1- What is the difference between Single and Double –Ampersand Substitution Variable ? Give one example each

Single Ampersand Substitution Variable ():

&

```
SELECT *
FROM Employees
WHERE Department = '&department_name';
```

Double Ampersand Substitution Variable ()

&&

```
SELECT *
FROM Employees
WHERE Department = '&&department_name';
```

2- Write SQL Statement to show below result : Show difference JOB_ID From table Employee who has no commission

```
SELECT DISTINCT Job_ID
FROM Employees
WHERE Commission IS NULL;
```

3- Write SQL statement to show below result : FullName (FirstName – LastName), HireDate, Salary, DepartmentID From Table Employees who Firstname started with A , B, C and they must have salary greater than 6500.

```
SELECT CONCAT(FirstName, ' ', LastName) AS FullName, HireDate, Salary,
DepartmentID FROM Employees
WHERE (FirstName LIKE 'A%' OR FirstName LIKE 'B%' OR FirstName LIKE 'C%')
And salary > 6500
```

4 -Show Top 20 Staff information (FirstName,Salary,Job_ID) who has lowest Salary

```
SELECT FirstName, Salary, Job_ID
FROM Employees
ORDER BY Salary ASC
LIMIT 20;
```

5. How many type of Sort you could use ? What are they in Oracle

-Bubble Sort

-Selection Sort

-Insertion Sort

-Merge Sort

-Quick Sort

-Heap Sort

-Radix Sort

-Bucket Sort

-Counting Sort

6- Show Employee information (FullName,Salary,Manager_ID,HireDate,Job_ID) who Job_ID difference from IT_Prog, SA_REP,MK_REP and Salary must in the Rank of 5500 and 20000 and Hiredate in the year of 2005-2006-2007

```
SELECT CONCAT(FirstName, ' ', LastName) AS FullName, Salary, Manager_ID, HireDate,
Job_ID
FROM Employees
WHERE Job_ID NOT IN ('IT_Prog', 'SA_REP', 'MK_REP')
  AND Salary BETWEEN 5500 AND 20000
  AND YEAR(HireDate) IN (2005, 2006, 2007);
```

7-Show Employees FirstName , LastName, Salary, Hiredate, ManagerID who join the company in 2000,2005,2007 and their firstName start with A, B, C

```
SELECT FirstName, LastName, Salary, HireDate, ManagerID
FROM Employees
WHERE YEAR(HireDate) IN (2000, 2005, 2007)
```

```
AND FirstName LIKE 'A%' OR FirstName LIKE 'B%' OR FirstName LIKE 'C%';
```

13. write sql statement

```
SELECT
  job_id,
  COUNT(*) AS num_staffs
FROM
  employees
GROUP BY
  job_id
HAVING
  COUNT(*) > 10;
```

14. How many department that has employee less than 10

```
SELECT
  department_name,
  COUNT(*) AS num_emps
FROM
  employees
GROUP BY
  department_name
HAVING
  COUNT(*) < 10;
```

15. How many employee who join the company from May to Sep

```
SELECT
  COUNT(*) AS num_emps
FROM
  employees
WHERE
  hire_date BETWEEN TO_DATE('2023-05-01', 'YYYY-MM-DD') AND TO_DATE('2023-09-30', 'YYYY-MM-DD');
```

16. Show Manager Name and number of Employee under him

```
SELECT
  manager_name,
  COUNT(*) AS num_emps
FROM
  employees
GROUP BY
  manager_id;
```

17. what is the Annual Expend for each department (format the total Expend as $9,999,999.00)

```
SELECT
  department_name,
  SUM(salary) AS annual_expend
FROM
  employees
GROUP BY
  department_name
ORDER BY
  annual_expend DESC;
```

18. What is the difference between Where and Having?

A. The WHERE clause is used to filter the rows that are returned by the SELECT statement.

   The HAVING clause is used to filter the groups that are returned by the GROUP BY clause.

19. Salary > ALL (200,400,800) , Salary > ?, Salary <Any (300,400,600), Salary < ?

```
SELECT *
FROM employees
WHERE
 salary > ALL (200,400,800)
 AND salary > ?
 AND salary < ANY (300,400,600)
 AND salary < ?;
```

20. What does Distinct Use for? Give one Example of it

The DISTINCT keyword is used to remove duplicate rows from the results of a SELECT statement. For example, the following SELECT statement will return all the departments in the employees table:

```
SELECT department_name
FROM employees;
```

21. What is NonEquiJoins ?Give one Example of it

A non-equi join is a join that does not use an equality comparison operator (=) in the join condition. For example, the following is a non-equi join:

```
SELECT *
FROM employees
JOIN departments
ON employees.department_id = departments.department_id
WHERE employees.salary > departments.average_salary;
```

22. When do we use Cross Join? Give one case of using it.

A cross join is a type of join that returns the Cartesian product of two tables. The Cartesian product is the set of all possible pairs of rows from the two tables. For example, if two tables have 3 and 4 rows respectively, then the Cartesian product will have 12 rows.

```
SELECT *
FROM employees
CROSS JOIN departments
```

Multiple Choise

1-Chose the correct answer below
a) Round (40.9933,0) →40.90
b) Trunc (33.22,1) →33.2
c) Mod (2999,3) →3c

2- Which function use to Compare two expressions and returns null if they are equal (4)

a) NVL

b) NVL2

c) NULLIF

d) COALECE

3-What does Sysdate function used for?

1. A-Return current date and time from server

2. B-Return system number

3. C-Return date time from your application

4. D-All are correct

4-what is the correct statement used to return current date time from your database

- A-Select getdate() from dual
- B-Select Today() from dual
- C-Select Now() from dual
- D-Select Sysdate from dual

5-What is the correct syntax to add hour in your date

- • A-date+number/24
- • B-date+Convert(hour)/24
- • C-date+number of hour
- • D-all are wrong

6-When we subtract two dates what is the result ? Sysdatedate-hiredate

- • A-number of day
- • B-number of weeks
- • C-number of month
- • D-number of year

7-(Sysdate-hire_date)/7

- • A-number of day
- • B-number of weeks
- • C-number of month
- • D-number of year

8-What function return last day of the month

- • A-Last_day
- • B-Next_day
- • C-Last_day_month
- • D-Months_between

9-Which of the following is True about CASE SQL Statement?

- • A-A way to establish and IF-THEN-ELSE is SQL
- • B-A way to establish a loop in SQL
- • C-A way to establish a data definition in SQL
- • D-All of the above

10-Which below format define Minute formatting

- • A-MI
- • B-MM
- • C-Minute
- • D-MM

11-What does SPTH mean?

- • A-Spelled out number
- • B-Ordinal number
- • C-Spelled out Ordinal number

12-Which Function below is complies with the ANSI SQL

- • CASE
- • DECODE
- • BOTH