**SQL Tech check**

1. **Define a Primary Key**

The primary key of a relational table uniquely identifies each record in the table. It can either be a normal attribute that is guaranteed to be unique (such as Social Security Number in a table with no more than one record per person) or it can be generated by the DBMS (such as a globally unique identifier, or GUID, in Microsoft SQL Server). Primary keys may consist of a single attribute or multiple attributes in combination.

1. **what are popular database vendors**

IBM Microsoft Oracle

1. **diff in unique and primary**

A unique key has the property that no two values in the range are the same. A primary key is chosen from among all the unique keys of a table to participate in referential integrity constraints. Primary keys are unique by definition.

1. **How many primary keys in a table?**

One primary key, but compound primary keys can have multiple values

1. **What is Left Outer Join?**

The Left Outer Join logical operator returns each row that satisfies the join of the first (top) input with the second (bottom) input. It also returns any rows from the first input that had no matching rows in the second input. The non matching rows in the second input are returned as null values. If no join predicate exists in the **Argument** column, each row is a matching row.

1. **DCL**
2. DCL is abbreviation of Data Control Language. It is used to create roles, permissions, and referential integrity as well it is used to control access to database by securing it.  
   Examples: GRANT, REVOKE statements
3. **DDL**
4. DDL is abbreviation of Data Definition Language. It is used to create and modify the structure of database objects in database.  
     
   Examples: CREATE, ALTER, DROP statements
5. **DML**

"DML is abbreviation of Data Manipulation Language. It is used to retrieve, store, modify, delete, insert and update data in database.

Examples: SELECT, UPDATE, INSERT statements"

1. **Difference between INSERT and UPDATE**

Insert' will add a new record or a new row into the database table.   
'Update' will modify a record in the DB table.

1. **Different types of constraints**
2. Constraint is a rule which cannot be violated by end users. Different types of constraints are available. They are:-  
   1) default constraint:-which is used to define a default   
   value.  
   2) primary key:-the key which does not allows duplication  
   and null values.  
   3) foreign key:-the key used to refer primary key defined   
   field in another table and it allows duplication.  
   4) null:-which allows NULL values.  
   5) Not null:-which does not allows NULL values.  
   6) unique key:-which does not allows duplication but allows   
   NULL values.
3. **What is cardinality; example**
4. The term data cardinality is used to mean the uniqueness of the data values which are contained in a particular column, known as attribute, of a database table.  
     
   There are actually three types of data cardinality each dealing with columnar value sets. These types are high-cardinality, normal-cardinality, and low-cardinality.
5. **What is a cross join?**

Cross join is used to return all records where each row from first table is combined with each row in second table.  
Cross join is also called as Cartesian Product join.

1. **Have u worked on any set operators?**

In table the data is represented as relation the operators used to operate this type of data is known as set operator. Some set operators are: 1.union () 2.intersection () 3.minus () etc…

1. **List some aggregate functions**

SUM Returns the total of the expression  
MIN Returns the minimal value of the expression  
MAX Returns the maximal value if the expression  
AVG Returns the average value of the expression  
COUNT Returns the number of lines in the data span

1. **what is index**

A database index is a data structure that improves the speed of data retrieval operations on a database table at the cost of slower writes and increased storage space. Indexes can be created using one or more columns of a database table…

1. **Where and Having**

The HAVING clause is used in combination with the GROUP BY clause. It can be used in a SELECT statement to filter the records that a GROUP BY returns. A WHERE clause is an optional part of a SelectExpression, DELETE statement, or UPDATE statement. The WHERE clause lets you select rows based on a boolean expression. Only rows for which the expression evaluates to TRUE are returned in the result, or, in the case of a DELETE statement, deleted, or, in the case of an UPDATE statement, updated

1. **how do you identify duplicates in a table**

SELECT tablefield, COUNT(tablefield) AS dup\_count   
FROM table  
GROUP BY tablefield  
HAVING (COUNT(tablefield) > 1)

1. **display only the first 3 letter of monthname**

substr(name,1,3)

1. **what is a SUBSTRING**

**substr** functions allows you to extract a substring from a string.

1. **what is CHAR and VARCHAR**

char limited - varchar variable

1. **what is a VIEW**

A SQL View is a virtual table, which is based on SQL SELECT query. Essentially a view is very close to a real database table (it has columns and rows just like a regular table), except for the fact that the real tables store data, while the views don’t. The view’s data is generated dynamically when the view is referenced.

1. **diff between view and materialized views**

A view uses a query to pull data from the underlying tables.

A materialized view is a table on disk that contains the result set of a query.

Materialized views are primarily used to increase application performance when it isn't feasible or desirable to use a standard view with indexes applied to it. The downside is that you have to use triggers, or some other automatic method, to keep the materialized view up to date.

1. **what is CAST**

CAST converts one built-in data type or collection-typed value into another built-in data type or collection-typed value.

1. **What function do you when you need to replace null values with a string?**
2. **CASE WHEN THEN with 'OR'**

The CASE keyword is used as a conditional operator that considers a value, examines it, and acts on an option depending on the value. The formula of the CASE statement is:  
  
CASE Expression  
 WHEN Value1 THEN Result  
 WHEN Value2 THEN Result  
  
 WHEN Value\_n THEN Result  
END

1. **Equivalent of case when in Oracle?**

If then else

1. **In a left outer join what would be the cardinality on left table?**
2. **Cardinality between dimension and FACT**
3. **Cardinality between Table A and Table B primary keys1..1**
4. **Fetch unique employees from each table**
5. **what is foreign key**

A foreign key is a field (or fields) that points to the primary key of another table. The purpose of the foreign key is to ensure referential integrity of the data. In other words, only values that are supposed to appear in the database are permitted.

1. **diff type of joins u know of**
2. **diff in insert and alter**
3. **delete & truncate - which is faster and why**

Truncate is faster as it does not keep it in memory to retrieve it later

1. **constraints**

You can place constraints to limit the type of data that can go into a table. Such constraints can be specified when the table when the table is first created via the CREATE TABLE statement, or after the table is already created via the ALTER TABLE statement.

1. **what is equi join**

It is a simple sql join condition which uses the equal sign as the comparison operator. Two types of equi joins are SQL Outer join and SQL Inner join.

1. **inner join**

SQL INNER JOIN returns records from both tables where a match is found based on join conditions (join\_condition). SQL INNER JOIN gets all the records from the table\_A and finds the matching records in the table\_B according to the join condition. The join condition determines whether both records are matched or not. If there is no match found, no records are returned.

1. **Running Total**
2. **Normalization**

Normalization usually involves dividing large tables into smaller (and less redundant) tables and defining relationships between them. The objective is to isolate data so that additions, deletions, and modifications of a field can be made in just one table and then propagated through the rest of the database via the defined relationships.

1. **List only managers with 3 or more employees on their team**
2. **Extract month order date field**

Extract (month from orderdate)

1. **ALTER**
2. **What do you use to add a new record into**

The INSERT INTO statement is used to add new records to a database table.

1. **Having query**

The HAVING clause is used in combination with the GROUP BY clause. It can be used in a SELECT statement to filter the records that a GROUP BY returns.

1. **where query**

The WHERE clause is used to extract only those records that fulfill a specified criterion.

1. **Subqeury**

The SQL sub query is a separate SELECT statement that is embedded in the main statement and may be executed once for each row of the result set.

1. **list city that generated 1M in revenue**
2. **What is a Data warehouse**

A data warehouse is a place where data is stored for archival, analysis and security purposes. Usually a data warehouse is either a single computer or many computers (servers) tied together to create one giant computer system.

1. **Slowly Changing Dimensions?**

The "Slowly Changing Dimension" problem is a common one particular to data warehousing. In a nutshell, this applies to cases where the attribute for a record varies over time.

Types of SCD:

1. **Conformed dimensions?**

A conformed dimension is a set of data attributes that have been physically implemented in multiple database tables using the same structure, attributes, domain values, definitions and concepts in each implementation.

1. **What is a Star Schema**

A star schema consists of fact tables and dimension tables. Fact tables contain the quantitative or factual data about a business--the information being queried. This information is often numerical, additive measurements and can consist of many columns and millions or billions of rows. Dimension tables are usually smaller and hold descriptive data that reflects the dimensions, or attributes, of a business. SQL queries then use joins between fact and dimension tables and constraints on the data to return selected information.

1. **Star versus Snowflake**

|  | **Star Schema** | **Snowflake Schema** |  |
| --- | --- | --- | --- |
| **DimTable Normalization:** | 2 Normal Denormalized Form | 3 Normal Form |  |
| **Joins:** | Fewer Joins | Higher number of Joins |  |
| **Ease of Use:** | Less complex queries and easy to understand | More complex queries and hence less easy to understand |  |
| **Query Performance:** | Less no. of foreign keys and hence lesser query execution time | More foreign keys-and hence more query execution time |  |
| **Ease of maintenance/change:** | Has redundant data and hence less easy to maintain/change | No redundancy and hence more easy to maintain and change |  |
| **Type of Datawarehouse:** | Good for large datawarehouses | Good to use for small datawarehouses/datamarts |  |
| **Dimension table:** | Contains only single dimension table for each dimension | It may have more than one dimension table for each dimension |  |

1. **Who is Kimball, Inmon?**
2. **Difference between delete, drop and truncate**

**DELETE**

The DELETE command is used to remove rows from a table. A WHERE clause can be used to only remove some rows. If no WHERE condition is specified, all rows will be removed. After performing a DELETE operation you need to COMMIT or ROLLBACK the transaction to make the change permanent or to undo it. Note that this operation will cause all DELETE triggers on the table to fire.

**TRUNCATE**

TRUNCATE removes all rows from a table. The operation cannot be rolled back and no triggers will be fired. As such, TRUCATE is faster and doesn't use as much undo space as a DELETE.

**DROP**

The DROP command removes a table from the database. All the tables' rows, indexes and privileges will also be removed. No DML triggers will be fired. The operation cannot be rolled back.

DROP and TRUNCATE are DDL commands, whereas DELETE is a DML command. Therefore DELETE operations can be rolled back (undone), while DROP and TRUNCATE operations cannot be rolled back.

***Difference between TRUNCATE and DELETE commands***

Submitted by Dipal havsar (not verified) on Tue, 2006-09-19 07:39.

1>TRUNCATE is a DDL command whereas DELETE is a DML command.

2>TRUNCATE is much faster than DELETE.

Reason:When you type DELETE.all the data get copied into the Rollback Tablespace first.then delete operation get performed.Thatswhy when you type ROLLBACK after deleting a table ,you can get back the data(The system get it for you from the Rollback Tablespace).All this process take time.But when you type TRUNCATE,it removes data directly without copying it into the Rollback Tablespace.Thatswhy TRUNCATE is faster.Once you Truncate you cann't get back the data.

3>You cann't rollback in TRUNCATE but in DELETE you can rollback.TRUNCATE removes the record permanently.

4>In case of TRUNCATE ,Trigger doesn't get fired.But in DML commands like DELETE .Trigger get fired.

5>You cann't use conditions(WHERE clause) in TRUNCATE.But in DELETE you can write conditions using WHERE clause

1. **Example of case statement and if then else**
2. **What should you be concerned when you are working with UNIONS**
3. **what is a typical cardinality on a left ourer join**
4. **If the cardinality was n..n what join would you give?**
5. **syntax for create and alter statements**

|  |  |  |  |
| --- | --- | --- | --- |
| CREATE TABLE suppliers | | | |
| ( | supplier\_id | number(10) | not null, |
|  | supplier\_name | varchar2(50) | not null, |
|  | contact\_name | varchar2(50) |  |
| ); |  |  |  |

ALTER TABLE supplier

ADD supplier\_name varchar2(50);

ALTER TABLE supplier

MODIFY ( supplier\_name varchar2(100) not null,

city varchar2(75) );

1. **what is the difference between functions & procedure**

Functions MUST return a value, procedures don't need to.

You can have DML (insert,update, delete) statements in a function. But, you cannot call such a function in a SQL query. For example, if you have a function that is updating a table, you cannot call that function from a SQL query.

- select myFunction(field) from sometable; will throw error.

Procedures VS Functions

Procedures are traditionally the workhorse of the coding world and functions are traditionally the smaller, more specific pieces of code. In general, if you need to update the chart of accounts, you would write a procedure. If you need to retrieve the organization code for a particular GL account, you would write a function.

Here are a few more differences between a procedure and a function:

■A function MUST return a value

■A procedure cannot return a value

■Procedures and functions can both return data in OUT and IN OUT parameters

■The return statement in a function returns control to the calling program and returns the results of the function

■The return statement of a procedure returns control to the calling program and cannot return a value

■Functions can be called from SQL, procedure cannot

■Functions are considered expressions, procedure are not

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

***To\_char***

In Oracle/PLSQL, the to\_char function converts a number or date to a string.

to\_char(sysdate, 'yyyy/mm/dd'); would return '2003/07/09'

to\_char(sysdate, 'Month DD, YYYY'); would return 'July 09, 2003'

to\_char(sysdate, 'FMMonth DD, YYYY'); would return 'July 9, 2003'

select ename, hiredate, to\_char((hiredate),'fmDay') "Day"

from emp

order by "Day";

***To\_date***

In Oracle/PLSQL, the to\_date function converts a string to a date

to\_date('2003/07/09', 'yyyy/mm/dd') would return a date value of July 9, 2003.

to\_date('070903', 'MMDDYY') would return a date value of July 9, 2003.

to\_date('20020315', 'yyyymmdd') would return a date value of Mar 15, 2002.

***To\_number***

to\_number function converts a string to a number.

to\_number('1210.73', '9999.99') would return the number 1210.73

to\_number('546', '999') would return the number 546

to\_number('23', '99') would return the number 23