

## Interview Questions for Accenture:

I have written the popular articles on SQL Questions for [Deloitte technologies](#) as well as [Infosys Software](#). I have studied lot of Websites and i myself have experienced the SQL interview for Accenture and come up with the set of Interview Questions for Accenture. Accenture is well known US based organization which will always well known for its work life balance.

Following are some Interview Questions for Accenture which will ask for SQL,PL SQL,ETL developer Interview:

### 1.What is SQL?(100 % Interview Questions for Accenture )

**Answer:** SQL Stands for Structured Query Language which is specially designed to communicate with databases. SQL pronounced as Sequel is very widely used language in most of the database management systems like Oracle, Mysql, Postgresql etc. SQL provides us a simple and efficient way of reading, writing, executing the data from the system. This is one of the SQL Interview Question ever asked in interviews

### 2.What is the use of NVL function in Oracle?(80% asked Interview Questions for Accenture )

**Answer:** NVL function is most important function to replace null value with another value.

Example: `select NVL(null,'Amit') from dual;` which will give you output as Amit.

### 3.What is Correlated Subquery? Explain with the steps of execution with example.

**Answer:**

Correlated query is the query which is executed after the outer query is executed. The outer query is always dependent on inner query. The approach of the correlated subquery is bit different than normal subqueries. In normal subqueries the inner queries are executed first and then the outer query is executed but in Correlated Subquery outer query is always dependent on inner query so first outer query is executed then inner query is executed. Correlated Subqueries always uses operator like **Exist, Not Exist, IN, Not IN**. (Source-[click here](#))

**Example:**

*Select \* from Employee E where Not exist*

*(Select Department\_no From Department D where E.Employee\_id=D.Employee\_ID);*

**Execution of query:**

**Step 1:**

Select \* from Employee E ;

It will fetch the all employees

**Step 2:**

The First Record of the Employee second query is executed and output is given to first query.

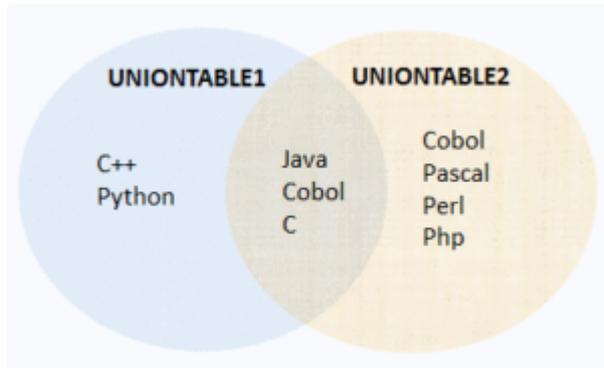
(Select Department\_no From Department D where E.Employee\_id=D.Employee\_ID);

Step 3:

Step 2 is repeated until and unless all output is been fetched. (Source-[Click here](#))

**4.What is difference Between UNION and UNION All?(90 % asked in Interview Questions for Accenture )**

[caption id="attachment\_225" align="aligncenter" width="300"]



Union and Union all[/caption]

**Answer:**

Union

1.Union Set operator is used to fetch the records from 2 different tables which eliminates the duplicate records

2.Syntax:

Select col1,col2...from table1;

Union

Select col1,col2...from table2;

3.For Performance tuning Union operator is not preferable as it takes time to eliminate duplicate records

Union ALL

1.Union all Set operator is used to fetch the records from 2 different tables which does not eliminates the duplicate records

2.Syntax:

Select col1,col2...from table1;

Union

Select col1,col2...from table2;

3.Union all is preferable operator in Performance tuning.

## 5.What is raw datatype?

### Answer:

Raw datatype is used to store values in binary data format.

There are 2 types of RAW datatype:

1.Raw

2.Long Raw.

Long raw datatype is used to store graphics,sound documents.

Raw datatype is variable length datatype like varchar2 but basically it only stores data in 1's and 0's means binary data format.

## 6.What is materialised view and what are its different fields while creating it?

### Answer :

Materialized view is also logical structure of one or more table in which data is stored physically in the view.Data has been stored physically in materialized view so data retrieval is faster as compare to simple view.

There are following different options we used to create materialized view or snapshot.

### 1.Build Immediate:

Means materialized views(mv) created immediately.

### 2.Build Deferred:

Means materialized views(mv) created after one refresh.

### 3.Refresh on commit:

This option committed the data in materialized view in SQL immediately after data inserted and committed in table.This option is known as incremental refresh option.View is not fully refreshed with this option

### 4.Refresh on Demand:

Using this option you can add the condition for refreshing data in materialized views.

## 7.Explain the difference between view and materialized view?

### Answer :

## View

1.View is nothing but the logical structure of the table which will retrieve data from 1 or more table.

2.You need to have Create view privileges to create simple or complex view

3.Data access is not as fast as materialized views

4.There are 2 types of views:

1.Simple View

2.Complex view

5.In Application level views are used to restrict data from database

## Materialized Views(Snapshots)

1.Materialized views(Snapshots) are also logical structure but data is physically stored in database.

2.You need to have create materialized view 's privileges to create Materialized views

3.Data retrieval is fast as compare to simple view because data is accessed from directly physical location

4.There are following types of Materialized views:

1.Refresh on Auto

2.Refresh on demand

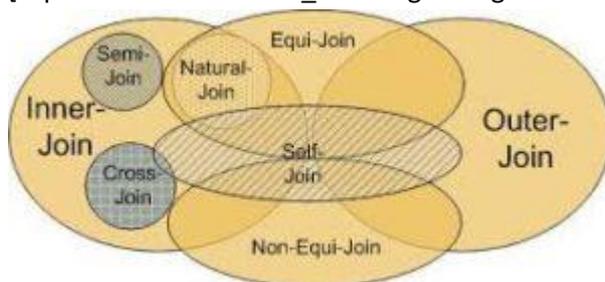
5.Materialized Views are used in Data Warehousing.

## 8.Explain different Joins in SQL?

### Answer:

Join is nothing but connecting 2 tables to fetch the records from 2 or more different tables.There are following types of joins in SQL:

[caption id="attachment\_202" align="aligncenter" width="300"]



Joins in SQL[/caption]

### 1.Inner join:

Inner join retrieves the records which are common between 2 or more tables.

### 2.Outer join:

Outer join retrieves the common records from the table as well as uncommon records from Left or right table.

### 2.1.Left outer join:

When user needs to fetch all data from left table and common records from left and right table then the join is called as left outer join.

### 2.2.Left outer join:

When user needs to fetch all data from right table and common records from left and right table then the join is called as right outer join.

### 2.3.Full Outer Join:

When user needs to fetch the data from both the tables and common records from both of the tables.

### 3.Cross join/Cartesian join:

When each and every record is connected to each and every record from other table then it is called as cross join or Cartesian join.

[Click Here to get information about SQL joins..](#)

### 9.What is mean by complete refresh in materialized view?

#### Answer:

Without a materialized view log, Oracle Database must re-execute the materialized view query to refresh the materialized view. This process is called **a complete refresh**. Usually, a fast refresh takes less time than a complete refresh. A materialized view log is located in the master database in the same schema as the master table. A master table can have only one materialized view log defined on it. Oracle Database can use this materialized view log to perform fast refreshes for all fast-refreshable materialized views based on the master table. To fast refresh a materialized join view, you must create a materialized view log for each of the tables referenced by the materialized view.

### 10.What is the query to fetch number of employees departmentwise?

#### Answer:

```
select count(EMPNO), b.deptno, dname from emp a, dept b where a.deptno(+) = b.deptno
group by b.deptno, dname;
```

### 11.What is the result of following query?

```
select case when null = null then 'Amit' else 'Rahul' end from dual;
```

#### Answer:

The null=null is always false.so the Answer of this query is Rahul.

### 12.What is Index?What is use of index in SQL?

#### Answer:

Index is optional structure associated with the table which may or may not improve the performance of Query.In simple words suppose we want to search the topic in to book we go to index page of that book and search the topic which we want.Just like that to search the values from the table when indexing is there you need not use the full table scan.

Indexes are used to improve the performance of the query.

### 13.What is the query to find the students who get the marks in between 30-50?

Tell different ways.

#### Answer:

Select \* from Student where marks between 30 and 50;

Select \* from Student where marks>=30 and marks<=50;

### 14.What are advantages of Indexes?(90 % asked in Interview Questions for Accenture )

#### Answer:

Indexes are memory objects which are used to improve the performance of queries which allows faster retrieval of records.

Following are advantages of Indexes:

- 1.It allows faster retrieval of data
- 2.It avoids the Full table scan so that the performance of retrieving data from the table is faster.
- 3.It avoids the table access altogether
- 4.Indexes always speeds up the select statement.
- 5.Indexes used to improve the Execution plan of the database

### 15.What is parser?

#### Answer:

When SQL Statement has been written and generated then first step is parsing of that SQL Statement.Parsing is nothing but checking the syntaxes of SQL query.All the syntax of Query is correct or not is checked by SQL Parser.

There are 2 functions of parser:

- 1.Syntax analysis
- 2.Semantic analysis

### **16.What are Explicit Indexes?**

**Answer:**

The indexes which is created by user are called as explicit indexes.You can say the indexes which are created by 'Create Index' statement are called as Explicit indexes.

**Syntax:**

```
create index indexname on tablename(columnname);
```

**Example:**

```
Create index IND_Employee_ID on Employee(Employee_ID);
```

### **17.What is Bit-map index?Explain with Example.(90 % asked in Interview Questions for Accenture )**

**Answer:**

1.If Table contains the distinct values which are not more than 20 distinct values then user should go for Bit map indexes.

2.User should avoid the indexing on each and every row and do the indexing only on distinct records of the table column.You should able to check drastic change in query cost after changing the normal index to Bit map index.

3.The bit map indexes are very much useful in dataware housing where there are low level of concurrent transactions.Bit map index stores row\_id as associated key value with bitmap and did the indexing only distinct values.

4.Means If in 1 million records only 20 distinct values are there so Bitmap index only stores 20 values as bitmap and fetches the records from that 20 values only.

**Syntax:**

```
Create bitmap index Index_name on Table_name(Columns which have distinct values);
```

**Example:**

```
CREATE BITMAP index BM_DEPT_NAME on DEPT(Department_name);
```

### **18.What is not null constraint?**

**Answer:**

By default, a table column can hold NULL values. The NOT NULL constraint enforces a column to NOT accept NULL values.

The NOT NULL constraint enforces a field to always contain a value. This means that you cannot insert a new record, or update a record without adding a value to this field.

**19.What is first and last function in SQL?**

**Answer:**

The FIRST and LAST functions can be used to return the first or last value from an ordered sequence. Say we want to display the salary of each Employee, along with the lowest and highest within their department we may use something like.

Example:

```
SELECT EmpNo, DeptNo, Sal ,MIN (Sal) KEEP (DENSE_RANK FIRST ORDER BY Sal)
OVER (PARTITION BY DeptNo)"Lowest", MAX (Sal) KEEP (DENSE_RANK LAST
ORDER BY Sal) OVER (PARTITION BY DeptNo) "Highest"FROM EMPLOYEE
ORDER BY DeptNo, Sal;
```

**20.What is visible/invisible property of index?**

**Answer:**

User can make the indexes visible and invisible by altering the indexes.Following statement is used to make indexes visible and invisible.

```
ALTER INDEX index_name VISIBLE;
```

**21.How to Find table name and its owner?**

**Answer:**

To Find table name and its owner following query is used:

```
Select table_name,Owner from All_tables order by table_name,owner;
```

[CLICK HERE TO GET MORE INFORMATION ABOUT ORACLE SYSTEM TABLES](#)

**22.What is mean by implicit index.Explain with example.**

**Answer:**

Whenever we define unique key or primary key constraints on the table the index will automatically create on the table.These indexes are known as implicit indexes because these are created implicitly whenever the constraint has been applied to the table.These indexes are

normal indexes not unique indexes. The indexes are normal because the columns already have defined as unique so uniqueness is already been applied.

*Example:*

*Create table Employee*

*(Employee\_ID varchar2(20) primary key,*

*Employee name varchar2(50),*

*salary number(10,0) not null);*

If We check description of table:

***Desc Employee;***

<i>Name</i>	<i>Null</i>	<i>Type</i>
-----		
<i>Employee_ID</i>	<i>not null</i>	<i>varchar2</i>
<i>Employee_name</i>		<i>varchar2</i>
<i>Salary</i>	<i>not null</i>	<i>number</i>

Here you will see index is already created for Employee\_ID as it has defined primary key.

These are some important interview questions for accenture. Hope you like it. If you like it dont forget to share it.