1. **List all employee names along with their manager names. Make sure to print the employee who has no manager**

select E.EMPNO, E.ENAME, E.MGR from SCOTT.emp E;

1. **List the employee name, salary, departent for those employees who earn salary greater than average salary for their department and sort the output in department number order.**

SELECT E.EMPNO, E.ENAME,E.SAL, E.DEPTNO FROM SCOTT.emp E

WHERE e.sal >

ALL(SELECT avg(e1.sal)FROM SCOTT.emp e1 where e1.DEPTNO = e.DEPTNO GROUP BY e1.DEPTNO) group by e.DEPTNO,E.EMPNO, E.ENAME,E.SAL order by e.DEPTNO ,e.sal;

1. **List the employee details if and only if more than 3 employees are present in department 20**

SELECT \*

FROM SCOTT.emp e

WHERE e.DEPTNO IN

(SELECT e.DEPTNO

FROM SCOTT.emp e

GROUP BY e.DEPTNO

HAVING COUNT(e.DEPTNO) > 3) and e.DEPTNO =20 ;

1. **List details of employee earning more than the highest paid MANAGER**

select \* from SCOTT.emp e where e.sal > (select max(e.sal) from SCOTT.emp e where e.job = 'MANAGER');

1. **List all employee details who don’t manage any one.**

select \* from SCOTT.emp e left join SCOTT.emp y on e.empno = y.mgr where y.mgr is null;

**In a SELECT query, what is the difference between a WHERE clause and a HAVING clause?**

WHERE:

1. When we want to filter anything based on the row ,where clause is used

2. it is used before the group by clause

3. Update and Delete clause can be used with where clause

Having

1. When we want to filter anything based on a group , having clause is used

2. it is used after the group by clause

3. Update and delete clause cannot be used with having clause

**What type of integrity is enforced when a primary key is declared?**

Entity integrity i.e. primary key cannot contain NULL values and it should be unique so that they don’t match the primary key of any other row in the table. Primary keys are basically unique identifiers for rows in the table. Entity integrity makes sure all the rows can be distinguished since the Primary keys are basically unique identifiers for rows.

**Explain why the two following commands produce different results:**

SELECT DISTINCT COUNT (SAL) FROM EMP – It primarily calculates and gives the count the of numbers in the SAL column. But the duplicate numbers will also be the part of this count. So now the DISTINCT will give us only one row for the duplicate row (duplicate values).

SELECT COUNT (DISTINCT SAL) FROM EMP- It will count the unique values in the column SAL.

Example below will illustrate both the commands in a better way

Suppose you write the query SELECT SAL FROM EMP and it gives the following table

2000 2000 3500 2000 3500 2000 3500 2000

Now, if we give the query SELECT DISTINCT COUNT (SAL) FROM EMP, the output will return ‘8’. And if we give the query SELECT COUNT (DISTINCT SAL) FROM EMP, the output will return ‘2’.

What **three join types are included in the OUTER JOIN classification? Explain with example.**

Outer join is a join that provides the output for all the matching value between 2 tables and for the unmatched values as well.

Three types of OUTER JOIN are:

LEFT OUTER JOIN:

This will yield the results for all the matching rows between the two joined tables and all the unmatched rows from the left table.

RIGHT OUTER JOIN:

This will yield the results for all the matching rows between the two joined tables and all the unmatched rows from the RIGHT table.

FULL OUTER JOIN:

This will yield the results for all the matching rows between the two joined tables and all the unmatched rows from both the tables.

To explain all the above 3 types of outer join lets consider the below Example

Table ***Store***

|  |  |  |
| --- | --- | --- |
| **Store\_Name** | **Sales** | **Txn\_Date** |
| Los Angeles | 1500 | Jan-05-1999 |
| San Diego | 250 | Jan-07-1999 |
| Los Angeles | 300 | Jan-08-1999 |
| Boston | 700 | Jan-08-1999 |

Table ***Region***

|  |  |
| --- | --- |
| **Region\_Name** | **Store\_Name** |
| East | Boston |
| East | New York |
| West | Los Angeles |
| West | San Diego |

**LEFT OUTER JOIN**

Select A1.STORE\_NAME STORE, SUM(A2.Sales) SALES from Region A1 left outer join Store A2 on A1.Store\_Name = A2.Store\_Name

|  |  |
| --- | --- |
| **STORE** | **SALES** |
| **Los Angeles** | **1800** |
| **San Diego** | **250** |
| **New York** | **NULL** |
| **Boston** | **700** |

Right OUTER JOIN

Select A1.STORE\_NAME STORE, REGION\_NAME from Region A1 Right outer join Store A2 on A1.Store\_Name = A2.Store\_Name

|  |  |
| --- | --- |
| **STORE** | **REGION\_NAME** |
| **Los Angeles** | **WEST** |
| **San Diego** | **WEST** |
| **New York** | **WEST** |
| **Boston** | **EAST** |

FULL OUTER JOIN

Select A1.STORE\_NAME STORE, SALES from Region A1 Full outer join Store A2 on A1.Store\_Name = A2.Store\_Name

|  |  |  |
| --- | --- | --- |
| **STORE** | **SALES** | |
| **Los Angeles** | **1500** |
| **San Diego** | **250** |
| **Los Angeles** | **300** |
| **New York** | **NULL** |
| **Boston** | **700** |

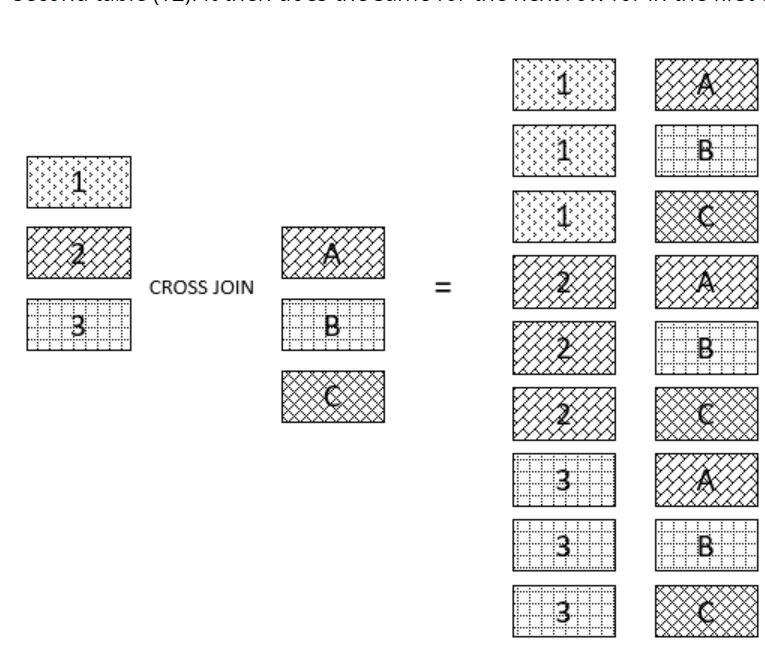
**What string function should you use to list the first three characters of employee Name from EMP table?**

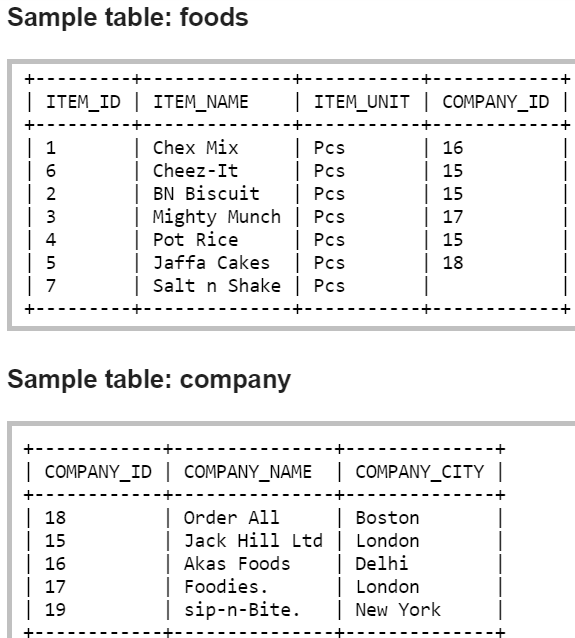
SELECT SUBSTRING(E.ENAME,1,3)

FROM SCOTT.emp e;

**What is a CROSS JOIN? Give an example of its syntax**

the Cross Join yields a dataset that contains the number of rows in first Table multiplied by the numbet of rows in the second table which is also called as cartesian join.





SELECT foods.item\_name,foods.item\_unit,

company.company\_name,company.company\_city

FROM foods

CROSS JOIN company;

ITEM\_NAME ITEM\_UNIT COMPANY\_NAME COMPANY\_CITY

--------------- ---------- --------------- ---------------

Chex Mix Pcs Order All Boston

Cheez-It Pcs Order All Boston

BN Biscuit Pcs Order All Boston

Mighty Munch Pcs Order All Boston

Pot Rice Pcs Order All Boston

Jaffa Cakes Pcs Order All Boston

Salt n Shake Pcs Order All Boston

Chex Mix Pcs Jack Hill Ltd London

Cheez-It Pcs Jack Hill Ltd London

BN Biscuit Pcs Jack Hill Ltd London

Mighty Munch Pcs Jack Hill Ltd London

Pot Rice Pcs Jack Hill Ltd London

Jaffa Cakes Pcs Jack Hill Ltd London

Salt n Shake Pcs Jack Hill Ltd London

Chex Mix Pcs Akas Foods Delhi

Cheez-It Pcs Akas Foods Delhi

BN Biscuit Pcs Akas Foods Delhi

Mighty Munch Pcs Akas Foods Delhi

Pot Rice Pcs Akas Foods Delhi

Jaffa Cakes Pcs Akas Foods Delhi

Salt n Shake Pcs Akas Foods Delhi

Chex Mix Pcs Foodies. London

**What is a correlated subquery? Give an example.**

Correlated subquery is called as synchronized subquery. It is a query nested inside another

query that will use the resultant values from the outer query.

