JAN-1987' AND '31-DEC-1987') AND (DEPTNO=20 AND SAL>2000);**

**ENAME SAL HIREDATE**

**SCOTT 3000 19-APR-87**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMPLOYEES THOSE ARE NOT EARNING SALARY IN THE RANGE OF 1250 TO 3000;**

**SELECT \* FROM EMP WHERE SAL NOT BETWEEN 1250 AND 3000;**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**7369 SMITH CLERK 7902 17-DEC-80 800 20**

**7839 KING PRESIDENT 17-NOV-81 5000 10**

**7876 ADAMS CLERK 7788 23-MAY-87 1100 20**

**7900 JAMES CLERK 7698 03-DEC-81 950 30**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMPLOYEES ALONG WITH ANNUAL SALARY IF THEIR EARNING SALARY BETWEEN 1250 TO 2975 AND ANNUAL SALARY IS GREATER THAN 20000;**

**SELECT EMP.\*,SAL\*12 AS "ANNUAL SALARY" FROM EMP WHERE (SAL BETWEEN 1250 AND 2975) AND SAL\*12>20000;**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO ANNUAL SALARY**

**7566 JONES MANAGER 7839 02-APR-81 2975 20 35700**

**7698 BLAKE MANAGER 7839 01-MAY-81 2850 30 34200**

**7782 CLARK MANAGER 7839 09-JUN-81 2450 10 29400**

**Q.WRITE A QUERY TO DISPLAY THE NAME, ANNUAL SALARY, JOB AND DEPTNO OF EMPLOYEES WORKING AS MANAGER OR CLERK IN DPETNO 10 OR 30;**

**SELECT ENAME,SAL\*12 AS "ANNUAL SALARY",JOB,DEPTNO FROM EMP WHERE JOB IN('MANAGER','CLERK') AND DEPTNO IN(10,30);**

**ENAME ANNUAL SALARY JOB DEPTNO**

**BLAKE 34200 MANAGER 30**

**CLARK 29400 MANAGER 10**

**JAMES 11400 CLERK 30**

**MILLER 15600 CLERK 10**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMPLOYEES THOSE ARE HIRED AFTER 1981 BUT BEFORE 1987;**

**SELECT \* FROM EMP WHERE HIREDATE BETWEEN '01-JAN-1982' AND '31-DEC-1986';**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**7934 MILLER CLERK 7782 23-JAN-82 1300 10**

**IS/IS NOT OPERATOR**

**🡪IT IS USE TO COMPARE THE NULL VALUES .**

**Q,WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMPLOYEES THOSE ARE NOT GETTING ANY COMMISSION**

**SELECT \* FROM EMP WHERE COMM IS NULL;**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**7369 SMITH CLERK 7902 17-DEC-80 800 20**

**7566 JONES MANAGER 7839 02-APR-81 2975 20**

**7698 BLAKE MANAGER 7839 01-MAY-81 2850 30**

**7782 CLARK MANAGER 7839 09-JUN-81 2450 10**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMPLOYEES THOSE ARE GETTING COMMISSION**

**SELECT \* FROM EMP WHERE COMM IS NOT NULL;**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**7499 ALLEN SALESMAN 7698 20-FEB-81 1600 300 30**

**7521 WARD SALESMAN 7698 22-FEB-81 1250 500 30**

**7654 MARTIN SALESMAN 7698 28-SEP-81 1250 1400 30**

**7844 TURNER SALESMAN 7698 08-SEP-81 1500 0 30**

**LIKE/NOT LIKE OPERATOR**

**🡪IT IS USED TO MATCH THE PATTERN**

**🡪TO MATCH ANY PATTERN WE HAVE TWO SYMBOLS % AND \_(UNDERSCORE)**

**%🡪SKIP 1 OR MORE THAN ONE CHARACTER**

**\_ 🡪 SKIP ATMOST 1 CHARACTER AT A TIME**

**Q.WRITE A QUERY TO DISPLAY THE NAME AND SALARY OF EMPLOYEES CONTAINING 'A' AS THE FIRST CHARACTER IN THEIR NAME;**

**SELECT ENAME,SAL FROM EMP WHERE ENAME LIKE 'A%';**

**ENAME SAL**

**ALLEN 1600**

**ADAMS 1100**

**Q.WRITE A QUERY TO DISPLAY THE NAME,SALARY AND ANNUAL SALARY OF EMPLOYEES WHO CONTAIN 'N' AS THE LAST CHARACTER;**

**SELECT ENAME, SAL, SAL\*12 AS "ANNUAL SALARY" FROM EMP WHERE ENAME LIKE '%N';**

**ENAME SAL ANNUAL SALARY**

**ALLEN 1600 19200**

**MARTIN 1250 15000**

**Q.WRITE A QUERY TO DISPLAY THE NAME OF EMPLOYEES WHOSE NAME CONTAIN 'R' IN THEIR NAME;**

**SELECT ENAME FROM EMP WHERE ENAME LIKE '%R%';**

**ENAME**

**WARD**

**MARTIN**

**CLARK**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMPLOYEES WHOSE NAME STARTS WITH 'A' ENDS WITH 'S';**

**SELECT \* FROM EMP WHERE ENAME LIKE 'A%S';**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**7876 ADAMS CLERK 7788 23-MAY-87 1100 20**

**Q.WRITE A QUERY TO DISPLAY THE NAME OF THE EMPLOYEES IF THE EMPLOYEE CONTAIN CHARACTER 'A' AS HIS SECOND CHARACTER;**

**SELECT ENAME FROM EMP WHERE ENAME LIKE '\_A%';**

**ENAME**

**WARD**

**MARTIN**

**JAMES**

**Q.WRITE A QUERY TO DISPLAY THE NAME OF THE EMPLOYEES WHOSE NAME CONTAIN 'A' AS SECOND CHARACTER 'S' AS LAST CHARCTER;**

**SELECT ENAME FROM EMP WHERE ENAME LIKE '\_A%S';**

Q.**WRITE A QUERY TO DISPLAY THE THE NAME OF THE EMPLOYEES IF THEIR NAME CONTAINS 'A' TWO TIMES?**

**SELECT ENAME FROM EMP WHERE ENAME LIKE '%A%A%';**

**ENAME**

**ADAMS**

**Q.WRITE A QUERY TO DISPLAY THE NAME OF THE EMPLOYEES WHOSE NAME CONTAIN TWO CONSECUTIVE 'L';**

**SELECT ENAME FROM EMP WHERE ENAME LIKE '%LL%';**

**ENAME**

**ALLEN**

**MILLER**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMPLOYEES THOSE ARE NOT WORKING AS MANAGER AND CLERK IN DEPT\_NO 10 OR 20 WITH A SALARY IN THE RANGE OF 900 TO 3000;**

**SELECT \* FROM EMP WHERE (JOB NOT IN ('MANAGER','CLERK') AND DEPTNO IN(10,20)) AND SAL BETWEEN 900 AND AND 3000;**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**7788 SCOTT ANALYST 7566 19-APR-87 3000 20**

**7902 FORD ANALYST 7566 03-DEC-81 3000 20**

**Q.WRITE A QUERY TO DISPLAY THE NAME AND HIREDATE OF THE EMPLOYEES THOSE ARE HIRED IN THE MONTH OF FEBRUARY;**

**SELECT ENAME,HIREDATE FROM EMP WHERE HIREDATE LIKE '%FEB%';**

**ENAME HIREDATE**

**ALLEN 20-FEB-81**

**WARD 22-FEB-81**

Q.**WRITE A QUERY TO DISPLAY THE NAME AND SALARY OF THE EMPLOYEES HAVING 'E' AS THEIR LAST CHARACTER AND EARING 4 DIGIT SALARY;**

**SELECT ENAME,SAL FROM EMP WHERE ENAME LIKE'%E' AND LENGTH(SAL)=4;**

**OR**

**SELECT ENAME,SAL FROM EMP WHERE ENAME LIKE '%E' AND SAL LIKE '\_\_\_\_';**

**ENAME SAL**

**BLAKE 2850**

**Q.WRITE A QUERY TO ADD TWO NUMBER;**

**DUAL**

**🡪IT IS A DUMMY TABLE WHICH IS PRESENT IN EACH AND EVERY DATABASE TO PERFORM SOME RANDOM OPERATIONS.**

**🡪IT CONTAINS ONLY ONE ROW AND ONE COLUMN**

**SELECT 2+3 FROM DUAL;**

**2+3**

**5**

**FUNCTION**

🡪**THESE ARE THE BLOCK OF CODE OR SET OF INSTRUCTION WHICH IS USED TO PERFORM A SPECIFIC TASK;**

🡪**FUNCTIONS ARE OF TWO TYPES**

**1)MULTI ROW FUNCTION**

**2)SINGLE ROW FUNCTION**

**SINGLE ROW FUNCTION**

**🡪IN SINGLE ROW FUNCTION THE NUMBER OF OUTPUT DEPENDS UPON NUMBER OF INPUTS .**

**I) LENGTH ()**

🡪**IT IS USED TO COUNT THE COUNT THE NUMBER OF CHARACTERS PRESENT IN A GIVEN STRING;**

**Q.WRITE A QUERY TO COUNT THE NUMBER OF CHARACTERS PRESENT IN 'KING';**

**SELECT LENGTH(ENAME) FROM EMP WHERE ENAME='KING';**

**LENGTH(ENAME)**

**4**

**SELECT LENGTH(ENAME) FROM EMP ;**

**LENGTH(ENAME)**

**5**

**5**

**4**

**5**

**6**

**Q.WRITE A QUERY TO COUNT THE LENGTH OF "SASWAT';**

**SELECT LENGTH('SASWAT') FROM DUAL;**

**LENGTH('SASWAT')**

**6**

**Q.WRITE A QUERY TO DISPLAY THE NUMBER OF CHARACTERS PRESENT IN THEIR NAME THOSE ARE WORKING AS 'SALESMAN';**

**SELECT LENGTH(ENAME) FROM EMP WHERE JOB='SALESMAN';**

**LENGTH(ENAME)**

**5**

**4**

**6**

**6**

**II) UPPER()**

**🡪IT IS USED TO CONVERT A STRING INTO UPPERCASE;**

**SELECT UPPER('QSPIDER') FROM DUAL;**

**UPPER('**

**QSPIDER**

**III) LOWER()**

**🡪IT IS USED TO CONVERT A GIVEN STRING INTO LOWERCASE;**

**SELECT LOWER(ENAME) FROM EMP;**

**LOWER(ENAM**

**SMITH**

**ALLEN**

**WARD**

**JONES**

**MARTIN**

**IV) INITCAP()**

**🡪IT IS USED TO CONVERT A GIVEN STRING INTO INITIAL CAPITAL LETTER. FIRST CHARACTER OF ANY STRING WILL BE UPPERCASE;**

**SELECT INITCAP(ENAME) FROM EMP;**

**INITCAP(EN**

**Smith**

**Allen**

**Ward**

**Jones**

**V) REVERSE()**

**🡪IT IS USED TO REVERSE A STRING;**

**SELECT REVERSE('SASWAT') FROM DUAL;**

**REVERS**

**TAWSAS**

**SELECT REVERSE(ENAME) FROM EMP;**

**REVERSE(E**

**HTIMS**

**NELLA**

**DRAW**

**VI) SUBSTR()**

**🡪SUBSTRING FUNCTION IS USED TO EXTRACT A PART OF A STRING;**

**-8 -7 -6 -5 -4 -3 -2 -1**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Q | S| P| I | D | E | R | S**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**1 2 3 4 5 6 7 8**

🡪**SYNTAX:- SELECT SUBSTR('STRING',I,J);**

🡪**HERE “I” IS THE STARTING INDEX FROM WHERE THE STRING WILL GET PRINTED**

🡪”J” **IS THE LENGTH OF THE STRING**

🡪**HERE TRAVERSAL IS DONE FROM LEFT TO RIGHT(FROM INDEX 1 TOWARDS THE END) AND REVERSE TRAVERSAL IS NOT POSSIBLE(RIGHT TO LEFT)**

**SELECT SUBSTR('QSPIDERS',2,4) FROM DUAL;**

**SUBS**

**SPID**

**SELECT SUBSTR('QSPIDERS',3,5) FROM DUAL;**

**SUBST**

**PIDER**

**SELECT SUBSTR('QSPIDERS',3) FROM DUAL;**

**SUBSTR**

**PIDERS //IF WE DON'T PROVIDE THE LENGTH THEN THE STRING WILL GET PRINTED FROM STARTING INDEX TILL THE END**

**SELECT SUBSTR('QSPIDERS',5,2) FROM DUAL;**

**SU**

**DE**

**SELECT SUBSTR('QSPIDERS',-6,4) FROM DUAL;**

**SUBS**

**PIDE**

**SELECT SUBSTR('QSPIDERS',-4,-3) FROM DUAL;**

**S**

**- // OUTPUT IS BLANKSPACE BECAUSE WE CAN'T PROVIDE LENGTH AS -VE AS REVERSE TRAVERSAL(RIGHT TO LEFT) IS NOT POSSIBLE**

**SELECT SUBSTR('QSPIDERS',3,0) FROM DUAL;**

**S**

**- // OUTPUT IS BLANKSPACE BECAUSE WE HAVE GIVEN THE LENGTH OF THE STRING AS 0**

**SELECT SUBSTR('QSPIDERS',2,1) FROM DUAL;**

**S**

**S**

**SELECT SUBSTR('QSPIDERS',0,5) FROM DUAL;**

**SUBST**

**QSPID //IF WE TAKE STARTING INDEX AS 0 THEN BY DEFAULT IT WILL GET PRINTED FROM INDEX 1;**

**VI) REPLACE()**

**🡪IT IS USED TO REPLACE A STRING WITH ANOTHER STRING.**

**SELECT REPLACE('QSPIDER','Q','J') FROM DUAL;**

**REPLACE**

**JSPIDER**

**SELECT REPLACE('BANANA','A','U') FROM DUAL;**

**REPLAC**

**BUNUNU**

**SELECT REPLACE('QSPIDER','Q','PI') FROM DUAL;**

**REPLACE(**

**PISPIDER**

**SELECT REPLACE('QSPIDER','Q','123') FROM DUAL;**

**REPLACE('**

**123SPIDER**

**SELECT REPLACE('QSPIDER','Q',123) FROM DUAL;**

**REPLACE('**

**123SPIDER**

**SELECT REPLACE('BANANA','A') FROM DUAL;**

**REP**

**BNN // IF WE WILL NOT PROVIDE THE REPLACE CHARACTER THEN NULL WILL BE PROVIDED**

**\*IF THE THIRD ARGUMENT IS MISSING IN REPLACE() THEN IT WILL BE TREATED AS NULL;**

**Q.WRITE A QUERY TO COUNT THE NUMBER OF 'A' PRESENT IN BANANA;**

**SELECT LENGTH('BANANA')-LENGTH(REPLACE('BANANA','A')) FROM DUAL;**

**LENGTH('BANANA')-LENGTH(REPLACE('BANANA','A'))**

**3**

**SELECT LENGTH('BANANAAA')-LENGTH(REPLACE('BANANAAA','A')) FROM DUAL;**

**LENGTH('BANANAAA')-LENGTH(REPLACE('BANANAAA','A'))**

**5**

**VII) MOD()**

**🡪IT IS USED TO OBTAIN THE REMAINDER OF ANY NUMBER;**

**SELECT MOD(5,2) FROM DUAL;**

**MOD(5,2)**

**1**

**SELECT MOD(4,2) FROM DUAL;**

**MOD(4,2)**

**0**

**SELECT MOD(10.5,2) FROM DUAL;**

**MOD(10.5,2)**

**.5**

**Q.WRITE A QUERY TO DISPLAY THE NAME AND EMPID OF THE EMPLOYEES WHO HAVE ODD EMPLOYEE\_ID;**

**SELECT ENAME,EMPNO FROM EMP WHERE MOD(EMPNO,2)=1;**

**ENAME EMPNO**

**SMITH 7369**

**ALLEN 7499**

**WARD 7521**

**KING 7839**

**VIII) TRUNC()**

**🡪IT IS USED TO ROUND UP ANY NUMBER BY FOLLOWING THE LOWEST VALUE;**

**SELECT TRUNC(2.5) FROM DUAL;**

**TRUNC(2.5)**

**2**

**SELECT TRUNC(2.9) FROM DUAL;**

**TRUNC(2.9)**

**2**

**SELECT TRUNC(2.6) FROM DUAL;**

**TRUNC(2.6)**

**2**

**IX) ROUND()**

**SELECT ROUND(2.6) FROM DUAL;**

**ROUND(2.6)**

**3**

**SELECT ROUND(2.4) FROM DUAL;**

**ROUND(2.4)**

**2**

**SELECT ROUND(2.5) FROM DUAL;**

**ROUND(2.5)**

**3**

**X) CURRENT\_DATE()/SYSDATE()**

**🡪IT IS USED TO GET CURRENT DATE .**

**Q.WRITE A QUERY TO DISPLAY THE CURRENT DATE BY USING SQL;**

**SELECT CURRENT\_DATE FROM DUAL;**

**CURRENT\_D**

**02-SEP-24**

**Q.WRITE A QUERY TO DISPLAY THE SYSDATE BY USING SQL;**

**SELECT SYSDATE FROM DUAL;**

**SYSDATE**

**02-SEP-24**

**XI) SYSTIMESTAMP()**

**🡪IT IS USED TO GET THE CURRENT DATE AND TIME .**

**Q.WRITE A QUERY TO DISPLAY THE DATE AND TIME ZONE BY USING SQL;**

**SELECT SYSTIMESTAMP FROM DUAL;**

**SYSTIMESTAMP**

**02-SEP-24 11.35.13.364000 AM +05:30**

**XII) LAST\_DAY()**

**🡪IT IS USED TO GET THE LAST DAY OF A PERTICULAR DATE .**

**Q.WRITE A QUERY TO DISPLAY THE LAST DAY OF A PARTICULAR DATE;**

**SELECT LAST\_DAY(SYSDATE) FROM DUAL;**

**LAST\_DAY(**

**30-SEP-24**

**MULTI ROW FUNCTION**

🡪**IN MULTI ROW FUNCTION THE NUMBER OF OUTPUT IS FIXED BY ONE. IT MEANS IT DOES NOT DEPENDS UPON THE NUMBER OF INPUTS;**

🡪**THERE ARE FIVE TYPES OF MULTI ROW FUNCTION:-**

**I) MIN():- IT IS USED TO PROVIDE THE MINIMUM VALUE OF A GIVEN COLUMN;**

**II) MAX():- IT IS USED TO PROVIDE THE MAXIMUM VALUE OF A GIVEN COLUMN;**

**III) AVG():- IT IS USED TO PROVIDE THE AVERAGE VALUE OF A GIVEN COLUMN;**

**IV) SUM(:- IT IS USED TO OBTAIN THE SUMMATION OF VALUES OF A PARTICULAR COLUMN;**

**V) COUNT():- IT IS USED TO NUMBER THE VALUES OF A GIVEN COLUMN.**

**COUNT() IS THE ONLY FUNCTION WHICH CAN ACCEPT STAR(\*) AS AN ARGUMENT;**

🡪**MULTI ROW FUNCTION CAN ACCEPT ONLY ONE ARGUMENT AT A TIME.**

🡪**THE MULTI ROW FUNCTION IGNORE THE NULL VALUES.**

🡪**ALONG WITH MULTI ROW FUNCTION WE CAN'T USE ANY OTHER COLUMN NAME WITH SELECT CLAUSE.**

**Q.WRITE A QUERY TO DISPLAY THE MAXIMUM SALARY OF THE EMPLOYEES.**

**SELECT MAX(SAL) FROM EMP;**

**MAX(SAL)**

**5000**

**Q.WRITE A QUERY TO DISPLAY THE MINIMUM SALARY OF THE EMPLOYEES.**

**MIN(SAL)**

**800**

**SELECT MIN(SAL),ENAME FROM EMP;**

**ERROR AT LINE 1:**

**ORA-00937: NOT A SINGLE-GROUP GROUP FUNCTION**

**SELECT ENAME,MIN(SAL) FROM EMP;**

**ERROR AT LINE 1:**

**ORA-00937: NOT A SINGLE-GROUP GROUP FUNCTION**

**SELECT MAX(\*) FROM EMP;**

**ERROR AT LINE 1:**

**ORA-00936: MISSING EXPRESSION**

**SELECT MAX(SAL,EMPNO) FROM EMP;**

**\***

**ERROR AT LINE 1:**

**ORA-00909: INVALID NUMBER OF ARGUMENTS**

**Q. WRITE A QUERY TO DISPLAY THE AVERAGE SALARY OF THE EMPLOYEES.**

**SELECT AVG(SAL) FROM EMP;**

**AVG(SAL)**

**2073.21429**

**Q. WRITE A QUERY TO DISPLAY THE TOTAL SALARY GIVEN TO THE EMPLOYEES.**

**SELECT SUM(SAL) FROM EMP;**

**SUM(SAL)**

**29025**

**Q.WRITE A QUERY TO COUNT THE TOTAL NUMBER OF EMPLOYEE OF A GIVEN TABLE;**

**SELECT COUNT(ENAME) FROM EMP;**

**COUNT(ENAME)**

**14**

**SELECT COUNT(\*) FROM EMP;**

**COUNT(\*)**

**14**

**SELECT COUNT(COMM) FROM EMP;**

**COUNT(COMM)**

**4**

**SELECT COUNT(EMPNO) FROM EMP;**

**COUNT(EMPNO)**

**14**

**Q.WRITE A QUERY TO DISPLAY THE MAXIMUM SALARY GIVEN TO MANAGER.**

**SELECT MAX(SAL) FROM EMP WHERE JOB='MANAGER';**

**MAX(SAL)**

**2975 // MULTIROW FUNCTION CANNOT BE USED WITH WHERE CLAUSE MEANS WE CANNOT WRITE WHERE MAX(SAL);**

**Q.WRITE A QUERY TO DISPLAY THE TOTAL SALARY GIVEN TO THE EMPLOYEES OF DEPTNO 10;**

**SELECT SUM(SAL) FROM EMP WHERE DEPTNO=10;**

**SUM(SAL)**

**8750**

**Q.WRITE A QUERY TO DISPLAY THE NUMBER OF EMPLOYEES EARNING MORE THAN 1500 IN DEPTNO 20;**

**SELECT COUNT(\*) FROM EMP WHERE SAL>1500 AND DEPTNO=20;**

**COUNT(\*)**

**3**

**OR**

**SELECT COUNT(EMPNO) FROM EMP WHERE SAL>1500 AND DEPTNO=20;**

**COUNT(EMPNO)**

**3**

**Q.WRITE A QUERY TO COUNT THE NUMBER OF EMPLOYEES HAVING 'E' IN THEIR NAME;**

**SELECT COUNT(\*) FROM EMP WHERE ENAME LIKE'%E%';**

**COUNT(\*)**

**6**

**Q.WRITE A QUERY TO DISPLAY THE MINIMUM SALARY GIVEN TO THE EMPLOYEE WORKING AS 'CLERK' IN DEPTNO 10 OR 20.**

**SELECT MIN(SAL) FROM EMP WHERE JOB='CLERK' AND DEPTNO IN(10,20);**

**MIN(SAL)**

**800**

**Q.WRITE A QUERY TO DISPLAY THE NUMBER OF EMPLOYEES HIRED AFTER 1982 AND BEFORE 1988 IN DEPTNO 20 OR 30**

**SELECT COUNT(\*) FROM EMP WHERE HIREDATE BETWEEN '01-JAN-1983' AND '31-DEC-1987' AND DEPTNO IN(20,30);**

**OR**

**SELECT COUNT(\*) FROM EMP WHERE (HIREDATE > '31-DEC-1982' AND HIREDATE <'01-JAN-1988') AND DEPT IN(20,30);**

**COUNT(\*)**

**2**

**Q.WRITE A QUERY TO COUNT THE NUMBER OF EMPLOYEES ARE GETTING COMMISSION.**

**SELECT COUNT(\*) FROM EMP WHERE COMM IS NOT NULL;**

**OR**

**SELECT COUNT(\*) FROM EMP WHERE COMM >=0;**

**OR**

**SELECT COUNT(\*) FROM EMP WHERE COMM LIKE '%';**

**COUNT(\*)**

**4**

**Q.WRITE A QUERY TO DISPLAY THE MAXIMUM SALARY GIVEN TO THE EMPLOYEES WHO HAS 'S' IN THEIR NAMES AND WORKING AS 'MANAGER' IN DEPTNO 20 WITH SALARY MORE THAN 1800.**

**SELECT MAX(SAL) FROM EMP WHERE ENAME LIKE '%S%' AND JOB='MANAGER' AND DEPTNO=20 AND SAL>1800;**

**MAX(SAL)**

**2975**

**Q.WRITE A QUERY TO COUNT THE NUMBER OF EMPLOYEES WORKING IN DEPTNO 10 OR 30 AND GETTING SALARY WITHOUT COMMISSION.**

**SELECT COUNT(\*) FROM EMP WHERE DEPTNO IN(10,30) AND (COMM IS NULL AND SAL IS NOT NULL);**

**COUNT(\*)**

**5**

**Q.WRITE A QUERY TO DISPLAY THE MAXIMUM SALARY GIVEN TO 'SALESMAN' WORKING IN DEPTNO 20 OR 30 AND EARNING COMMISSION MORE THAN SALARY.**

**SELECT MAX(SAL) FROM EMP WHERE JOB='SALESMAN' AND DEPTNO IN(20,30) AND COMM>SAL;**

**MAX(SAL)**

**1250**

**GROUP BY CLAUSE**

🡪**IT IS USED TO GROUP THE RECORDS OR ROWS.**

🡪**THE GROUP BY CLAUSE EXECUTES ROW BY ROW**

🡪**THE GROUP BY CLAUSE CREATES THE GROUP SO THAT WE CAN FILTER OUR CONDITION**

🡪**THE GROUP BY CLAUSE CAN BE USED WITH OR WITHOUT WHERE CLAUSE**

**Q.WRITE A QUERY TO DISPLAY THE NUMBER OF EMPLOYEES WORKING IN EACH DEPARTMENT**

**SELECT COUNT(\*), DEPTNO FROM EMP GROUP BY DEPTNO;**

**COUNT(\*) DEPTNO**

**6 30**

**5 20**

**3 10**

**Q.WRITE A QUERY TO DISPLAY THE MAXIMUM SALARY GIVEN TO EACH JOB.**

**SELECT MAX(SAL), JOB FROM EMP GROUP BY JOB;**

**MAX(SAL) JOB**

**1300 CLERK**

**1600 SALESMAN**

**5000 PRESIDENT**

**2975 MANAGER**

**3000 ANALYST**

**Q.WRITE A QUERY TO DISPLAY THE NUMBER OF EMLOYEES WORKING IN EACH DEPARTMENT EXCEPT 'ANALYST'.**

**SELECT COUNT(\*), DEPTNO FROM EMP WHERE JOB!='ANALYST' GROUP BY DEPTNO;**

**COUNT(\*) DEPTNO**

**6 30**

**3 20**

**3 10**

**Q.WRITE A QUERY TO DISPLAY THE NUMBER OF EMLOYEES WORKING IN EACH JOB IF THE EMPLOYEES HAVING 'A' IN THEIR NAMES.**

**SELECT COUNT(\*),JOB FROM EMP WHERE ENAME LIKE '%A%' GROUP BY JOB;**

**COUNT(\*) JOB**

**3 SALESMAN**

**2 CLERK**

**2 MANAGER**

**Q.WRITE A QUERY TO DISPLAY THE NUMBER OF EMLOYEES GETTING COMMISSION IN EACH DEPARTMENT.**

**SELECT COUNT(\*),DEPTNO FROM EMP WHERE COMM IS NOT NULL GROUP BY DEPTNO;**

**COUNT(\*) DEPTNO**

**4 30**

**Q.WRITE A QUERY TO DISPLAY THE TOTAL SALARY NEEDED TO PAY TO THE EMPLOYEES IN EACH JOB.**

**SELECT SUM(SAL),JOB FROM EMP GROUP BY JOB;**

**SUM(SAL) JOB**

**4150 CLERK**

**5600 SALESMAN**

**5000 PRESIDENT**

**8275 MANAGER**

**6000 ANALYST**

**Q.WRITE A QUERY TO COUNT THE NUMBER OF EMPLOYEES WORKING AS 'MANAGER' IN EACH DEPARTMENT.**

**SELECT COUNT(\*),DEPTNO FROM EMP WHERE JOB='MANAGER' GROUP BY DEPTNO;**

**COUNT(\*) DEPTNO**

**1 30**

**1 20**

**1 10**

**Q.WRITE A QUERY TO DISPLAY THE AVERAGE SALARY NEEDED TO PAY ALL THE EMPLOYEES IN EACH DEPARTMENT NUMBER EXCEPT DEPARTMENT NUMBER 20.**

**SELECT AVG(SAL),DEPTNO FROM EMP WHERE DEPTNO!=20 GROUP BY DEPTNO;**

**OR**

**SELECT AVG(SAL),DEPTNO FROM EMP WHERE DEPTNO NOT IN 20 GROUP BY DEPTNO;**

**AVG(SAL) DEPTNO**

**1566.66667 30**

**2916.66667 10**

**HAVING CLAUSE**

**🡪IT IS USED TO FILTER THE GROUPS AND IT EXECUTES AFTER GROUP BY CLAUSE.**

**🡪WE CAN USE MULTI ROW FUNCTION IN HAVING CLAUSE.**

**🡪HAVING CLAUSE IS EXECUTED AFTER GROUP BY CLAUSE.**

**Q.WRITE A QUERY TO DISPLAY THE NUMBER OF EMLOYEES WORKING IN EACH DEPARTMENT WHICH CONTAINS ATLEAST FOUR EMPLOYEES.**

**SELECT COUNT(\*),DEPTNO FROM EMP GROUP BY DEPTNO HAVING COUNT(\*)>=4;**

**COUNT(\*) DEPTNO**

**6 30**

**5 20**

**Q.WRITE A QUERY TO DISPLAY THE JOB OF THE EMPLOYEES WHICH CONTAINS AT LEAST TWO EMPLOYEES.**

**SELECT COUNT(\*),JOB FROM EMP GROUP BY JOB HAVING COUNT(\*)>=2;**

**COUNT(\*) JOB**

**4 CLERK**

**4 SALESMAN**

**3 MANAGER**

**2 ANALYST**

**Q.WRITE A QUERY TO DISPLAY THE SALARY THAT ARE REPEATED.**

**SELECT COUNT(\*),SAL FROM EMP GROUP BY SAL HAVING COUNT(\*)>=2;**

**OR**

**SELECT COUNT(\*),SAL FROM EMP GROUP BY SAL HAVING COUNT(\*)>1;**

**COUNT(\*) SAL**

**2 1250**

**2 3000**

**Q.WRITE A QUERY TO COUNT THE NUMBER OF EMPLOYEES WORKING IN EACH DEPARTMENT HAVING ATLEAST TWO EMPLOYEES WHICH CONTAIN 'A' OR 'S' IN THEIR NAME.**

**SELECT COUNT(\*),DEPTNO FROM EMP WHERE ENAME LIKE '%A%' OR ENAME LIKE '%S%' GROUP BY DEPTNO HAVING COUNT(\*)>=2;**

**COUNT(\*) DEPTNO**

**5 30**

**4 20**

**Q. WRITE A QUERY TO DISPLAY THE NUMBER OF EMLOYEES EARNING SALARY MORE THAN 1200 IN EACH JOB AND THE TOTAL SALARY NEEDED TO PAY TO THE EMPLOYEES OF EACH JOB MUST EXCEEDS 3800.**

**SELECT COUNT(\*),JOB,SUM(SAL) FROM EMP WHERE SAL>1200 GROUP BY JOB HAVING SUM(SAL)>3800;**

**COUNT(\*) JOB SUM(SAL)**

**4 SALESMAN 5600**

**1 PRESIDENT 5000**

**3 MANAGER 8275**

**2 ANALYST 6000**

**SUBQUERY**

**🡪WHENEVER WE HAVE UNKNOWN CONDITION IN OUR QUERY THEN WE SHOULD GO FOR SUBQUERY.**

**🡪SUBQUERY CONTAIN TWO PARTS:- OUTER QUERY AND INNER QUERY**

**🡪THE OUTPUT OF THE INNER QUERY ACTS AS AN INPUT TO THE OUTER QUERY. IT MEANS THE INNER QUERY WILL GET EXECUTED FIRST**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMLOYEES THOSE ARE WORKING IN SAME DEPTNO AS 'SMITH'.**

**SELECT \* FROM EMP WHERE DEPTNO=(SELECT DEPTNO FROM EMP WHERE ENAME='SMITH');**

**------------------------------ -------------------------------------------**

**🡪OUTER QUERY 🡪INNER QUERY**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**7369 SMITH CLERK 7902 17-DEC-80 800 20**

**7566 JONES MANAGER 7839 02-APR-81 2975 20**

**7788 SCOTT ANALYST 7566 19-APR-87 3000 20**

**7876 ADAMS CLERK 7788 23-MAY-87 1100 20**

**7902 FORD ANALYST 7566 03-DEC-81 3000 20**

**-> WRITE A QUERY TO DISPLAY THE NAME AND SALARY OF THE EMPLOYEES THOSE ARE EARNING LESS THAN 'MILLER'.**

**SELECT ENAME,SAL FROM EMP WHERE SAL<(SELECT SAL FROM EMP WHERE ENAME='MILLER');**

**ENAME SAL**

**SMITH 800**

**WARD 1250**

**MARTIN 1250**

**ADAMS 1100**

**JAMES 950**

**Q.WRITE A QUERY TO DISPLAY THE NAME AND HIREDATE OF THE EMPLOYEES THOSE ARE HIRED AFTER 'JONES'.**

**SELECT ENAME,HIREDATE FROM EMP WHERE HIREDATE>(SELECT HIREDATE FROM EMP WHERE ENAME='JONES');**

**ENAME HIREDATE**

**MARTIN 28-SEP-81**

**BLAKE 01-MAY-81**

**CLARK 09-JUN-81**

**SCOTT 19-APR-87**

**Q.WRITE A QUERY TO DISPLAY THE NAME, SALARY AND DEPTNO OF EMPLOYEES THOSE ARE EARNING MORE THAN 2000 AND WORKING IN THE SAME DEPTNO AS 'CLARK'.**

**SELECT ENAME,SAL,DEPTNO FROM EMP WHERE SAL>2000 AND DEPTNO=(SELECT DEPTNO FROM EMP WHERE ENAME='CLARK');**

**ENAME SAL DEPTNO**

**CLARK 2450 10**

**KING 5000 10**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMPLOYEES WORKING IN THE SAME DESIGNATION AS 'MILLER' AND EARNING MORE THAN 1500.**

**SELECT \* FROM EMP WHERE JOB=(SELECT JOB FROM EMP WHERE ENAME='MILLER') AND SAL>1500;**

**NO ROWS SELECTED**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMPLOYEES EARNING MORE THAN 'SMITH' BUT LESS THAN 'CLARK'.**

**SELECT \* FROM EMP WHERE SAL>(SELECT SAL FROM EMP WHERE ENAME='SMITH') AND SAL<(SELECT SAL FROM EMP WHERE ENAME='CLARK');**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**7499 ALLEN SALESMAN 7698 20-FEB-81 1600 300 30**

**7521 WARD SALESMAN 7698 22-FEB-81 1250 500 30**

**7654 MARTIN SALESMAN 7698 28-SEP-81 1250 1400 30**

**7844 TURNER SALESMAN 7698 08-SEP-81 1500 0 30**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMPLOYEES THOSE ARE EARNING COMMISSION IN DEPTNO 30 AND EARNING SALARY MORE THAN 'WARD'.**

**SELECT \* FROM EMP WHERE COMM IS NOT NULL AND DEPTNO=30 AND SAL>(SELECT SAL FROM EMP WHERE ENAME='WARD');**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**7499 ALLEN SALESMAN 7698 20-FEB-81 1600 300 30**

**7844 TURNER SALESMAN 7698 08-SEP-81 1500 0 30**

**Q. WRITE A QUERY TO DISPLAY THE LIST OF THE EMPLOYEES THOSE ARE NOT WORKING AS 'MANAGER' OR 'CLERK' IN DEPTNO 20 OR 30 WITH A SALARY BETWEEN 1250 TO 2975.**

**SELECT \* FROM EMP WHERE JOB NOT IN('MANAGER','CLERK') AND DEPTNO IN(20,30) AND SAL BETWEEN 1250 AND 2975;**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**7499 ALLEN SALESMAN 7698 20-FEB-81 1600 300 30**

**7521 WARD SALESMAN 7698 22-FEB-81 1250 500 30**

**7654 MARTIN SALESMAN 7698 28-SEP-81 1250 1400 30**

**7844 TURNER SALESMAN 7698 08-SEP-81 1500 0 30**

**Q.WRITE A QUERY TO DISPLAY THE NAME OF THE EMPLOYEES WHOSE NAME CONTAIN ONLY FOUR CHARACTERS.**

**SELECT ENAME FROM EMP WHERE LENGTH(ENAME)=4;**

**ENAME**

**WARD**

**KING**

**FORD**

**Q.WRITE A QUERY TO COUNT THE NUMBER OF 'A' IN "PRABHUDATTA';**

**SELECT LENGTH('PRABHUDATTA')-LENGTH(REPLACE('PRABHUDATTA','A')) FROM DUAL;**

**LENGTH('PRABHUDATTA')-LENGTH(REPLACE('PRABHUDATTA','A'))**

**3**

**Q.WRITE A QUERY TO DISPLAY THE NAME OF JOBS ENDING WITH 'MAN';**

**SELECT JOB FROM EMP WHERE JOB LIKE '%MAN';**

**JOB**

**SALESMAN**

**SALESMAN // USING LIKE OPERATOR**

**SALESMAN**

**SALESMAN**

**SELECT JOB FROM EMP WHERE SUBSTR(JOB,-3)='MAN';**

**OR**

**SELECT JOB FROM EMP WHERE SUBSTR(JOB,-3,3)='MAN';**

**JOB**

**SALESMAN**

**SALESMAN**

**SALESMAN // WITHOUT USING LIKE OPERATOR**

**SALESMAN**

**WHENEVER THE DATA TO BE SELECTED AND THE CONDITION TO BE EXECUTED ARE PRESENT IN THE DIFFERENT TABLE AT THAT TIME WE SHOULD GO FOR SUBQUERY.**

**Q.WRITE A QUERY TO DISPLAY THE DEPTNO OF THE EMPLOYEES WHOSE NAME IS 'MILLER'.**

**SELECT DEPTNO FROM EMP WHERE ENAME='MILLER';**

**DEPTNO**

**10**

**Q.WRITE A QUERY TO DISPLAY THE DEPARTMENT NAME OF THE EMPLOYEES WHOSE NAME IS 'MILLER'.**

**SELECT DNAME FROM DEPT WHERE DEPTNO=(SELECT DEPTNO FROM EMP WHERE ENAME='MILLER');**

**DNAME**

**ACCOUNTING**

**Q.WAQTD THE LOCATION OF 'ADAMS'.**

**SELECT LOC FROM DEPT WHERE DEPTNO=(SELECT DEPTNO FROM EMP WHERE ENAME='ADAMS');**

**LOC**

**DALLAS**

**Q.WRITE A QUERY TO DISPLAY THE NAME OF THE EMPLOYEES WORKING IN 'NEW YORK'.**

**SELECT ENAME FROM EMP WHERE DEPTNO=(SELECT DEPTNO FROM DEPT WHERE LOC='NEW YORK');**

**ENAME**

**CLARK**

**KING**

**MILLER**

**Q.WRITE A QUERY TO DISPLAY THE NAME, SALARY OF THE EMPLOYEES WORKING AS 'MANAGER' IN 'ACCOUNTING' DEPARTMENT.**

**SELECT ENAME,SAL FROM EMP WHERE JOB='MANAGER' AND DEPTNO=(SELECT DEPTNO FROM DEPT WHERE DNAME='ACCOUNTING');**

**OR**

**SELECT ENAME,SAL FROM EMP WHERE DEPTNO=(SELECT DEPTNO FROM DEPT WHERE DNAME='ACCOUNTING') AND JOB='MANAGER';**

**ENAME SAL**

**CLARK 2450**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE EMPLOYEES WORKING IN THE SAME DESIGNATION AS 'MILLER' AND WORKS IN LOCATION 'NEW YORK'.**

**SELECT \* FROM EMP WHERE JOB=(SELECT JOB FROM EMP WHERE ENAME='MILLER') AND DEPTNO=(SELECT DEPTNO FROM DEPT WHERE LOC='NEW YORK');**

**EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO**

**7934 MILLER CLERK 7782 23-JAN-82 1300 10**

**Q.WRITE A QUERY TO DISPLAY THE NUMBER OF THE EMPLOYEES WORKING AS 'CLERK' IN THE SAME DEPTNO AS 'JAMES' AND EARNING MORE THAN 'ALLEN' AND HIRED AFTER 'WARD' IN LOCATION 'CHICAGO'.**

**SELECT ENAME FROM EMP WHERE JOB='CLERK' AND DEPTNO=(SELECT DEPTNO FROM EMP WHERE ENAME='JAMES') AND SAL>(SELECT SAL FROM EMP WHERE ENAME='ALLEN') AND HIREDATE>(SELECT HIREDATE FROM EMP WHERE ENAME='WARD') AND DEPTNO=(SELECT DEPTNO FROM DEPT WHERE LOC='CHICAGO');**

**COUNT(\*)**

**0**

**Q.WRITE A QUERY TO DISPLAY THE MAXIMUM SALARY GIVEN TO THE EMPLOYEES WORKING IN 'DALLAS'.**

**SELECT MAX(SAL) FROM EMP WHERE DEPTNO=(SELECT DEPTNO FROM DEPT WHERE LOC='DALLAS');**

**MAX(SAL)**

**3000**

**Q.WRITE A QUERY TO DISPLAY THE DETAILS OF THE DEPARTMENT TABLE WHOSE NAME IS 'SMITH'.**

**SELECT \* FROM DEPT WHERE DEPTNO=(SELECT DEPTNO FROM EMP WHERE ENAME = 'SMITH');**

**DEPTNO DNAME LOC**

**20 RESEARCH DALLAS**

**Q.WRITE A QUERY TO DISPLAY THE DEPARTMENT NAME AND LOCATION OF THE EMPLOYEES THOSE ARE GETTING ONLY SALARY 800.**

**SELECT DNAME,LOC FROM DEPT WHERE DEPTNO=(SELECT DEPTNO FROM EMP WHERE SAL=800);**

**DNAME LOC**

**RESEARCH DALLAS**

**Q.WRITE A QUERY TO DISPLAY THE NAME OF THE EMPLOYEES WHO IS EARNING MAXIMUM SALARY.**

**SELECT ENAME FROM EMP WHERE SAL=(SELECT MAX(SAL) FROM EMP);**

**ENAME**

**KING**

**Q.WRITE A QUERY TO DISPLAY THE SECOND MAXIMUM SALARY.**

**SELECT MAX(SAL) FROM EMP WHERE SAL<(SELECT MAX(SAL) FROM EMP);**

**MAX(SAL)**

**3000**

**Q.WRITE A QUERY TO DISPLAY THE THIRD MAXIMUM SALARY.**

**SELECT MAX(SAL) FROM EMP WHERE SAL<(SELECT MAX(SAL) FROM EMP WHERE SAL<(SELECT MAX(SAL) FROM EMP));**

**MAX(SAL)**

**2975**

**Q.WAQTD THE NAME AND ANNUAL SAL OF THE EMPLOYEES IF THEIR ANNUAL SALARY IS MORE THAN MAXIMUM ANNUAL SALARY OF THE 'SALESMAN'.**

**SELECT ENAME, SAL\*12 AS ANNUAL\_SAL FROM EMP WHERE SAL\*12>(SELECT MAX(SAL\*12) FROM EMP WHERE JOB='SALESMAN');**

**ENAME ANNUAL\_SAL**

**JONES 35700**

**BLAKE 34200**

**CLARK 29400**

**SCOTT 36000**

**KING 60000**

**FORD 36000**

**🡪 HOW TO DELETE THE TABLE FROM DATABASE.**

**WE CAN DELETE THE TABLE BY USING 'DROP' COMMAND.**

**SYNTAX :- DROP TABLE TABLE\_NAME;**

**🡪 HOW TO RECOVER TABLE FROM BIN.**

**SYNTAX :- FLASHBACK TABLE TABLE\_NAME TO BEFORE DROP;**

**🡪HOW TO DELETE TABLE FROM BIN FOLDER.**

**SYNTAX :- PURGE TABLE TABLE\_NAME;**

**JOIN**

**🡪IF WE WANT TO RETRIEVE DATA FROM MULTIPLE TABLE SIMULTANEOUSLY THEN WE SHOULD GO FOR JOIN**

**THERE ARE FOUR TYPES OF JOIN:-**

**I) CARTESIAN JOIN OR CROSS JOIN**

**II) INNER JOIN OR EQUI JOIN**

**III) OUTER JOIN**

**IV) NATURAL JOIN**

**CARTESIAN JOIN OR CROSS JOIN**

**🡪IN CROSS JOIN ALL THE RECORDS OF TABLE 1 WILL BE MERGED WITH EACH AND EVERY RECORDS OF TABLE 2.**

**SYNTAX :- SELECT COL\_NAME FROM TABLE1 CROSS JOIN TABLE2;**

**SELECT \* FROM STD1 CROSS JOIN STD2;**

**SID SNAME SID GENDE**

**1 A 5 E**

**1 A 2 F**

**1 A 6 G**

**1 A 4 H**

**2 B 5 E**

**2 B 2 F**

**2 B 6 G**

**2 B 4 H**

**3 C 5 E**

**3 C 2 F**

**3 C 6 G**

**3 C 4 H**

**4 D 5 E**

**4 D 2 F**

**4 D 6 G**

**4 D 4 H**

**OR**

**SYNTAX :- SELECT COL\_NAME FROM STD1,STD2;**

**SELECT \* FROM STD1,STD2;**

**SID SNAME SID GENDE**

**1 A 5 E**

**1 A 2 F**

**1 A 6 G**

**1 A 4 H**

**2 B 5 E**

**2 B 2 F**

**2 B 6 G**

**2 B 4 H**

**3 C 5 E**

**3 C 2 F**

**3 C 6 G**

**3 C 4 H**

**4 D 5 E**

**4 D 2 F**

**4 D 6 G**

**4 D 4 H**

**🡪IN CROSS JOIN THE NUMBER OF COLUMN IN THE RESULTANT TABLE IS EQUIVALENT TO THE SUMMATION OF TABLE1 COLUMN AND TABLE2 COLUMN.**

**🡪IN CROSS JOIN THE NUMBER OF ROWS IN THE RESULTANT TABLE WILL BE EQUIVALENT TO THE PRODUCT OF TABLE1 ROW AND TABLE2 ROW.**

**SELECT STD1.SID FROM STD1 CROSS JOIN STD2;**

**SID**

**1**

**1**

**1**

**1**

**2**

**2**

**2**

**2**

**3**

**3**

**3**

**3**

**4**

**4**

**4**

**4**

**SELECT STD1.SID,GENDER FROM STD1 CROSS JOIN STD2;**

**SID GENDE**

**1 E**

**2 E**

**3 E**

**4 E**

**1 F**

**2 F**

**3 F**

**4 F**

**1 G**

**2 G**

**3 G**

**4 G**

**1 H**

**2 H**

**3 H**

**4 H**

**INNER JOIN OR EQUI JOIN**

**🡪IT IS USED TO OBTAIN THE MATCHED RECORDS FROM BOTH THE TABLE.**

**SYNTAX :- SELECT COL\_NAME FROM TABLE1 INNER JOIN TABLE2 ON JOIN CONDITION;**

**SELECT \* FROM STD1 INNER JOIN STD2 ON STD1.SID=STD2.SID;**

**SID SNAME SID GENDE**

**2 B 2 F**

**4 D 4 H**

**OR**

**SYNTAX :- SELECT COL\_NAME FROM TABLE1,TABLE2 WHERE JOIN CONDITION;**

**SELECT \* FROM STD1,STD2 WHERE STD1.SID=STD2.SID;**

**SID SNAME SID GENDE**

**2 B 2 F**

**4 D 4 H**

**Q.WRITE A QUERY TO DISPLAY THE EMPLOYEE NAME AND DEPARTMENT NAME OF EMPLOYEE.**

**SELECT ENAME,DNAME FROM EMP,DEPT WHERE EMP.DEPTNO=DEPT.DEPTNO;**

**ENAME DNAME**

**SMITH RESEARCH**

**ALLEN SALES**

**WARD SALES**

**JONES RESEARCH**

**MARTIN SALES**

**BLAKE SALES**

**Q.WRITE A QUERY TO DISPLAY THE NAME OF THE EMPLOYEES AND LOCATION IF THEY ARE WORKING AS 'MANAGER'.**

**SELECT ENAME,LOC FROM EMP,DEPT WHERE EMP.DEPTNO=DEPT.DEPTNO AND EMP.JOB='MANAGER';**

**OR**

**SELECT ENAME,LOC FROM EMP,DEPT WHERE EMP.DEPTNO=DEPT.DEPTNO AND JOB='MANAGER';**

**ENAME LOC**

**JONES DALLAS**

**BLAKE CHICAGO**

**CLARK NEW YORK**

**Q. WRITE A QUERY TO DISPLAY THE EMPLOYEE NAME, SALARY AND DEPARTMENT NUMBER OF EMPLOYEES WORKING AS 'CLERK' IN DEPT NUMBER 20 AND EARNING SALARY MORE THAN 950.**

**SELECT ENAME,EMP.DEPTNO,DNAME FROM EMP,DEPT WHERE EMP.DEPTNO=DEPT.DEPTNO AND JOB='CLERK' AND EMP.DEPTNO=20 AND SAL>950;**

**ENAME DEPTNO DNAME**

**ADAMS 20 RESEARCH**

**Q.WRITE A QUERY TO DISPLAY THE NAME OF THE EMPLOYEES,DEPTNO,DNAME,LOCATION OF THE EMPLOYEES AND EARNING MORE THAN 'MILLER' IN 'CHICAGO'.**

**SELECT ENAME,EMP.DEPTNO,DNAME,LOC FROM EMP,DEPT WHERE EMP.DEPTNO=DEPT.DEPTNO AND SAL>(SELECT SAL FROM EMP WHERE ENAME='MILLER') AND LOC='CHICAGO';**

**ENAME DEPTNO DNAME LOC**

**ALLEN 30 SALES CHICAGO**

**BLAKE 30 SALES CHICAGO**

**TURNER 30 SALES CHICAGO**

**OUTER JOIN**

**🡪IT IS USED TO RETRIEVE THE BOTH MATCH AND UNMATCH RECORDS.**

**THERE ARE TYPES OF OUTER JOIN:-**

**I) LEFT OUTER JOIN**

**🡪IT IS USED TO RETRIEVE THE UNMATCHED RECORDS OF LEFT TABLE ALONG WITH THE MATCHED RECORDS**

**SYNTAX :- SELECT COLUMN\_NAME FROM TABLE1 LEFT JOIN TABLE2 ON JOIN\_CONDITION;**

**SELECT \* FROM STD1 LEFT JOIN STD2 ON STD1.SID=STD2.SID;**

**SID SNAME SID GENDE**

**2 B 2 F**

**4 D 4 H**

**3 C**

**1 A**

**II) RIGHT OUTER JOIN**

**🡪IT IS USED TO OBTAIN THE UNMATCHED RECORDS OF RIGHT TABLE ALONG WITH THE MATCHED RECORDS.**

**SYNTAX :- SELECT COLUMN\_NAME FROM TABLE1 RIGHT JOIN TABLE2 ON JOIN\_CONDITION;**

**SELECT \* FROM STD1 RIGHT JOIN STD2 ON STD1.SID=STD2.SID;**

**SID SNAME SID GENDE**

**2 B 2 F**

**4 D 4 H**

**6 G**

**5 E**

**III) FULL OUTER JOIN**

**🡪IT IS USED TO OBTAIN THE UNMATCHED RECORDS OF BOTH LEFT AND RIGHT TABLE ALONG WITH THE MATCHED RECORDS.**

**SYNTAX :- SELECT COLUMN\_NAME FROM TABLE1 FULL JOIN TABLE2 ON JOIN\_CONDITION;**

**SELECT \* FROM STD1 FULL JOIN STD2 ON STD1.SID=STD2.SID;**

**SID SNAME SID GENDE**

**2 B 2 F**

**4 D 4 H**

**3 C**

**1 A**

**6 G**

**5 E**

**NATURAL JOIN**

**🡪IT BEHAVES AS INNER JOIN IF THERE IS A RELATION BETWEEN TABLES OTHERWISE IT ACTS AS CROSS JOIN.**

**SYNTAX :- SELECT COLUMN\_NAME FROM TABLE1 NATURAL JOIN TABLE2;**

**SELECT \* FROM STD1 NATURAL JOIN DEPT;**

**SID SNAME DEPTNO DNAME LOC**

**1 A 10 ACCOUNTING NEW YORK**

**2 B 10 ACCOUNTING NEW YORK**

**3 C 10 ACCOUNTING NEW YORK**

**4 D 10 ACCOUNTING NEW YORK**

**1 A 20 RESEARCH DALLAS**

**2 B 20 RESEARCH DALLAS**

**3 C 20 RESEARCH DALLAS**

**4 D 20 RESEARCH DALLAS**

**1 A 30 SALES CHICAGO**

**2 B 30 SALES CHICAGO**

**3 C 30 SALES CHICAGO**

**4 D 30 SALES CHICAGO**

**1 A 40 OPERATIONS BOSTON**

**2 B 40 OPERATIONS BOSTON**

**3 C 40 OPERATIONS BOSTON**

**4 D 40 OPERATIONS BOSTON**

**DML**

**🡪IT IS USED TO MANIPULATE THE OBJECT BY PERFORMING SOME OPERATIONS LIKE INSERTION,DELETION AND UPDATION.**

**🡪THERE ARE THREE STATEMENTS OF DML:-**

**I) INSERT**

**🡪 IT IS USED TO RECORDS OR ROWS INSIDE THE TABLE.**

**II) UPDATE**

**🡪 IT IS USED TO MODIFY THE EXISTING VALUE;**

**SYNTAX :-UPDATE TABLE\_NAME SET COLUMN\_NAME=VALUE WHERE CONDITION;**

**UPDATE STD1 SET SNAME='ABC' WHERE SID=4;**

**SELECT \* FROM STD1;**

**SID SNAME**

**1 A**

**2 B**

**3 C**

**4 ABC**

**UPDATE STD1 SET SNAME='DEF' WHERE SID=1;**

**1 ROW UPDATED.**

**SELECT \* FROM STD1;**

**SID SNAME**

**1 DEF**

**2 B**

**3 C**

**4 ABC**

**III) DELETE**

**🡪 IT IS USED TO DELETE THE RECORDS OR ROWS.**

**SYNTAX :- DELETE FROM TABLE\_NAME WHERE CONDITION;**

**SQL> DELETE FROM STD1 WHERE SID=1;**

**1 ROW DELETED.**

**SQL> SELECT \* FROM STD1;**

**SID SNAME**

**2 B**

**3 C**

**4 ABC**

**SQL> DELETE FROM STD1 WHERE SNAME='ABC';**

**1 ROW DELETED.**

**SQL> SELECT \* FROM STD1;**

**SID SNAME**

**2 B**

**3 C**

**SQL> ROLLBACK;**

**ROLLBACK COMPLETE.**

**SQL> SELECT \* FROM STD1;**

**SID SNAME**

**1 A**

**2 B**

**3 C**

**4 D**

**TCL**

**🡪THERE ARE THREE STATEMENTS OF TCL:-**

**I)COMMIT**

**🡪IT IS USED TO SAVE THE OPERATION**

**II)ROLLBACK**

**🡪IT IS USED TO OBTAIN THE SAVED DATA FROM THE DATABASE AND IT WILL BRING YOU TO THE POINT WHERE YOU HAVE COMMITTED FOR THE LAST TIME.**

**SQL> INSERT INTO STD1 VALUES(4,'D');**

**1 ROW CREATED.**

**SQL> SELECT \* FROM STD1;**

**SID SNAME**

**1 A**

**1 B**

**2 B**

**3 C**

**4 D**

**SQL> ROLLBACK;**

**ROLLBACK COMPLETE.**

**SQL> SELECT \* FROM STD1;**

**SID SNAME**

**1 A**

**1 B**

**III)SAVEPOINT**

**🡪IT IS USED TO MARK THE POSITION.**

**SYNTAX :- SAVEPOINT SAVEPOINT\_NAME;**

**DCL**

**🡪IT IS USED TO CONTROL THE FLOW OF DATA BETWEEN THE USERS.**

🡪**THERE ARE TWO STATEMENTS OF DCL:-**

**I)GRANT**

**🡪THE GRANT STATEMENT IS USED TO GIVE PERMISSION TO THE USER.**

**II)REVOKE**

**🡪THE REVOKE STATEMENT IN SQL IS USED TO TAKE BACK THE PERMISSION FROM THE USER.**

**CONSTRAINTS**

**🡪IT IS USED TO PROVIDE THE EXTRA FUNCTIONALITY TO THE COLUMN.**

**I)UNIQUE**

**🡪IT IS USED TO AVOID THE DUPLICATE VALUE.**

**II)NOT NULL**

**🡪IT IS USED TO AVOID THE NULL VALUES.**

**III)PRIMARY KEY**

🡪**IT IS THE COMBINATION OF UNIQUE PLUS NOT NULL CONSTRAINT.**

🡪**A TABLE CAN HAVE AT MOST ONE PRIMARY KEY.**

**IV)FOREIGN KEY**

🡪**IT IS NOT THE COMBINATION OF UNIQUE AND NOT NULL CONSTRAINT.**

🡪**TO BECOME A FOREIGN KEY IN ANOTHER TABLE IT MUST HAVE THE PRIMARY KEY OF ITS OWN TABLE.**

🡪**A TABLE CAN HAVE MULTIPLE FOREIGN KEY.**

**V)CHECK**

**🡪IT IS USED TO PROVIDE THE EXTRA CONDITION TO THE VALUES.**