DDL (DATA DEFINITION LANGUAGE) – create, alter, drop, truncate

DML (DATA MANIPULATION LANGUAGE) - insert , update, delete

**What is database ?**

**Database** is a Logical container for set of related objects such as databases, tables, procedures, functions, views etc. It is a base for data.

**WHAT IS DATA WAREHOUSE (DWH) ?**

Data warehouse refers to a central repository of data from multiple sources of information. Those data are consolidated, transformed and made available for the mining as well as online processing.

USUALLY DWH CONTAINS HISTORICAL DATA DERIVED FROM TRANSACTIONAL DATA (FROM DIFFERENT HETEROGENEOUS TRANSACTIONAL SYSTEMS)

**CHARACTERISTICS OF DWH:**

SUBJECT ORIENTED

INTEGRATED

TIME VARIANT

NON-VOLATILE

**SUBJECT ORIENTED:**

DATAWAREHOUSE IS COLLECTION OF DIFFEERENT INDEPENDENT SUBJECT AREAS.

EX: SALES, ACCOUNT, MARKETING, FINANCE (INSURANCE, LOANS)…

**INTEGRATED:**

A DWH IS AN INTEGRATED DATABASE WHICH CONTAINS BUSINESS INFO COLLECTED FROM VARIOUS OPERATIONAL DATA SOURCES.

**TIME VARIANT:**

A DWH IS A TIME VARIANT DATABASE WHICH WILL ALLOW US TO ANALYZE AND COMPARE THE BUSINESS W.R.T VARIOUS TIME PERIODS (YEARLY, HALF YEARLY, QUARTERLY, MONTHLY, WEEKLY…)

**NON-VOLATILE: (PERMANENT)**

WHICH MEANS ONCE THE DATA ENTERED INTO DWH THAT CANNOT BE CHANGED. IT DOESN’T REFLECT TO THE CHANGES TAKEN PLACE IN OPERATIONAL SYSTEM. HERE DATA IS STATIC.

**DATAMART:**

DATA MART IS A SMALLER VERSION OF DWH.

DATA MART DEALS WITH A SINGLE SUBJECT.

DATA MART IS SUBSET OF DWH.

**What is Agile ?**

Agile is a time boxed, iterative approach to software delivery that builds software incrementally from the start of the project, instead of trying to deliver it all at once near the end.

It works by breaking projects down into little bits of user functionality called [user stories](http://www.agilenutshell.com/user_stories), prioritizing them, and then continuously delivering them in 2 week to 4 weeks cycles called [iterations](http://www.agilenutshell.com/iterations).

**What is Scrum?**

Scrum is a software product development strategy that organizes software developers as a team to reach a common goal. It is a widely used subset of [agile software development.](https://stackify.com/tips-making-agile-less-fragile/)

In scrum we discussed about what I did yesterday, what I am going to do for today and what is my road block.

# **What is sprint (software development) ?**

In product development, a sprint is a set of period of time during which specific work has to be completed and made ready for review.

During the meeting, the product owner and the development team agrees upon exactly what work will be accomplished during the sprint. The development team has the final say how much work can be accomplished during the sprint, and the product owner has the final say on what criteria need to be met for the work to be approved and accepted. It is used to conduct for 2 to 4 weeks.

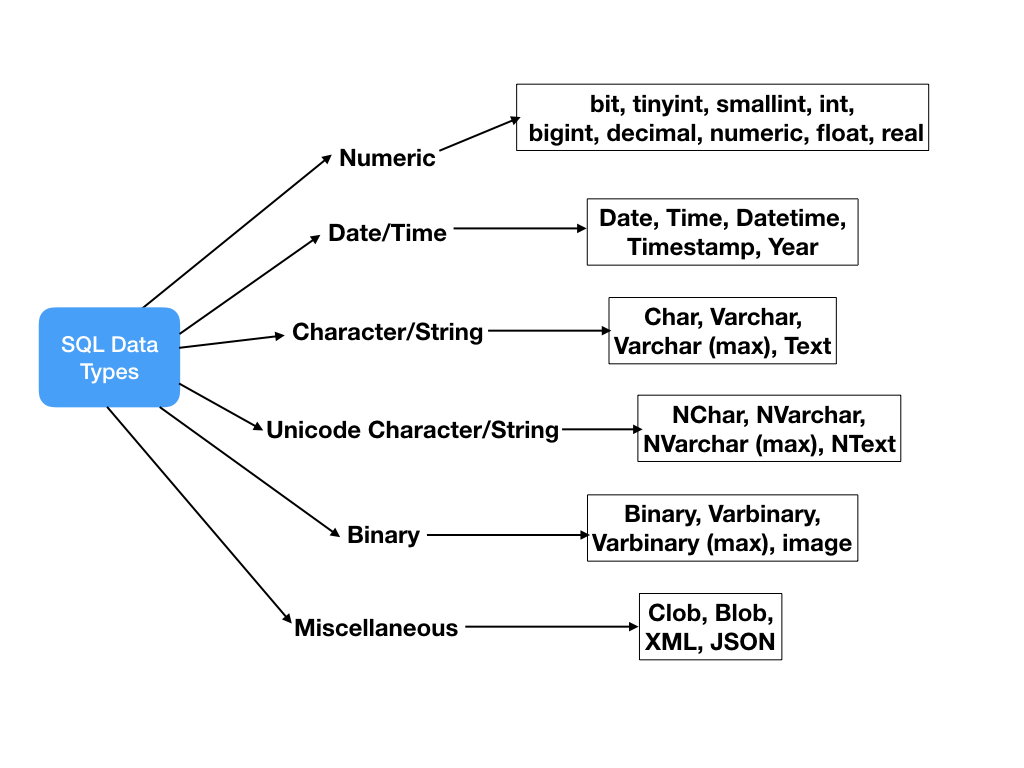
**What is sprint backlog ?**

The sprint backlog is a list of tasks identified by the Scrum team to be completed during the Scrum sprint (1 or 2 to 4 weeks). During the sprint planning meeting, the team selects some number of product backlog items, usually in the form of user stories, and identifies the tasks necessary to complete each user story. Most of the teams also estimate that how many hours for each task will take by someone on the team to complete.

**What is Product backlog ?**

**Product Backlog is an ordered list of everything that is to be needed in the product.** It isthe single source of requirements for any changes to be made to the product. The **Product Owner** is responsible for the Product Backlog, including its content, availability, and ordering.

**SQL Data Types :**



Difference between **delete** and **truncate**?

**Delete**

• We can use where clause to filter & delete specific records.

• Identity of column keep Delete retain the identity.

• Delete uses the more transaction space than Truncate statement.

• Delete can be used with indexed views.

• It is a DML command.

**Truncate:**

• We can’t use Where clause.

• It removes all rows with data from a table.

• Identify column is reset to its seed value if table contains any identity

column.

• To use Truncate on a table need at least alter permission on the table.

• Truncate cannot be used with indexed views.

• Truncate is a DDL command.

**Drop**

* + The drop command removes a table from the database.
* DROP operations cannot be rolled back. But Delete operations can be rolled

back (undone) by put **began transaction** in the first and **rollback** in the last

of query.

**Difference between temp table and CTE?**

**CTEs**

• Persist until the next query is run.

• Can’t be indexed.

• Can’t have constraints

• Are essentially disposable Views

• Can be recursive

**Temp Tables**

• Persist for the life of the current connection.

• Are real materialized tables exist in temp database.

• Can be indexed.

• Can have constraints.

• Can be referenced by other queries or sub procedures.

**Two kind of Temp Tables**

**1. #temp:** It works in the same page of sql file within the current connection.

**2. ##temp**: It works in the next page of sql file as well within the current connection.

**Q. What is CTE:**

A CTE is a temporary result set that can be referenced within a select, insert, delete or update statement immediately follows the CTE.

## Why to use a CTE :

In SQL, we will use sub-queries to join the records or filter the records from a sub-query. Whenever we refer the same data or join the same set of records using a sub-query, the code maintainability will be difficult. A CTE makes improved readability and maintenance easier.

Difference between **Primary Key** and **Unique Key**

**Primary key**

• It doesn’t allows the duplicate and null values in the column.

• There can be only one primary key in a table.

• Primary Key is a unique key identifier of the values and it keep the values uniquely in the table column.

**Unique Key**

• It can be more than one unique key in one table.

• It can be a candidate key.

• Unique key can have only one NULL value.

**Foreign Key**

* The table containing the foreign key is called the child table.
* It is a collection of fields in one table column that refers to the Primary Key

of another table column.

* The table containing the candidate key is called the referenced or parent

table.

* It is used to link or make a relation between two or multiple tables.

**Candidate Key:**

A candidate key is a column or a set of columns that can qualify as a primary key in the database. There can be multiple candidate keys in a database relation and each candidate key can work as a primary key for the table.

**Primary Key:**

The Primary Key constraint uniquely identifies each record in a table.

Primary keys must contain unique values, and can’t contain Null values.

A table can have only one primary key, which may consist of single or multiple fields.

## Foreign Key:

It is a key which is used to link (make a relation ) two tables.

It is a field (or collection of fields) in one table column that refers to the Primary Key in another table column.

The table containing the foreign key is called the child table, and the table containing the candidate key is called the referenced or parent table.

**Composite Key:**

A table can have only one Primary Key either on one column or multiple columns. When multiple columns are defined as Primary Key, then, it is called Composite Key.

[**What's the difference between a Primary and Identity Key?**](https://stackoverflow.com/questions/4293426/whats-the-difference-between-a-primary-key-and-identity)

**Identity Column:**

1. Identity column is auto incremented.

2 . Incremented numeric values only.

3 . 0nly one Identity column in a table.

4. All identity column is an primary Key.

5 . Values cannot be updated.

**Primary Key:**

1. Primary Key value will be entered by the user.

2 . Can be created more than one column (composite primary key).

3 . All primary key is not an identity column.

4. Can be update the value.

5. Can be refer by other table as a foreign key.

**SUB QUERIES:**

QUERY INSIDE THE QUERY.

GENERALLY A SELECT STATEMENT NESTED WITH ANOTHER SELECT STATEMENT IS CALLED SUBQUERY.

TWO TYPES OF SUBQUERIES:

1. NESTED SUBQUERY

2. CORRELATED SUBQUERY

**1. NESTED SUB-QUERY:**

IN A SUBQUERY, OUTER QUERY DEPENDS ON RESULT OF INNER QUERY IS CALLED NESTED SUB-QUERY

**2. CORRELATED SUBQUERY.**

IN A SUBQUERY , IF INNER QUERY DEPENDS ON OUTER QUERY THEN IT IS CALLED AS CORRELATED SUBQUERY.

**JOINS:**

JOINS ARE USED TO RETRIEVE DATA FROM MULTIPLE TABLE COLUMNS. IN SQL SERVER, WE HAVE BASICALLY 3 TYPES OF JOINS:

1. CROSS JOIN
2. INNER JOIN
3. OUTER JOIN

**1. CROSS JOIN:**

A JOIN WITHOUT CONDITION IS CALLED AS CROSS JOIN.

CROSS JOIN PRODUCES RESULT SET WHERE EVERY ROW IN FIRST TABLE JOIN WITH EVERY ROW IN SECOND TABLE.

ASSUME, 1ST TABLE HAS 5 RECORDS

2ND TABLE HAS 3 RECORDS

CROSS JOIN = 5\*3=15 RECORDS.

**2. INNER JOIN:**

**RULES OF JOINS:**

1. TABLES SHOULD HAVE RELATED COMMON COLUMN .
2. COMMON COLUMN DATA TYPES SHOULD BE SAME.
3. JOINING TABLES PRODUCES TEMPORARY RESULT SET WITH COMBINED STRUCTURE OF BOTH TABLES.

INNER JOIN IS DEFAULT TYPE OF JOIN. INNER JOIN PRODUES RESULT SET WHICH CONTAINS ONLY MATCHED RECORDS.

WE CAN GET ONLY MATCHED DATA ACCORDING TO THE JOIN CONDITION FROM THE TABLES WHICH INVOLVED IN INNER JOIN.

**3. OUTER JOIN:**

OUTER JOIN PRODUCES RESULT SET WHICH CONTAIN MATCHED ROWS AND UNMATCHED ROWS.

**WE HAVE 3 TYPES OF OUTER JOINS:**

1. LEFT OUTER JOIN
2. RIGHT OUTER JOIN
3. FULL OUTER JOIN

**A. LEFT OUTER JOIN**

TOTAL DATA FROM LEFT TABLE + MATCHED DATA FROM RIGHT TABLE

+ FOR NOT MATCHING IN RIGHT TABLE – FILLED WITH NULLS.

**B. RIGHT OUTER JOIN**

TOTAL DATA FROM RIGHT TABLE + MATCHED DATA FROM LEFT TABLE

+ FOR NOT MATCHING IN LEFT TABLE – FILLED WITH NULLS.

**C. FULL OUTER JOIN**

TOTAL DATA FROM RIGHT TABLE +TOTAL DATA FROM LEFT TABLE + MATCHING DATA + UNMATCHED DATA IS FILLED WITH NULLS.

**Set Operators:**

**1.Union**

Combining the tables data without any duplicate values.

**2. Union All**

Combining the tables data with duplicate values.

**3.Intersect**

Combining the common data from both tables.

**4.Except**

To get the certain data except data from the bottom table.

Set operators used to combine the data of two or more quires.

**Rule for set operators:**

Rule 1

Queries should have same no of columns and same corresponding data types.

Rule 2

Column heading in the result are based on the very first query used in operation.

Rule 3

Order by clause can be used in the last query only.

**Diff between Stored Procedures and User Defined Functions**

**Stored Procedures**

It can perform with input and output parameters.

Returns can return in numeric values.

It creates for reusing the query.

It contains all ddl/dml statement to perform with tables.

We can use transaction statements.

**User Defined Functions**

It can have only input parameter.

Returns can return scaler or set of rows.

we can’t use transaction statements.

It contains only select statement to perform with tables.

A state of the table can’t be changed.

We can read only data in the table.

**System define function:**

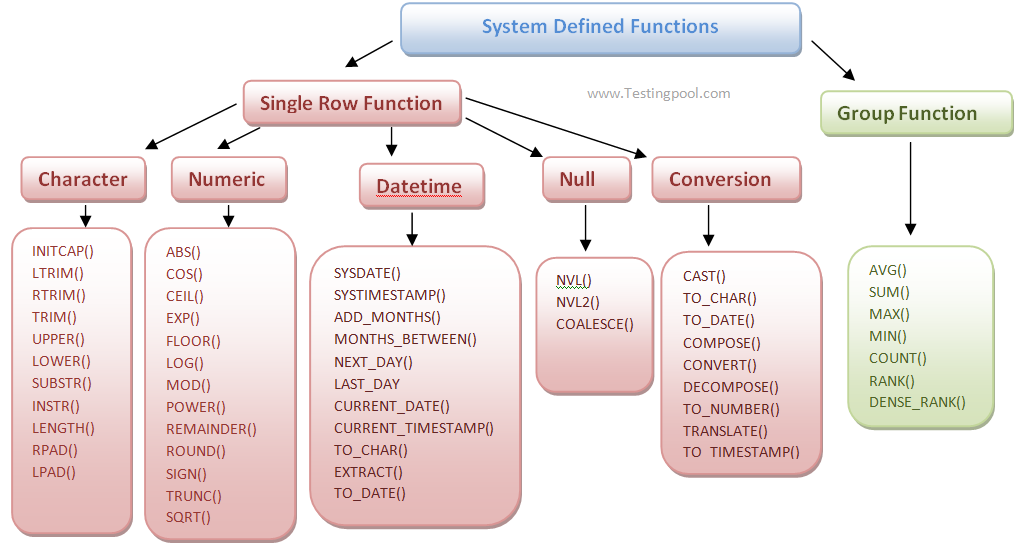
It’s also called global variables. System Defined functions are provided by SQL engine.

Basically it is used to convert data from one format to another business format.

There are two kinds of SDF:

1. Single-Row Functions

2. Group Functions



**EX:**

SELECT@@ERROR **-- IT STORES ERROR INFORMATION OF RECENT T-SQL**

**STATEMENT.**

SELECT@@VERSION **– IT STORES THE CURRENT SQL SERVER VERSION.**

SELECT@@SERVERNAME **– IT STORES THE SYSTEMNAME WHERE SQL SERVER IS INSTALLED.**

SELECT@@SERVICENAME **– THIS IS NOTHING BUT SQL SERVER SERVICE.**

**Computed column:**

Computed column is a virtual column in SQL server, which is not stored the data physically in the table. It’s data is not materialized in the database and gets calculated during query execution on that column.

**Identity column:**

An identity column is a [column](https://en.wikipedia.org/wiki/Column_(database)) in a [database](https://en.wikipedia.org/wiki/Database) [table](https://en.wikipedia.org/wiki/Table_(database)) that is made up of values generated by the database. This is much like an [Auto Number](https://en.wikipedia.org/wiki/AutoNumber) field it has own terminology.

## CHECK Constraint:

It is used to block the unnecessary data into the database/table column.

It is used to limit the value range that can be placed in a table column.

If it is used on a single column, it allows only certain values for that column.

**DEFAULT Constraint:**

It is used to provide a default value for a column.

Default value will be added to all new records if no other value is specified.

**Row\_number**

SELECT COL1, COL2, ROW\_NUMBER() OVER(PARTITION BY COL\_NAME ORDER BY COL\_NAME DESC) FROM TABLE\_NAME

**Rank**

SELECT COL1, COL2, Rank() OVER(PARTITION BY COL\_NAME ORDER BY COL\_NAME DESC) FROM TABLE\_NAME

**Advantages of Jobs in SQL**

1. Can run a process at a set time on a regular schedule.

2. It does work as automated processes.

3. SQL Jobs are very useful way of automating work.

4. It saves the time.

5. It does work within the short period of time.

6. There won’t be any problem even power is off in a local machine.

**What is Index ?**

An index is used to speed up the performance of queries. It makes faster retrieval of data from the table. The index can be created on one column or a group of columns in a table.

**What are all the different types of indexes?**  
There are three types of indexes  
**1. Unique Index:** Unique Indexes helps maintain data integrity by ensuring that no two rows of data in a table have identical key values. A unique index can be applied automatically when a primary key is defined. It ensures that the values in the index key columns are unique.  
**2. Clustered Index:** Clustered Index reorders the physical order of the table and search based on the key values. There will be only one clustered index per table.  
**3. Non-Clustered Index:** Non-Clustered Index doesn’t alter the physical order of the table and maintains a logical order of the data. Each table can have many non-clustered indexes.

**Advantages and disadvantages of Indexes**

**Advantages**

* Speed up SELECT query.
* Helps to make a row unique or without duplicates (primary, unique)
* If index is set to fill-text index, then we can search against large string values. for example to find a word from a sentence etc.

**Disadvantages**

* Indexes take additional disk space.
* Indexes slow down INSERT,UPDATE and DELETE, but will speed up UPDATE if the WHERE condition has an indexed field.  INSERT, UPDATE and DELETE becomes slower because on each operation the indexes must also be updated.

**What is a shared folder?**

A shared folder is a folder that is used for virtualization software to access files on the local machine from the remote machine.

# **SQL Logical Operators:**

SQL logical operators used to test for the truth of a condition.

A logical operator allows to test for the truth of a condition. Similar to a [comparison operator](http://www.sqltutorial.org/sql-comparison-operators/), a logical operator returns a value of true, false, or unknown.

**The following table illustrates the SQL logical operators:**

| **Operator** | **Meaning** |
| --- | --- |
| [ALL](http://www.sqltutorial.org/sql-all/) | Return true if all comparisons are true |
| [AND](http://www.sqltutorial.org/sql-and/) | Return true if both expressions are true |
| [ANY](http://www.sqltutorial.org/sql-any/) | Return true if any one of the comparisons is true. |
| [BETWEEN](http://www.sqltutorial.org/sql-between/) | Return true if the operand is within a range |
| [EXISTS](http://www.sqltutorial.org/sql-exists/) | Return true if a subquery contains any rows |
| [IN](http://www.sqltutorial.org/sql-in/) | Return true if the operand is equal to one of the value in a list |
| [LIKE](http://www.sqltutorial.org/sql-like/) | Return true if the operand matches a pattern |
| [NOT](http://www.sqltutorial.org/sql-not/) | Reverse the result of any other Boolean operator. |
| [OR](http://www.sqltutorial.org/sql-or/) | Return true if either expression is true |
| [SOME](http://www.sqltutorial.org/sql-any/) | Return true if some of the expressions are true |

## Arithmetic Operators:

These operators, just as the name implies, are used for arithmetic computations. The use of the arithmetic operators is very intuitive and they can be used in virtually every clause of the SQL statement.

For ex: +, - , \*, /, %, ||

­­­

**Comparison operator:**

A comparison (or relational) operator is a mathematical symbol which is used to compare two values.

Comparison operators are used in conditions that compares one expression with another. The result of a comparison can be TRUE, FALSE, or UNKNOWN (an operator that has one or two NULL expressions returns UNKNOWN).

The following are comparison operators:

ex : = (is equal to), > ( Greater than), < ( Less than.), >= (Greater than or equal to.), <= (Less than or equal to.), <> (Not equal to.)

**Q. What is view ? Can we insert, delete, update the data in view ?**

In SQL, a view is a virtual table based on the result-set of an SQL statement.

A view contains rows and columns, just like a real table.

We can add SQL functions, WHERE, and JOIN statements to a view and present the data as if the data were coming from one single table.

In a join query we can’t insert and delete the data from the view table.

A view can be updated with update both tables values.

**VIEWS:**

1. VIEWS ARE NOTHING BUT A LOGICAL REPRESENTATION BASED ON ONE OR MORE TABLES.
2. VIEW CONTAINS ONLY SELECT STATEMENT (IT STORES

SELECT QUERY).

1. VIEW IS ALSO CALLED AS VIRTUAL TABLE.
2. VIEW DOESN’T CONTAIN DATA IN IT.

## What is Normalization?

Normalization is a database design technique which organizes tables in a manner that reduces redundancy and dependency of data.

It divides larger tables to smaller tables and links them using relationships.

**1NF (First Normal Form) Rules**

* Each table cell should contain a single value.
* Each record needs to be unique.

**2NF (Second Normal Form) Rules**

* Rule 1- Be in 1NF
* Rule 2- Single Column Primary Key:

NO PARTIAL DEPENDENCY: EVERY NON KEY COLUMN SHOULD DEPEND ON PRIMARY KEY COLUMN.

It is clear that we can't move forward to make our simple database in 2nd Normalization form unless we partition the table

## 3NF (Third Normal Form) Rules

* Rule 1- Be in 2NF
* Rule 2- Has no transitive functional dependencies

(NO TRANSITIVE DEPENDENCY: A NON KEY COLUMN SHOULD NOT DEPENDS ON OTHER NON-KEY COLUMN.)

To move our 2NF table into 3NF, we again need to again divide our table.

### **What is Denormalization?**

Denormalization is a strategy used on a previously-normalized database to increase performance. The idea behind it is to add redundant data where we think it will help us the most. We can use extra attributes in an existing table, add new tables, or even create instances of existing tables. The usual goal is to decrease the running time of select queries by making data more accessible to the queries or by generating summarized reports in separate tables. This process can bring some new problems, and we’ll discuss them later.

A normalized database is the starting point for the denormalization process. It’s important to differentiate from the database that has not been normalized and the database that was normalized first and then denormalized later. The second one is okay; the first is often the result of bad database design or a lack of knowledge.

# **PIVOT** rotates a table-valued expression by turning the unique values from one column in the expression into multiple columns in the output, and runs aggregations where they're required on any left over column values that are wanted in the final output.

# **UNPIVOT**carries out the opposite operation to PIVOT by rotating columns of a table-valued expression into column values.

**Trigger** : A trigger is a special type of stored procedure that is fired when a specific type of data modification is made against a specific table or column.

**Data mapping document:**

The data mapping document will be an integral piece to the success of your data migration or integration. It’s with the data mapping document that you are able to mitigate your risk factors.

The data mapping process will determine what data needs to be transferred over to the new system. This includes:

* Which entities (a.k.a. tables) will be moved over along with the specific fields.
* The format of the data after it is transferred.
* The frequency of the transfer (for integrations).
* What will trigger the transfer (manual or automatic).

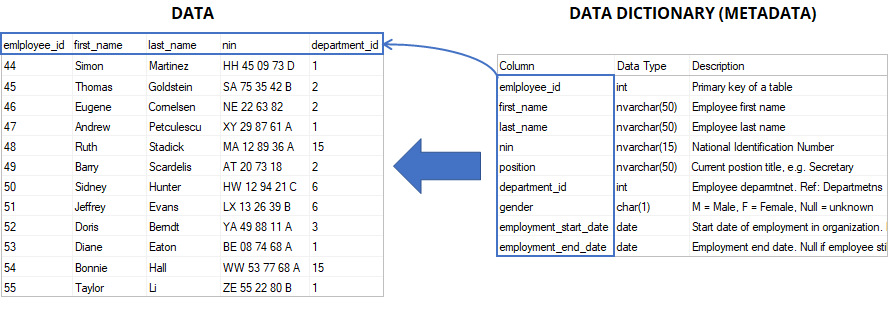
In order to figure out how the data needs to be formatted, or mapped, it is essential to build a data mapping document.  The data mapping document must include specifically the source and target data mappings. It must also include the primary key of all tables in source system.

### **Table of Contents:**

* [Elements of Data Dictionary](https://dataedo.com/kb/data-glossary/what-is-data-dictionary#toc_0)
* [Functions of Data Dictionary](https://dataedo.com/kb/data-glossary/what-is-data-dictionary#toc_1)
* [Forms of Data Dictionary](https://dataedo.com/kb/data-glossary/what-is-data-dictionary#toc_2)
* [Data Dictionary in Database Lifecycle](https://dataedo.com/kb/data-glossary/what-is-data-dictionary#toc_3)

**Data dictionary** is an inventory of data elements in a database or data model with detailed description of its format, relationships, meaning, source and usage. This term is a little ambiguous and can mean slightly different things:

1. A document
2. [Metadata tables in database systems (DBMS)](https://dataedo.com/kb/databases/all/data-dictionary)
3. Independent (in relation to DBMS) metadata repository



### Export Data Dictionary from your databases and share in HTML

**CHANGE COLUMN NAME USING SQL QUERY :**

TABLE NAME : 'CUSTOMERS, OLD COLUMN NAME : IID, TO BE CHANGE: CID FOR THIS QUERY WILL AS BELOW: (EXEC SP\_RENAME ‘TABLENAME.OLD COL NAME’,’NEW COL NAME’, ‘COLUMN’)

EXEC sp\_rename 'CUSTOMERS.IID', 'CID', 'COLUMN';

## What are the benefits of User-Defined Functions?

a. Can be used in a number of places without restrictions as compared to stored

procedures.   
b. Code can be made less complex and easier to write.  
c. Parameters can be passed to the function.  
d. They can be used to create joins and also be sued in a select, where or case

statement.  
e. Simpler to invoke.

**What is the advantage of stored procedure**

* **Maintainability**

Because scripts are in one location, updates and tracking of dependencies based on schema changes becomes easier

* **Testing**

Can be tested independent of the application

* **Isolation of Business Rules**

Having Stored Procedures in one location means that there’s no confusion of having business rules spread over potentially disparate code files in the application

* **Speed / Optimization**

Stored procedures are cached on the server

Execution plans for the process are easily reviewable without having to run the application

* **Utilization of Set-based Processing**

The power of SQL is its ability to quickly and efficiently perform set-based processing on large amounts of data; the coding equivalent is usually iterative looping, which is generally much slower

* **Security**

Limit direct access to tables via defined roles in the database

Provide an “interface” to the underlying data structure so that all implementation and even the data itself is shielded.

Securing just the data and the code that accesses it is easier than applying that security within the application code itself

# **What is SSIS ? (**[**SQL Server Integration Service**](https://www.tutorialgateway.org/ssis/)**)**

The SSIS is a business intelligence tool that provides data transformation solutions for various organizations.

SQL Server Integration Services is an ETL (Extract, Transform and Load) tool. It means SSIS can be used to extract data from a wide variety of sources such as Excel Files, Flat Files, XML Files, Relational databases, and transform (slice and dice) them as per requirements and finally load the data into the destination.

### **What is SSIS Work environment:**

In SSIS we can divideSSIS window into different parts as SSIS Work environment following:

1. **Solution Explorer:** This is a combination of project level connection managers, actual packages, and project parameters.
2. **Properties:** Use this window to change the properties of each and every task.
3. **SSIS** **Toolbox:** SSIS Toolbox provides a lot of built-in tasks, containers, transformations, sources, destinations, and administrative tasks to solve complex business problems. We can use these graphical SSIS tools by drag and drop those tasks in the work environment. It means we do not have to write a single line of code to perform most of the operations.
4. **Information:** We can see the information of each and every item in a toolbox by clicking the toolbox items.
5. **Connection Managers:** This window is to create a package level connection managers.
6. **Package:** It is the place where we design the complete data flow and perform all the transformations.

#### **Deployment**

After the completion of the package development, we can deploy the package into the production environment.

**Q. What is fuzzy lookup transformation in SSIS?**

The Fuzzy Lookup Transformation in [SSIS](https://www.tutorialgateway.org/ssis/) is used to replace the wrongly typed words with correct words. It uses fuzzy matching to find one or more close matches in the reference table and replace the source data with reference table data.

**Q. What is lookup transformation in SSIS ?**

The Lookup transformation in [SSIS](https://www.tutorialgateway.org/ssis/) is one of the most useful transformation which is used to compare the source data to the existing data from the reference table (or Lookup Table) and find matching ones. For the non-matching rows, we can further do some more work like as loading to the another table.

**Q. What is Foreach Loop Container ?**

The Foreach Loop container defines a repeating control flow in a package. It repeats the control flow for each member of a specified enumerator.

A Foreach Loop container can include multiple tasks and containers, but it can use only one type of enumerator. If the Foreach Loop container includes multiple tasks, we can map the enumerator collection value to multiple properties of each task.

**Difference between For Loop and Foreach Loop Container**

**For Loop Container** will execute the tasks a specified number of times, in other words 10 times, or 25 times, and the number of times is specified in the definition of the container. You can use a variable to specify what that count is.

**Foreach Loop Container** is used to execute once for each item in the collection of many items that it is looking at. A good example would be if users are putting an Excel file into a directory for import into the DB. We cannot tell ahead of time how many will be in the directory, because a user might be late, or there might be more than one file from a given user. When we define the Foreach container, we would tell it to execute for each \*.xls in the directory, and it will then loop through, importing each one individually, regardless of how many files are actually there.

**Transformations**

Transformations or transforms are key components to the Data Flow that transform the data to a desired format.

**Aggregate**

The Aggregate Transformation allows to aggregate data from the Data Flow to apply certain T-SQL functions that are done in a GROUP BY statement, such as Average, Minimum, Maximum, and Count.

**Conditional Split**

The Conditional Split Transformation is a fantastic way to add complex logic to Data Flow. This transformation enables to send the data from a single data path to various outputs or paths based on conditions that use the SSIS expression language.

**Data Conversion**

The Data Conversion Transformation performs a similar function to the CONVERT or CAST functions in T-SQL. This transformation is used to check each column that needed to convert and then specify to what we wish to convert it under the Data Type column.

**Derived Column**

The Derived Column Transformation creates a new column that is calculated (derived) from the output of an another column or set of columns.

**Lookup**

The Lookup Transformation performs what equates to an INNER JOIN on the Data Flow and a second data set. The second data set can be an OLE DB table or a cached file, which is loaded in the Cache Transformation. After perform the lookup, we can retrieve additional columns from the second column. If no match is found, an error occurs by default. We can later choose, using the Configure Error Output button, to ignore the failure.

**Cache**

The Cache Transformation enables to load a cache file on disk in the Data Flow. This cache file is later used for fast lookups in a Lookup Transformation. The Cache Transformation can be used to populate a cache file in the Data Flow as a transformation.

**Row Count**

The Row Count Transformation provides the capability to count rows in a stream that is directed to its input source. This transformation must place that count into a variable that could be used in the Control Flow.

**Sort**

The Sort Transformation is a fully blocking asynchronous transformation that enables to sort data based on any column in the path.

**Union All**

It  works as the same way as the Merge Transformation, but it does not require sorted data. It takes the outputs from multiple sources or transformations and combines them into a single result set.

**Merge**

The Merge Transformation can merge data from two paths into a single output. This transformation is useful when we wish to break out Data Flow into a path that handles certain errors and then merge it back into the main Data Flow downstream after the errors have been handled. It’s also useful if we wish to merge data from two Data Sources.

The data must be sorted for the Merge Transformation.

**Merge Join**

The Merge Join transformation provides an output that is generated by joining two sorted datasets using a FULL, LEFT, or INNER join.

This transformation will merge the output of two inputs and perform an INNER or OUTER join on the data.

**Multicast**

The Multicast Transformation, as the name implies, can send a single data input to multiple output paths easily. We can use this transformation to send a path to multiple destinations sliced in different ways.

**OLE DB Command**

The OLE DB Command Transformation is a component designed to execute a SQL statement for each row in an input stream.

**What is Business Intelligence (BI) ?**

The term **Business Intelligence (BI)** refers to technologies, applications and practices for the collection, integration, analysis, and presentation of business information.

**Business intelligence (BI)** is a technology-driven process for analyzing data and presenting actionable information to help executives, managers and other corporate [end users](https://whatis.techtarget.com/definition/end-user) make informed business decisions.

**How to validate the loaded file from the source to Raw Table in SQL ?**

There are lots of ways one can validate data. Much of it depends on three things:

1. How much time do you have for validation?
2. What are your processing capabilities?
3. Is the data on a QA or Production SQL server?

**If you are in QA and have lots of processing power, you can do basic checks (Validation):**

* Where there any warnings or errors during the data load?
* Count the total number of items in the database vs. the raw file
* Count the total number of null records in the database
* Check the total number of columns vs. the raw file
* Check the length of the variables. Are they as expected?
* Are any character columns unexpectedly truncated?
* Are numeric columns out to the correct number of significant digits?
* Are dates reasonable? For example, if you expected dates from 2004, do they say 1970?
* How many duplicates are there?
* Check if the data in the columns make sense. A few questions you can ask: are any rows "shifted?" Are numeric variables in numeric columns? Is the key column actually a key? Do the column names make sense? Your check of null records should help detect these things.
* Can you manually calculate any columns and compare your calculation to the one in the file?

If you are low on processing power or are on a production server and do not want to risk degrading performance for other users, you can do many of the above checks with a [simple random sample](https://stackoverflow.com/questions/848872/select-n-random-rows-from-sql-server-table). Take, say, 100,000 rows at a time.; or, stratify it if needed.

These are just a few checks you can do. The more comparisons and sanity checks, the better off you are.

**Most importantly, communicate these findings and anything that seems strange to the file owner.** They should be able to give you additional insight to the data load is correct, or if they even gave you the right file in the first place.

You're loading the data and providing as many reasonable checks as possible. If they're satisfied with the outcome, and you're satisfied with the outcome, you should consider the data valid.

# **Why are Views used in SQL?**

1. It helps in limiting user to access subset of data in table
2. It can summarize data from various tables to generate reports.
3. To restrict data access
4. To provide data independence.
5. To make complex queries easy.
6. To present different views of the same data.

# **What is SQL Server Analysis Services (SSAS)?**

Analysis Services is an analytical data engine used in decision support and business analytics, providing the analytical data for business reports and client applications such as Power BI, Excel, Reporting Services reports, and other data visualization tools.  
A typical workflow includes authoring a multidimensional or tabular data model, deploying the model as a database to an on-premises SQL Server Analysis Services or Azure Analysis Services server instance, setting up recurring data processing, and assigning permissions to allow data access by end-users. When it’s ready to go, your semantic data model can be accessed by any client application supporting Analysis Services as a data source.” — says us Microsoft.

In other words, SQL Server Analysis Services (SSAS) — is a Microsoft technology from BI stack, which helps us to develop Online Analytical Processing (OLAP) solutions. OLAP is part of the broader category of business intelligence, which also encompasses relational database, report writing and data mining. SSAS places predictive analytic capabilities in the hands of information workers by creating an instant connection to backend data using familiar applications such as Microsoft Excel and SharePoint for analysis.

**ER/Studio**

ER/Studio is data architecture and database design software developed by Embarcadero Technologies. ER/Studio is compatible with multiple database platforms and is used by data architects, data modelers, database administrators and business analysts to create and manage database designs, document and reuse data assets.

## Top Answers to SQL Interview Questions

**1. Compare SQL & PL/SQL**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **SQL** | **PL/SQL** |
| What it is | Single query or command execution | Full programming language |
| What it comprises | Data source for reports, web pages | Application language to build, format and display report, web pages |
| Characteristic | Declarative in nature | Procedural in nature |
| Used for | Manipulating data | Creating applications |

**2. What is BCP? When is it used?**

It is a tool used to duplicate enormous quantity of information from tables and views. It does not facsimile the structures same as foundation to target.  
**BULK INSERT** command helps to bring in a data folder into a record, table or view in a user-specific arrangement.

**3. When is the UPDATE\_STATISTICS command used?**

This command is used, ones the processing of large data is done.  
When we delete a large number of files, alteration or reproduction takes place in the tables, to be concerned of these changes we need to restructure the indexes This is done **UPDATE\_STATISTICS**.

**4. Explain the steps needed to Create the scheduled job?**

**Steps to create a Scheduled Job :**

1. Connect to the database of SQL server in SQL Server Management Studio. On the SQL Server Agent, we will find a Jobs folder.
2. Right click on jobs and choose Add New.
3. A New Job window will come into view. Give an associated name for the same.
4. Click next on the “Steps” in the left list of options. An SQL job can have multiple steps either in the form of SQL declaration or a stored practice call.
5. Click on the “Schedules” in the left list of options. An SQL job can comprise of one or supplementary schedules. It is basically the instance at which SQL job will jog itself. We can spell out returning schedules also.

**5. What are the different types of constraints? Explain primary key, foreign key, unique key & not null constraints?**

**6. When are we going to use truncate and delete?**

1. TRUNCATE is a DDL command, whereas DELETE is a DML command.
2. We can’t execute a trigger in case of TRUNCATE whilst with DELETE, we can accomplish a trigger.
3. TRUNCATE is quicker than DELETE, for the reason that when we use DELETE to delete the data, at that time it store the whole statistics in the rollback gap on or after where we can get the data back after removal. In case of TRUNCATE, it will not store data in rollback gap and will unswervingly rub it out. TRUNCATE do not recover the deleted data.
4. We can use any condition in WHERE clause using DELETE but it is not possible with TRUNCATE.5.If a table is referenced by any foreign key constraints, then TRUNCATE won’t work.

Go through [this SQL tutorial](https://intellipaat.com/tutorial/sql-tutorial/delete-query/) to learn more about SQL commands.

**7. Explain correlated query work?**

It’s most important to be attentive of the arrange of operations in an interrelated subquery.  
**First,** a row is processed in the outer doubt.  
**Then,** for that exacting row, the subquery is executed – as a result for each row processed by the outer query, the subquery will also be processed. In correlated subquery, each time a line is worked for Emp1, the subquery will also make a decision on the exacting row’s value for Emp1.Salary and run. And the outer query will move on to the next row, and the subquery will execute for that row’s value of Emp1.Salary.  
It will persist in anticipation of the **“WHERE (1) = (… )”** state is pleased.

**8. When is the Explicit Cursor Used ?**

If the developer needs to perform the row by row operations for the result set containing more than one row, then he unambiguously declares a pointer with a name. They are managed by OPEN, FETCH and CLOSE.%FOUND, %NOFOUND, %ROWCOUNT and %ISOPEN characteristics are used in all types of pointers.

**9. Find What is Wrong in this Query?  
SELECT subject\_code, AVG (marks) FROM students WHERE AVG(marks) > 75 GROUP BY subject\_code; The WHERE clause cannot be used to restrict groups. Instead, the HAVING clause should be used.**

SELECT subject\_code, AVG (marks)

FROM students

HAVING AVG(marks) > 75

GROUP BY subject\_code;

**10. Write the Syntax for STUFF function in an SQL server?**

STUFF (String1, Position, Length, String2)

String1 - String to be overwritten

Position - Starting location for overwriting

Length - Length of substitute string

String2- String to overwrite.

**11. Name some commands that can be used to manipulate text in T-SQL code. For example, a command that obtains only a portion of the text or replace a text string, etc.**

* **CHARINDEX( findTextData, textData, [startingPosition] )** – Returns the starting position of the specified expression in a character string. The starting position is optional.
* **LEFT( character\_expression , integer\_expression )** – Returns the left part of a character string with the specified number of characters.
* **LEN( textData )** – Returns integer value of the length of the string, excluding trailing blanks.
* **LOWER ( character\_expression )** – Returns a character expression after converting uppercase character data to lowercase.
* **LTRIM( textData)** – Removes leading blanks. PATINDEX( findTextData, textData ) – Returns integer value of the starting position of text found in the string.
* **REPLACE( textData, findTextData, replaceWithTextData )** – Replaces occurrences of text found in the string with a new value.
* **REPLICATE( character\_expression , integer\_expression )** – Repeats a character expression for a specified number of times.
* **REVERSE( character\_expression )** – Returns the reverse of a character expression.
* **RTRIM( textData)** – Removes trailing blanks. SPACE( numberOfSpaces ) – Repeats space value specified number of times.
* **STUFF( textData, start , length , insertTextData )** – Deletes a specified length of characters and inserts another set of characters at a specified starting point.
* **SUBSTRING( textData, startPosition, length )** – Returns portion of the string.
* **UPPER( character\_expression )** – Returns a character expression with lowercase character data converted to uppercase.

**12. What are the three ways that Dynamic SQL can be executed?**

* Writing a query with parameters.
* Using EXEC.
* Using sp\_executesql.

Get a clear understanding of SQL in [this riveting blog](https://intellipaat.com/blog/nosql-vs-sql-what-is-better/).

**13. In what version of SQL Server were synonyms released? How do synonyms work and explain its use cases? Synonyms were released with SQL Server 2005.**

* Synonyms enable the reference of another object (View, Table, Stored Procedure or Function) potentially on a different server, database or schema in your environment. In simple words, the original object that is referenced in the whole code is using a completely different underlying object, but no coding changes are necessary. Think of this as an alias as a means to simplify migrations and application testing without the need to make any dependent coding changes.
* Synonyms can offer a great deal of value when converting underlying database objects without breaking front end or middle tier code. This could be useful during a re-architecture or upgrade project.

Become Master of SQL by going through [this SQL training course](https://intellipaat.com/microsoft-sql-server-certification-training/#course-content).

**14. If you are a SQL Developer, how can you delete duplicate records in a table with no primary key?**

Use the SET ROWCOUNT command. For instance,  
if you have 2 duplicate rows, you would SET ROWCOUNT 1, execute DELETE command and then SET ROWCOUNT 0.

**15. Is it possible to import data directly from T-SQL commands without using SQL Server Integration Services? If so, what are the commands?**

**Yes, six commands are available to import data directly in the T-SQL language. These commands include :**

* **BCP :** The bulk copy (bcp) command of Microsoft SQL Server provides you with the ability to insert large numbers of records directly from the command line. In addition to being a great tool for command-line aficionados, bcp is a powerful tool for those seeking to insert data into a SQL Server database from within a batch file or other programmatic method.
* **Bulk Insert :** The BULK INSERT statement was introduced in SQL Server 7 and allows you to interact with bcp (bulk copy program) via a script.
* **Open Row Set :** The OPEN ROW SET function can be referenced in the FROM clause of a query as if it were a table name. The OPENROWSET function can also be referenced as the target table of an INSERT, UPDATE, or DELETE statement, subject to the capabilities of the OLE DB provider. Although the query might return multiple result sets, OPENROWSET returns only the first one.
* **OPEN DATA SOURCE :** Provides ad hoc connection information as part of a four-part object name without using a linked server name.
* **OPEN QUERY :** Executes the specified pass-through query on the specified linked server. This server is an OLE DB data source. OPENQUERY can be referenced in the FROM clause of a query as if it were a table name.
* **Linked Servers :** Configure a linked server to enable the SQL Server Database Engine to execute commands against OLE DB data sources outside of the instance of SQL Server. Typically linked servers are configured to enable the Database Engine to execute a Transact-SQL statement that includes tables in another instance of SQL Server, or another database product such as Oracle.

**16. What is the native system stored procedure to execute a command against all databases?**

* The sp\_MSforeachdb system stored procedure accepts the **@Command** parameter which can be exetecuted against all databases. The ‘?’ is used as a placeholder for the database name to execute the same command.
* The alternative is to use a cursor to process specific commands against each database.

**17. How can a SQL Developer prevent T-SQL code from running on a production SQL Server?**

Use IF logic with the **@@SERVERNAME function** compared against a string with a RETURN command before any other logic.

**18. How do you maintain database integrity where deletions from one table will automatically cause deletions in another table?**

You can create a trigger that will automatically delete elements in the second table when elements from the first table are removed.

**19. What port does SQL server run on?**

1433 is the standard port for SQL server.

Go through [this SQL Video](https://intellipaat.com/microsoft-sql-server-certification-training/#course-preview) to get clear understanding of SQL.

**20. What is the SQL CASE statement used for? Explain with an example?**

It allows you to embed an if-else like clause in the SELECT clause.

SELECT Employee\_Name, CASE Location

WHEN 'alex' THEN Bonus \* 2

WHEN 'robin' THEN Bonus \*, 5

ELSE Bonus

END

"New Bonus"

FROM Intellipaat\_employee;

**Read**[**this blog**](https://intellipaat.com/blog/sql-optimization-techniques/)**to learn why SQL Optimization has always been a important aspect of database management.**

**21. What are the risks of storing a hibernate-managed object in cache? How do you overcome the problems?**

The primary problem here is that the object will outlive the session it came from. Lazily loaded properties won’t get loaded if needed later. To overcome the problem, perform cache on the object’s id and class and then retrieve the object in the current session context.

**22. When is the use of UPDATE\_STATISTICS command ?**

Updating statistics ensures that queries compile with up-to-date statistics. However, updating statistics causes queries to recompile. We recommend not updating statistics too often because there is a performance tradeoff between improving query plans and the time it takes to recompile queries. The specific tradeoffs depend on your application. UPDATE STATISTICS can use tempdb to sort the sample of rows for building statistics.

**Syntax**

UPDATE STATISTICS table\_or\_indexed\_view\_name

[

{

{ index\_or\_statistics\_\_name }

| ( { index\_or\_statistics\_name } [ ,...n ] )

}

]

[ WITH

[

FULLSCAN

| SAMPLE number { PERCENT | ROWS }

| RESAMPLE

[ ON PARTITIONS ( { | } [, …n] ) ]

| [ ,...n ]

]

[ [ , ] [ ALL | COLUMNS | INDEX ]

[ [ , ] NORECOMPUTE ]

[ [ , ] INCREMENTAL = { ON | OFF } ]

] ;

::=

[ STATS\_STREAM = stats\_stream ]

[ ROWCOUNT = numeric\_constant ]

[ PAGECOUNT = numeric\_contant ]

**23. What is SQL Profiler?**

Microsoft SQL Server Profiler is a graphical user interface to SQL Trace for monitoring an instance of the Database Engine or Analysis Services. You can capture and save data about each event to a file or table to analyze later.

Use SQL Profiler to monitor only the events in which you are interested.

If traces are becoming too large, you can filter them based on the information you want, so that only a subset of the event data is collected. Monitoring too many events adds overhead to the server and the monitoring process and can cause the trace file or trace table to grow very large, especially when the monitoring process takes place over a long period of time.

**24. What command using Query Analyzer will give you the version of SQL server and operating system?**

SELECT SERVERPROPERTY (‘productversion’), SERVERPROPERTY (‘productlevel’), SERVERPROPERTY (‘edition’).

**25. What does it mean to have QUOTED\_IDENTIFIER ON? What are the implications of having it OFF?**

When **SET QUOTED\_IDENTIFIER** is **ON,** identifiers can be delimited by double quotation marks, and literals must be delimited by single quotation marks. When **SET QUOTED\_IDENTIFIER** is **OFF**, identifiers cannot be quoted and must follow all Transact-SQL rules for identifiers.

**26. What is the STUFF function and how does it differ from the REPLACE function in SQL?**

**Stuff function :** – This function is used to replace string from the given start position, passed as 2nd argument with string passed as last argument. In Stuff function, 3rd argument defines the number of characters which are going to be replaced.  
**Syntax :-**

STUFF ( character\_expression , start , length , replaceWith\_expression )

**For example :-**

Select Stuff ('Intellipaat', 3, 3, 'abc')

This query will return the string **"Iabcllipaat"**. In this example, Stuff function replaces the string **"Intellipaat"** onwards the 3rd position('nte') with 'abc'.

**Replace Function :**– Replace function is used to replace all occurrence of a specified with the string passed as last argument.  
**Syntax :-**

REPLACE ( string\_expression , string\_pattern , string\_replacement )

**For example :-**

Select Replace ('Abcabcabc', 'bc', 'xy')

This query will return the string **Axyaxyaxy**. In this example, Replace function replaces the occurrence of each **'bc'** string with **'xy'**.

[**Learn SQL from Experts! Enrol Today**](https://intellipaat.com/microsoft-sql-server-certification-training/?utm_source=IQ&utm_campaign=IQ_SQL_CTA&utm_medium=Website#course-content)

**27. How to get @@ERROR and @@ROWCOUNT at the same time?**

If @@Rowcount is checked after Error checking statement then it will have 0 as the value of @@Recordcount as it would have been reset. And if @@Recordcount is checked before the error-checking statement then @@Error would get reset. To get @@error and @@rowcount at the same time do both in same statement and store them in local variable.

SELECT @RC = @@ROWCOUNT, [@ER](https://intellipaat.com/members/er/) = @@ERROR

**28. What is de-normalization in SQL database administration? Give examples**

De-normalization is used to optimize the readability and performance of the database by adding redundant data. It covers the inefficiencies in the relational database software.  
De-normalization logical data design tend to improve the query responses by creating rules in the database which are called as constraints.  
**Examples include the following :**

* Materialized views for implementation purpose such as :
* Storing the count of “many” objects in one-to-many relationship.
* Linking attribute of one relation with other relations.
* To improve the performance and scalability of web applications.

**29. Can you explain about buffer cash and log Cache in SQL Server?**

* **Buffer Cache :** Buffer cache is a memory pool in which data pages are read. The ideal performance of the buffer cache is indicated as: 95% indicates that pages that were found in the memory are 95% of time. Another 5% is need physical disk access.  
  If the value falls below 90%, it is the indication of more physical memory requirement on the server.
* **Log Caches :** Log cache is a memory pool used to read and write the log pages. A set of cache pages are available in each log cache. The synchronization is reduced between log and data buffers by managing log cache separately from the buffer cache.

**30. Describe how to use Linked Server.**

MS SQL Server supports the connection to different OLE DB on an ad hoc basis. This persistent connection is referred as Linked Server.  
**Following are the steps to use Linked Server for any OLE DB. You can refer this to use an MS-Excel workbook.**

1. Open SQL Server Management Studio in SQL Server.
2. Expand Server Objects in Object Explorer.
3. Right-click on Linked Servers. Click on New Linked Server.
4. Select General page in the left pane and
   * Type any name for the linked server in the first text box.
   * Select the Other Data Source option.
   * Click on Microsoft Jet 4.0 OLE DB Provider from the Provider list.
   * Type the Excel as the name of the OLE DB data source.
   * Type the full path and file name of the Excel file in Data Source box.
   * Type the Excel version no. (7.0, 8.0 etc) in the Provider String. Use Excel 8.0 for Excel 2000, Excel 2002 or Excel 97.
   * To create a linked server click on OK.

**31. How to find second highest salary of an Employee?**

There are many ways to find second highest salary of Employees in SQ. You can either use SQL Join or Subquery to solve this problem.  
**Here is SQL query using Subquery :**

Select MAX(Salary) from Intellipaat\_emplyee WHERE Salary NOT IN ( select MAX(Salary) from Intellipaat\_employee.

**32. Explain how to send email from SQL database.**

SQL Server has a feature for sending mails. Stored procedures can also be used for sending mail on demand. With SQL Server 2005, MAPI client is not needed for sending mails.  
**The following is the process for sending emails from database.**

* Make sure that the SQL Server Mail account is configured correctly and enable Database Mail.
* Write a script to send an e-mail. The following is the script.
* USE [YourDB]
* EXEC msdb.dbo.sp\_send\_dbmail
* @recipients = 'xyz@intellipaat.com; abc@intellipaat.com;pqr@intellipaat.com’
* @body = ' A warm wish for your future endeavor',
* @subject = 'This mail was sent using Database Mail' ;

GO

**33. How to make remote connection in database?**

**The following is the process to make a remote connection in database :**

1. Use SQL Server Surface Area Configuration Tool for enabling the remote connection in database.
2. Click on Surface Area Configuration for Services and Connections.
3. Click on SQLEXPRESS/Database Engine/RemoteConnections.
4. Select the radio button: Local and Remote Connections and select ‘Using TCP/IP only’ under Local and Remote Connections.
5. Click on OK button / Apply button

**34. What is the purpose of OPENXML clause SQL server stored procedure?**

OPENXML parses the XML data in SQL Server in an efficient manner. It’s primary ability is to insert XML data to the RDB. It is also possible to query the data by using Open XML. The path of the XML element needs to be specified by using ‘xpath’.  
**The following is a procedure for retrieving xml data:**

DECLARE @index int

DECLARE @xmlString varchar(8000)

SET @xmlString ='

abc

9343463943/PhoneNo>

xyz

9342673212

'

EXEC sp\_xml\_preparedocument @index OUTPUT, @xmlString

SELECT \* FROM OPENXML (@index, 'Persons/Person') WITH (id varchar(10), Name varchar(100) 'Name' , PhoneNo varchar(50) 'PhoneNo')

EXEC sp\_xml\_removedocument @index

The above code snippet results the following:

15201 abc 9343463943

15202 xyz 9342673212

**35. How to store pdf file in SQL Server?**

Create a column as type ‘blob’ in a table. Read the content of the file and save in ‘blob’ type column in a table.  
**Or**  
Store them in a folder and establish the pointer to link them in the database.

**36. Explain the use of keyword WITH ENCRYPTION. Create a Store Procedure with Encryption.**

It is a way to convert the original text of the stored procedure into encrypted form. The stored procedure gets obfuscated and the output of this is not visible to

CREATE PROCEDURE Abc

WITH ENCRYPTION

AS

<< SELECT statement>>

GO

WITH ENCRYPTION indicates that SQL Server will convert the original text of CREATE PROCEDURE statement to an encrypted format. Users that do not have no access to system tables or database files cannot retrieve the encrypted text. However, the text will be available to privileged users.

**Example:**

**CREATE PROCEDURE salary\_sum  
WITH ENCRYTION  
AS  
SELECT sum(salary)  
FROM employee  
WHERE emp\_dept LIKE Develop**

**37. What is lock escalation?**

Lock escalation is used to convert row locks and page locks into table locks thereby “escalating” the smaller or finer locks. This increases the system performance as each lock is nothing but a memory structure. Too many locks would mean more consumption of memory. Hence, escalation is used.  
Lock escalation from SQL Server 7.0 onwards is dynamically managed by SQL Server. It is the process of converting a lot of low level locks into higher level locks.

**38. What is Failover clustering overview?**

Failover clustering is mainly used for data availability. **Typically, in a failover cluster, there are two machines.**

* One machine provides the basic services and the second is available to run the service when the primary system fails.
* The primary system is monitored periodically to check if it works. This monitoring may be performed by the failover computer or an independent system also called as cluster controller. In an event of failure of primary computer, the failover system takes control.

**39. What is Builtin/Administrator?**

The Builtin/Administrator account is basically used during some setup to join some machine in the domain. It should be disabled immediately thereafter. For any disaster recovery, the account will be automatically enabled. It should not be used for normal operations.

**40. What XML support does the SQL server extend?**

**SQL Server (server-side) supports 3 major elements :**

1. Creation of XML fragments: This is done from the relational data using FOR XML to the select query.
2. Ability to shred xml data to be stored in the database.
3. Finally, storing the xml data.

Client-side XML support in SQL Server is in the form of SQLXML. **It can be described in terms of :**

* **XML Views :** providing bidirectional mapping between XML schemas and relational tables.
* **Creation of XML Templates :** allows creation of dynamic sections in XML.

SQL server can return XML document using FOR XML clause. XML documents can be added to SQL Server database and you can use the OPENXML clause to display the data from the document as a relational result set. SQL Server 2000 supports XPath queries.

**Advantages and Disadvantages of Index.**

Advantages

* Speed up SELECT query
* Helps to make a row unique or without duplicates(primary, unique)
* If index is set to fill-text index, then we can search against large string values. for example to find a word from a sentence etc.

Disadvantages

* Indexes take additional disk space.
* indexes slow down INSERT,UPDATE and DELETE, but will speed up UPDATE if the WHERE condition has an indexed field.  INSERT, UPDATE and DELETE becomes slower because on each operation the indexes must also be updated.

**General SQL Interview Questions:**

**1. What is a Database?**  
A database is a collection of information in an organized form for faster and better access, storage and manipulation. It can also be defined as a collection of tables, schema, views, and other database objects.

**2. What is Data warehouse?**  
Data warehouse refers to a central repository of data from multiple sources of information. Those data are consolidated, transformed and made available for the mining as well as online processing.

**3. What is a Table in a Database?**  
A table is a database object used to store records in a field in the form of columns and rows that holds data.

**4. What is a Field in a Database?**  
A field in a Database table is a space allocated to store a particular record within a table.

**5. What is a Record in a Database?**  
A record (also called a row of data) is an ordered set of related data in a table.

**6. What is a column in a Table?**  
A column is a vertical entity in a table that contains all information associated with a specific field in a table.

**7. What is DBMS?**  
Database Management System is a collection of programs that enables a user to store, retrieve, update and delete information from a database.

**8. What are the types of DBMS?**  
There are two types of DBMS  
1. Relational Database Management System (RDBMS)  
2. Non-Relational Database Management System

**9. What is RDBMS?**  
RDBMS stands for **R**elational **D**atabase **M**anagement **S**ystem. RDBMS is a database management system (DBMS) that is based on the relational model. Data from a relational database can be accessed using Structured Query Language (SQL)

**10. What are the popular Database Management Systems in the IT Industry?**  
Oracle, MySQL, Microsoft SQL Server, PostgreSQL, Sybase, MongoDB, DB2, and Microsoft Access etc.,

**11. What is SQL?**  
SQL Overview: SQL stands for Structured Query Language. It is an American National Standard Institute (ANSI) standard. It is a standard language for accessing and manipulating databases. Using SQL, some of the action we could do are to create databases, tables, stored procedures (SP’s), execute queries, retrieve, insert, update, delete data against a database.

**12. What are the different types of SQL commands?**  
SQL commands are segregated into the following types:

* DDL – Data Definition Language
* DML – Data Manipulation Language
* DQL – Data Query Language
* DCL – Data Control Language
* TCL – Transaction Control Language

**13. What are the different DDL commands in SQL?**  
DDL commands are used to define or alter the structure of the database.

* CREATE: To create databases and database objects
* ALTER: To alter existing database objects
* DROP: To drop databases and databases objects
* TRUNCATE: To remove all records from a table but not its database structure
* RENAME: To rename database objects

**14. What are the different DML commands in SQL?**  
DML commands are used for managing data present in the database.

* SELECT: To select specific data from a database
* INSERT: To insert new records into a table
* UPDATE: To update existing records
* DELETE: To delete existing records from a table

**15. What are the different DCL commands in SQL?**  
DCL commands are used to create roles, grant permission and control access to the database objects.

* GRANT: To provide user access
* DENY: To deny permissions to users
* REVOKE: To remove user access

**16. What are the different TCL commands in SQL?**  
TCL commands are used to manage the changes made by DML statements.

* COMMIT: To write and store the changes to the database
* ROLLBACK: To restore the database since the last commit

**17. What is an Index?**  
An index is used to speed up the performance of queries. It makes faster retrieval of data from the table. The index can be created on one column or a group of columns.

**18. What are all the different types of indexes?**  
There are three types of indexes  
**1. Unique Index:** Unique Indexes helps maintain data integrity by ensuring that no two rows of data in a table have identical key values. A unique index can be applied automatically when a primary key is defined. It ensures that the values in the index key columns are unique.  
**2. Clustered Index:** Clustered Index reorders the physical order of the table and search based on the key values. There will be only one clustered index per table.  
**3. Non-Clustered Index:** Non-Clustered Index doesn’t alter the physical order of the table and maintains a logical order of the data. Each table can have many non-clustered indexes.

**19. What is the difference between Cluster and Non-Cluster Index?**  
The difference between the clustered and non-clustered index in SQL is as follows:  
**Clustered Index:**  
It is used for easy retrieval of data from the database and it is faster.  
One table can only have one clustered index  
It alters the way records are stored in a database as it sorts out rows by the column which is set to be clustered index.  
**Non-Clustered Index:**  
It is slower compared to the Clustered index.  
One table can have multiple non clustered index  
It doesn’t alter the way it was sorted but it creates a separate object within a table which points back to the original table rows after searching.

**20. What is a View?**  
A view is like a subset of a table which is stored logically in a database. A view is a virtual table. It contains rows and columns similar to a real table. The fields in the view are fields from one or more real tables. Views do not contain data of their own. They are used to restrict access to the database or to hide data complexity.



|  |  |
| --- | --- |
| 1 | CREATE VIEW view\_name AS SELECT column\_name1, column\_name2 FROM table\_name WHERE CONDITION; |

**21. What are the advantages of Views?**  
Some of the advantages of Views are

1. Views occupy no space
2. Views are used to simply retrieve the results of complicated queries that need to be executed often.
3. Views are used to restrict access to the database or to hide data complexity.

**22. What is a relationship and what are they?**  
Database Relationship is defined as the connection between the tables in a database. There are various database relationships namely  
1. One to One Relationship  
2. One to Many Relationship  
3. Many to One Relationship  
4. Self-Referencing Relationship

**23. What is a query?**  
A database query is a request for data or information from a database table or combination of tables. A database query can be either a select query or an action query.

**24. What is a Subquery?**  
A Subquery is a SQL query within another query. It is a subset of a Select statement whose return values are used in filtering the conditions of the main query.

**25. What are the types of subquery?**  
There are two types of subquery:  
**1. Correlated:** In a SQL database query, a correlated subquery is a subquery that uses values from the outer query in order to complete. Because a correlated subquery requires the outer query to be executed first, the correlated subquery must run once for every row in the outer query. It is also known as a synchronized subquery.  
**2. Non-Correlated:** A Non-correlated subquery is a subquery in which both outer query and inner query are independent to each other.

**26. What is Synchronized Subquery?**  
Refer Correlated Subquery.

**27. What is the difference between Local Variables and Global Variables?**  
**Local Variables:** Local variables can be used or exist only inside the function. These variables are not used or referred by any other functions. These are not known to other functions. Variables can be created whenever that function is called.  
**Global Variables:** Global variables can be used or exist throughout the program. Same variable declared in global cannot be used in functions. Global variables cannot be created whenever that function is called.

**28. What is data Integrity?**  
Data integrity defines the accuracy and consistency of the data stored in a database. It also defines integrity constraints to enforce business rules on the data when it is entered into an application or a database.

**29. What is Auto Increment in SQL?**  
It is one of the important Oracle DBA Interview Questions.  
Auto increment keyword allows the user to create a unique number to get generated when a new record is inserted into a table. Auto increment keyword can be used whenever Primary Key is used.

AUTO INCREMENT keyword is used in Oracle and IDENTITY keyword is used in SQL Server.

**30. What is a temp table?**  
**Ans.** A temp table is a temporary storage structure to store the data temporarily.

**31. How to avoid duplicate records in a query?**  
The SQL SELECT DISTINCT query is used to return only unique values. It eliminates all the duplicated values.  
[View Detailed Post](https://www.softwaretestingmaterial.com/sql-select-distinct/)

**32. What is the difference between Rename and Alias?**  
‘Rename’ is a permanent name given to a table or column  
‘Alias’ is a temporary name given to a table or column.

**33. What is a Join?**  
Join is a query, which retrieves related columns or rows from multiple tables.

**34. What are the different types of joins?**  
Types of Joins are as follows:

* INNER JOIN
* LEFT JOIN
* RIGHT JOIN
* OUTER JOIN

[View Complete Post](https://www.softwaretestingmaterial.com/sql-joins/)

**35. What is the difference between an inner and outer join?**  
An inner join returns rows when there is at least some matching data between two (or more) tables that are being compared.  
An outer join returns rows from both tables that include the records that are unmatched from one or both the tables.

**36. What are SQL constraints?**  
SQL constraints are the set of rules that enforced some restriction while inserting, deleting or updating of data in the databases.

**37. What are the constraints available in SQL?**  
Some of the constraints in SQL are – Primary Key, Foreign Key, Unique Key, SQL Not Null, Default, Check and Index constraint.

**38. What is a Unique constraint?**  
A unique constraint is used to ensure that there are no duplication values in the field/column.

**39. What is a Primary Key?**  
A *PRIMARY KEY* constraint uniquely identifies each record in a database table. All columns participating in a primary key constraint must not contain NULL values.

**40. Can a table contain multiple *PRIMARY KEY’s*?**

The short answer is no, a table is not allowed to contain multiple primary keys but it allows to have one composite primary key consisting of two or more columns.

**41. What is a Composite *PRIMARY KEY*?**  
Composite *PRIMARY KEY*is a primary key created on more than one column (combination of multiple fields) in a table.

**42. What is a *FOREIGN KEY*?**  
A *FOREIGN KEY* is a key used to link two tables together. A *FOREIGN KEY*in a table is linked with the *PRIMARY KEY* of another table.

**43. Can a table contain multiple *FOREIGN KEY’s*?**A table can have many *FOREIGN KEY’s.*

**44. What is the difference between *UNIQUE* and *PRIMARY KEY*constraints?**  
There should be only one *PRIMARY KEY* in a table whereas there can be any number of *UNIQUE* Keys.  
*PRIMARY KEY*doesn’t allow *NULL* values whereas Unique key allows NULL values.

**45. What is a *NULL* value?**  
A field with a *NULL* value is a field with no value. A *NULL* value is different from a zero value or a field that contains spaces. A field with a *NULL* value is one that has been left blank during record creation. Assume, there is a field in a table is optional and it is possible to insert a record without adding a value to the optional field then the field will be saved with a *NULL* value.

**46. What is the difference between NULL value, Zero, and Blank space?**  
As I mentioned earlier, Null value is field with no value which is different from zero value and blank space.  
*Null value* is a field with no value.  
*Zero* is a number  
*Blank space* is the value we provide. The ASCII value of space is CHAR(32).

**47. How to Test for *NULL* Values?**A field with a *NULL* value is a field with no value. *NULL* value cannot be compared with other NULL values. Hence, It is not possible to test for *NULL* values with comparison operators, such as =, <, or <>. For this, we have to use the *IS NULL*and *IS NOT* *NULL* operators.



|  |  |
| --- | --- |
| 1 | SELECT column\_names FROM table\_name WHERE column\_name IS NULL; |



|  |  |
| --- | --- |
| 1 | SELECT column\_names FROM table\_name WHERE column\_name IS NOT NULL; |

**48. What is SQL *NOT NULL* constraint?**  
*NOT NULL* constraint is used to ensure that the value in the filed cannot be a NULL

**49. What is a *CHECK*constraint?**  
*A CHECK* constraint is used to limit the value that is accepted by one or more columns.

E.g. ‘Age’ field should contain only the value greater than 18.



|  |  |
| --- | --- |
| 1 | CREATE TABLE EMP\_DETAILS(EmpID int NOT NULL, NAME VARCHAR (30) NOT NULL, Age INT CHECK (AGE &gt; 18), PRIMARY KEY (EmpID)); |

**50. What is a *DEFAULT*constraint?**  
*DEFAULT* constraint is used to include a default value in a column when no value is supplied at the time of inserting a record.

**51. What is Normalization?**  
Normalization is the process of table design to minimize the data redundancy.

**52. What are all the different Normalization?**  
There are different types of Normalization forms in SQL.

* First Normal Form (1NF)
* Second Normal Form (2NF)
* Third Normal Form (3NF)
* Boyce and Codd Normal Form (BCNF)

**53. What is Denormalization?**  
Denormalization is a database optimization technique used to increase the performance of a database infrastructure. It involves in the process of adding redundant data to one or more tables. In a normalized database, we store data in separate logical tables and attempt to minimize redundant data.

**54. What is Stored procedure?**A Stored Procedure is a collection of SQL statements that have been created and stored in the database to perform a particular task. The stored procedure accepts input parameters and processes them and returns a single value such as a number or text value or a result set (set of rows).

**55. What is a Trigger?**  
A Trigger is a SQL procedure that initiates an action in response to an event (Insert, Delete or Update) occurs. When a new Employee is added to an Employee\_Details table, new records will be created in the relevant tables such as Employee\_Payroll, Employee\_Time\_Sheet etc.,

**56. Explain SQL Data Types?**In SQL Server, each column in a database table has a name and a data type. We need to decide what type of data to store inside each and every column of a table while creating a SQL table.

[View Detailed Post](https://www.softwaretestingmaterial.com/sql-data-types/)

**57. What are the possible values that can be stored in a *BOOLEAN* data field?**  
*TRUE* and *FALSE*

**58. What is the largest value that can be stored in a *BYTE* data field?**The largest number that can be represented in a single byte is 11111111 or 255. The number of possible values is 256 (i.e. 255 (the largest possible value) plus 1 (zero), or 2**8).**

**59. What are Operators available in SQL?**  
SQL Operator is a reserved word used primarily in an SQL statement’s WHERE clause to perform operations, such as arithmetic operations and comparisons. These are used to specify conditions in an SQL statement.

There are three types of Operators.

1. Arithmetic Operators
2. Comparison Operators
3. Logical Operators

[View Detailed Post](https://www.softwaretestingmaterial.com/sql-operators/)

**60. Which TCP/IP port does SQL Server run?**  
By default, it is 1433

**61. List out the ACID properties and explain?**  
Following are the four properties of ACID. These guarantees that the database transactions are processed reliably.

* **Atomicity**
* **Consistency**
* **Isolation**
* **Durability**

**62. Define the SELECT INTO statement.**  
The SELECT INTO statement copies data from one table into a new table. The new table will be created with the column-names and types as defined in the old table. You can create new column names using the AS clause.



|  |  |
| --- | --- |
| 1 | SELECT \* INTO newtable FROM oldtable WHERE condition; |

**63. What is the difference between Delete, Truncate and Drop command?**  
The difference between the Delete, Truncate and Drop command is

* Delete command is a DML command, it is used to delete rows from a table. It can be rolled back.
* Truncate is a DDL command, it is used to delete all the rows from the table and free the space containing the table. It cant be rolled back.
* Drop is a DDL command, it removes the complete data along with the table structure(unlike truncate command that removes only the rows). All the tables’ rows, indexes, and privileges will also be removed.

**64. What is the difference between Delete and Truncate?**  
The difference between the Delete, and Truncate are

| **DELETE** | **TRUNCATE** |
| --- | --- |
| Delete statement is used to delete rows from a table. It can be rolled back. | Truncate statement is used to delete all the rows from the table and free the space containing the table. It cant be rolled back. |
| We can use WHERE condition in DELETE statement and can delete required rows | We cant use WHERE condition in TRUNCATE statement. So we cant delete required rows alone |
| We can delete specific rows using DELETE | We can only delete all the rows at a time using TRUNCATE |
| Delete is a DML command | Truncate is a DDL command |
| Delete maintains log and performance is slower than Truncate | Truncate maintains minimal log and performance wise faster |
| We need DELETE permission on Table to use DELETE command | We need at least ALTER permission on the table to use TRUNCATE command |

**65. What is the difference between Union and Union All command?**  
This is one of the tricky SQL Interview Questions. Interviewer may ask you this question in another way as what are the advantages of Union All over Union.

Both Union and Union All concatenate the result of two tables but the way these two queries handle duplicates are different.

***Union:***It omits duplicate records and returns only distinct result set of two or more select statements.  
***Union All:*** It returns all the rows including duplicates in the result set of different select statements.

Performance wise Union All is faster than Union, Since Union All doesn’t remove duplicates. Union query checks the duplicate values which consumes some time to remove the duplicate records.

Assume: *Table1* has 10 records, *Table2* has 10 records. Last record from both the tables are same.

If you run Union query.



|  |  |
| --- | --- |
| 1  2  3 | SELECT \* FROM Table1  UNION  SELECT \* FROM Table2 |

**Output:** Total 19 records

If you run Union query.



|  |  |
| --- | --- |
| 1  2  3 | SELECT \* FROM Table1  UNION ALL  SELECT \* FROM Table2 |

**Output:** Total 20 records

*Data type of all the columns in the two tables should be same.*

**66. What is CLAUSE in SQL?**SQL CLAUSE helps to limit the result set by providing a condition to an SQL Query. A CLAUSE helps to filter the rows from the entire set of records. SQL CLAUSES are WHERE & HAVING.

**67. What is the difference between Having and Where clause?**Where clause is used to fetch data from a database that specifies particular criteria whereas a Having clause is used along with ‘GROUP BY’ to fetch data that meets particular criteria specified by the Aggregate functions. Where clause cannot be used with Aggregate functions, but the Having clause can.

**68. What are aggregate functions in SQL?**SQL aggregate functions return a single value, calculated from values in a column. Some of the aggregate functions in SQL are as follows

* AVG() – This function returns the average value
* COUNT() – This function returns the number of rows
* MAX() – This function returns the largest value
* MIN() – This function returns the smallest value
* ROUND() – This function rounds a numeric field to the number of decimals specified
* SUM() – This function returns the sum

[View Detailed Post](https://www.softwaretestingmaterial.com/sql-aggregate-functions/)

**69. What are string functions in SQL?**  
SQL string functions are used primarily for string manipulation. Some of the widely used SQL string functions are

* LEN() – It returns the length of the value in a text field
* LOWER() – It converts character data to lower case
* UPPER() – It converts character data to upper case
* SUBSTRING() – It extracts characters from a text field
* LTRIM() – It is to remove all whitespace from the beginning of the string
* RTRIM() – It is to remove all whitespace at the end of the string
* CONCAT() – Concatenate function combines multiple character strings together
* REPLACE() –  To update the content of a string.

[View Detailed Post](https://www.softwaretestingmaterial.com/sql-string-functions/)

**70. What are user defined functions?**  
As the name suggests these are written by users as per their requirement. User-defined functions are the functions written to use a logic whenever required.

**71. What are all types of user-defined functions?**  
There are three types of user-defined function, namely  
• Scalar Functions  
• Inline Table-valued functions  
• Multi-statement valued functions

Scalar functions return unit, variant defined the return clause.  
Inline Table-valued functions and Multi-statement valued functions return table as a return.

**72. What is Self-Join?**  
A self-join is a join in which a table is joined with itself, especially when the table has a Foreign Key which references its own Primary Key.

**73. What is Cross-Join?**  
Cross join produces a result set which is the number of rows in the first table multiplied by a number of rows in the second table if no WHERE clause is used along with Cross join. This kind of result is known as Cartesian Product. If suppose, Where clause is used in cross join then the query will work like an Inner join.

**74. What is Collation?**  
Collation is defined as a set of rules that determine how character data can be sorted as well as compared. Character data is sorted using rules that define the correct character sequence along with options for specifying case-sensitivity, character width, accent marks, kana character types.

**75. What are all different types of collation sensitivity?**  
Different types of collation sensitivity are as follows  
Case Sensitivity: A and a and B and b.  
Kana Sensitivity: Japanese Kana characters.  
Width Sensitivity: Single byte character and double byte character.  
Accent Sensitivity.

* Venkat sir +91-8553132092

SQL SERVER

|  |  |
| --- | --- |
| **ORACLE** | **SQLSERVER** |
| STARTED IN 1979 | STARTED IN 1989 |
| **PLATFORM INDEPENDENT** - ORACLE CAN BE INSTALLED IN ANY OPERATING SYSTEM. | **PLATFORM DEPENDENT** - THIS CAN BE INSTALLED ONLY IN WINDOWS OPEARATING SYSTEMS. |
| THIS IS VERY COSTLY DATABASE IN THE MARKET. | THIS IS VERY CHEAPER WHEN COMPARED TO ORACLE. |
| LESS GUI AND ONLY COMMAND PROMT QUERYING IS ALLOWED. TO ACHIVE GUI WE NEED TO INSTALL THIRD PARTY TOOLS. | WE CAN USE COMMAND PROMPT AS WELL GUI AND MOSTLY GUI IS USED ACROSS ALL. |
| THIS IS MORE SECURED | THIS WAS LESS SECURED - NOW IN THE LATER VERSIONS IT IS SECURED. |
| ORACLE PROVIDES ONLY DATABASE. | SQL SERVER PROVIDES DATABASE AS WELL AS BI SUITE. |

**SQL SERVER 1.0…….**

**SQL SERVER 8.0 – SQL SERVER 2000**

**DB**

**DTS – DATA TRANSFORMATION SERVICES. (INITIAL ETL)**

**MSAS – MICROSOFT ANALYSIS SERVICES**

**SQL SERVER 2005 – CODE NAME - KATMAI**

**DB - DATABASE**

**SSIS – SQL SERVER INTEGRATION SERVICES – ETL (EXTRACT TRANSFORM LOAD)**

**EXTENSION OF DTS**

**SSAS - SQL SERVER ANALYSIS SERVICES**

**SSRS - SQL SERVER REPOTING SERVICES (REPORTS)**

**SSNS - SQL SERVER NOTIFCATION SERVICES**

**SQL AGENT – TOOL USED TO SCHEDULE THE JOBS.**

**SQL SERVER 2008/2008 (R2) – CODE NAME - KILLIMANJARO**

**DB - DATABASE**

**SSIS – SQL SERVER INTEGRATION SERVICES – ETL (EXTRACT TRANSFORM LOAD)**

**SSAS - SQL SERVER ANALYSIS SERVICES**

**SSRS - SQL SERVER REPOTING SERVICES (REPORTS)**

REMOVED SSNS

**SQL AGENT – TOOL USED TO SCHEDULE THE JOBS.**

**SQL SERVER 2012 – CODE NAME – DENALI**

**SQL SERVER 2014**

**DB – DATABASE**

**DQS – DATA QUALITY SERVICES.**

**SSIS – SQL SERVER INTEGRATION SERVICES – ETL (EXTRACT TRANSFORM LOAD)**

**SSAS - SQL SERVER ANALYSIS SERVICES**

**SSRS - SQL SERVER REPOTING SERVICES (REPORTS)**

**MDS – MASTER DATA SERVICES**

**SQL AGENT – TOOL USED TO SCHEDULE THE JOBS.**

**SQL CLASSES :**

1. **DESIGN**
2. **IMPLEMENTATION**
3. **MAINTENANCE – ADMINSITRATION**

**DESIGN:**

**BASICS**

**SQL SERVER ENVIRONMENT – DONE**

**DATA INTEGRITY –** IMPORTMANT

**NORMALIZATION -** IMPORTMANT

**DATABASE DIAGRAMS**

**DDL, DML.**

**IMPLEMENTATION:**

**SUBQUERIES AND JOINS**

**VIEWS**

**STORED PROCEDURES**

**USER DEFINED FUNCTIONS**

**TRANSACTIONS AND LOCKS**

**TRIGGERS**

**CURSORS**

**INDEXES**

**ADMIN**

**BACKUP AND RESTORE**

**SCHEDULING PROGRAM OR PACKAGES (ETL)**

**DATABASE :**

**BASE FOR DATA.**

**IT IS NOTHING BUT A LOGICAL CONTAINER FOR SET OF RELATED OBJECTS**

**OBJECTS → DATABASES, TABLES, PROCEDURES, FUNCTIONS, VIEWS, ETC..**

**DBMS – DATABASE MANAGEMENT SYSTEM.**

1. *NDBMS – NETWORK DATABASE MANAGEMENT SYSTEM*
2. *HDBMS – HEIRARCHICHAL DATABASE MANAGEMENT SYSTEM*
3. **RDBMS – RELATIONAL DATABASE MANAGEMENT SYSTEM**

**HERE WE USE SQL – STRUCTURED QUERY LANGUAGE TO PERFORM OPERATIONS IN DATABASE.**

**RDBMS:**

**RDBMS = RELATIONS + DBMS**

**MAINTAINS RELATIONSHPS AMONG THE TABLES**

**FAST ACCESSING OF DATA FROM TABLES**

**PROVIDES MORE SECURITY**

**RDBMS FOLLOW E.F.CODD RULES (12 RULES)**

**UGSERVER – RDBMS , WHILE DEVELOPING THE SOFTWARE WE NEED TO FOLLOW AND IMPLEMENT 12 RULES OF E.F.CODD**

**EX: SQL SERVER, ORACLE, DB2, MYSQL, POSTRAGESQL, TERRADATA, MS ACCESS, ETC..**

**TWO TYPES OF DATABASES:**

1. **SYSTEM DATABASES**

**MASTER**

**MODEL**

**MSDB**

**TEMPDB**

1. **USER DEFINED DATABASES**

**DATABASES CREATED BY DEVELOPERS**

**CREATE DATABASE:**

**TO CREATE A DATABASE IN SQL SERVER WE REQUIRE ATLEAST TWO FILES:**

1. **MASTER DATA FILE**

**DATA FILE STORES TABLE AND OTHER OBJECTS.**

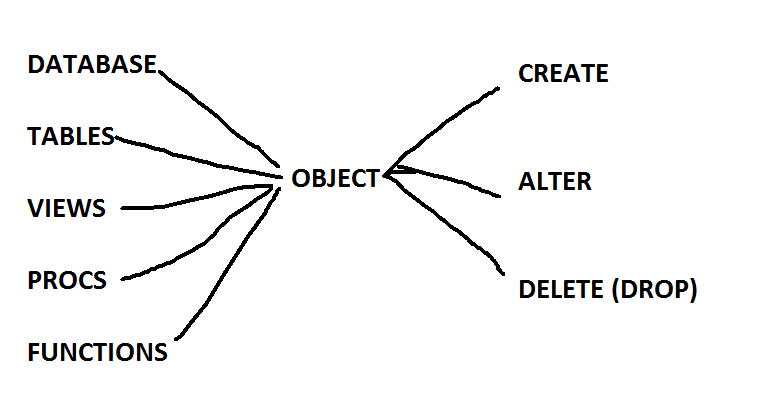
**ACTUAL DATA IS STORED HERE.**

**.MDF AND .NDF**

1. **LOG DATA FILE**

**LOG FILE WHICH IS USED TO MAINTAIN THE LOG INFORMATION.**

**NDF – NEXT DATA FILE**

****

**SYNTAX:**

**CREATE DATABASE DBNAME**

CREATE DATABASE MSSQL\_aug

CREATE DATABASE [MSSQL\_aug\_2016] ONPRIMARY

( NAME = MSSQL\_aug\_2016',

FILENAME=N'C:\Program Files\Microsoft SQL Server\MSSQL10.MSSQLSERVER\MSSQL\DATA\ MSSQL\_aug\_2016.mdf',

SIZE = 2048KB ,

MAXSIZE=20GB,

FILEGROWTH = 1024KB )

LOGON

( NAME = MSSQL\_aug\_2016\_log',

FILENAME=N'C:\Program Files\Microsoft SQL Server\MSSQL10.MSSQLSERVER\MSSQL\DATA\ MSSQL\_aug\_2016\_log.ldf',

SIZE = 1024KB ,

MAXSIZE=5GB,

FILEGROWTH = 10%)

**ALTER DATABASE:**

**ALTER DATABASE IS NOTHING BUT MODIFYING THE STURCTURE OF DATABASE.**

**ALTER DATABASE TO ADD A DATA FILE:**

ALTER DATABASE MSSQL\_CLASSES

ADD FILE

(

NAME=PQR,

FILENAME=' C:\Program Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER\MSSQL\DATA

\ MSSQL\_CLASSES.NDF',

SIZE=2MB,

MAXSIZE=20GB,

FILEGROWTH=1MB

)

**ALTER DATABASE TO ALTER SIZE**

ALTER DATABASE YESM\_DATAALEMU

MODIFY FILE

(

NAME=PQR ,

SIZE=10GB

)

**ALTER DATABASE TO REMOVE DATA FILE:**

ALTER DATABASE YESM\_DATAALEMU

REMOVE FILE PQR

**DROP DATABASE:**

**DROP DATABASE DBNAME**

DROP DATABASE MSSQL\_CLASSES

**USING GUI WE CAN DELETE DATABASE**

**RIGHT CLICK ON THE DATABASE NAME → CLICK ON DELETE OPTION →POPUP OPENS → SELECT THE CHECK BOX “CLOSE EXISTING CONNECTIONS” → CLICK OK.**

**RESULT WINDOW: CONTROL+R**

--CREATING TABLES

SYNTAX:

CREATE TABLE TABLENAME

(

COLUMNNAME DATATYPE CONSTRAINTS,

COLUMNNAME DATATYPE ,

COLUMNNAME DATATYPE ,

..............

)

**DATATYPES:**

1. **NUMERIC DATATYPES:**

**INTEGER**

**TINYINT – MOSTLY USED FOR COLUMN AGE (1-155)**

**SMALLINT**

**INT**

**BIGINT**

**FLOATING TYPE**

**REAL(PRECISION)**

**FLOAT(PRECISION)**

**NUMERIC(PRECISION, SCALE)**

**DECIMAL(PRECISION,SCALE)**

**PRECISION → TOTAL NO. OF DIGITS**

**SCLAE→ NO OF DIGITS AFTER DECIMAL POINT**

**FLOAT(5) – 178.95, 17.985**

**DECIMAL(3,2) – 178.95, 185.64**

**MONEY**

**SMALLMONEY**

**MONEY**

DECLARE @X SMALLMONEY

SET @X=5000

PRINT @X

DECLARE @Y MONEY

SET @Y=5000000

PRINT @Y

1. **BINARY DATATYPES:**

**USED TO STORE BINARY DATA. (COLLECTION OF BITS – 0 OR 1)**

**BIT**

**BINARY(SIZE) – FIXED SIZE**

**VARBINARY(SIZE) – VARIABLE LEGNTH**

**EX:**

**BINARY(4) – ALLOCATE 4 BYTES OF MEMORY IMMEDIATELY**

**VARBINARY(4) – BASED ON INPUT PASSED, SYSTEM WILL ALLOCATE MEMORY DYNAMICALLY.**

1. **DATETIME DATATYPE:**

**SMALLDATETIME**

**DATETIME**

**DATE**

**TIME**

1. **STRING DATATYPES:**

**CHARACTERS CAN BE REPRESENTED IN 2 WAYS:**

**ASCII (ENGLISH CHARACTERS)**

**UNICODE (UNIVERSAL CODE →LANGUAGES OTHER THAN ENGLISH AS WELL ENGLISH)**

**ASCII: EACH CHARACTER TAKES 1 BYTE OF MEMORY. SUITABLE FOR LOCAL LANGUAGE ( ENGLISH)**

**CHAR(SIZE) – FIXED LENGTH**

**VARCHAR(SIZE) – VARIABLE LENGTH**

**TEXT – USED TO STORE TEXT DOCUMENTS… ( 2GB)**

**CHAR(SIZE):**

**WHENEVER WE DEFINE A VARIABLE OR COLUMN WITH CHAR DATATYPE THEN IT WILL ALLOCATE MEMORY IMMEDIATELY WITH THE SPECIFIED SIZE.**

**CHAR(5) → 5\*1=5 BYTES.**

**CHAR(1)**

**CUSTOMERNAME CHAR(10)**

**EMPNAME CHAR(10)**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **R** | **A** | **M** |  |  |  |  |  |  |  |

**0 1 2 3 4 5 6 7 8 9**

**RAM – ABOVE 3 TO 9 CELLS ARE WASTED (MEMORY GOT WASTED)**

**KRISHNA**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **K** | **R** | **I** | **S** | **H** | **N** | **A** |  |  |  |

**0 1 2 3 4 5 6 7 8 9**

**7,8,9 CELLS GOT WASTED (MEMORY GOT WASTED)**

**TO OVERCOME THIS PROBLEM..VARCHAR(SIZE) DATATYPE.**

**VARCHAR(SIZE)**

**VARCHAR → VARIABLE CHARACTER**

**VARIABLE – WHICH MAY CHANGE.**

**VARCHAR(SIZE) – VARIABLE SIZE**

**CUSTOMERNAME VARCHAR(10)**

**NOTE: IT WON’T ALLOCATE MEMORY IMMEDIATELY.**

**MEMORY IS ALLOCATED AT RUNTIME BASED ON INPUT VALUE.**

**CUSTOMERNAME = RAM**

|  |  |  |
| --- | --- | --- |
| **R** | **A** | **M** |

**0 1 2**

**RAM -3 CELLS ARE ALLOCATED AT RUNTIME AND MEMORY IS NOT WASTED.**

**CUSTOMERNAME=KRISHNA**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **K** | **R** | **I** | **S** | **H** | **N** | **A** |

**0 1 2 3 4 5 6**

|  |  |
| --- | --- |
| **CHAR(SIZE)** | **VARCHAR(SIZE)** |
| **MEMORY IS ALLOCATED IMMEDIATELY ONCE WE DECLARE** | **MEMORY IS NOT ALLOCATED IMMEDIATELY AFTER WE DECLARE** |
| **MEMORY WASTE WILL BE THERE** | **NO MEMORY WASTE - MEMORY IS ALLOCATED AT RUNTIME AS PER GIVEN INPUT** |

**TEXT :**

**THIS WILL STORE DATA UPTO 2GB.**

**COMMENTS TEXT**

**FEEDBACK TEXT**

**HERE WE CAN ENTER PARAGRAPH’S**

**UNICODE:**

**IT CAN STORE ENGLISH LANGUAGE AS WELL AS OTHER LANGUAGE DATA ALSO (FRENCH, GERMEN, PORTUGUESE, TAMIL, TELUGU, HINDI…ENGLISH,)**

**EACH CHAR TAKES 2 BYTES.**

1. **NCHAR(SIZE) – FIXED SIZE – ALLOCATES MEMORY WHEN DECLARED**
2. **NVARCHAR(SIZE)- VARIABLE SIZE- ALLOCATE MEMORY AT RUNTIME.**
3. **NTEXT – THIS WILL STORE DATA UPTO 2 GB**

**N-UNICODE – NATIONALIZATION.**

**NCHAR(10)**

**Customer name Nchar(10) RAMU**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| రా | ము |  |  |  |  |  |  |  |  |

0 1 2 3 4 5 6 7 8 9

**రాము**– **above 2 to 9 cells are wasted (memory got wasted)**

**HANUMANTHU- హనుమంతు**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| హ | ను | మం | తు |  |  |  |  |  |  |

0 1 2 3 4 5 6 7 8 9

**4 TO 9 CELLS GOT WASTED (MEMORY GOT WASTED)**

హనుమంతు

**TO OVERCOME THIS PROBLEM..NVARCHAR(SIZE) DATATYPE.**

**NVARCHAR(SIZE)**

**NVARCHAR → VARIABLE CHARACTER OF ANY LANGUAGE**

**VARIABLE – WHICH MAY CHANGE.**

**NVARCHAR(SIZE) – VARIABLE SIZE**

**CUSTOMERNAME NVARCHAR(10)**

**NOTE: IT WON’T ALLOCATE MEMORY IMMEDIATELY.**

**MEMORY IS ALLOCATED AT RUNTIME BASED ON INPUT VALUE.**

**CUSTOMERNAME =RAMU**

|  |  |
| --- | --- |
| **రా** | **ము** |

0 1

**రాము-2 CELLS ARE ALLOCATED AT RUNTIME AND MEMORY IS NOT WASTED.**

**CUSTOMERNAME= HANUMANTHU- హనుమంతు**

|  |  |  |  |
| --- | --- | --- | --- |
| హ | ను | మం | తు |

0 1 2 3

|  |  |
| --- | --- |
| **NCHAR(SIZE)** | **NVARCHAR(SIZE)** |
| **MEMORY IS ALLOCATED IMEEDIATELY ONCE WE DECLARE** | **MEMORY IS NOT ALLOCATED DYNAMICALLY AFTER WE PASS THE VALUE** |
| **MEMORY WASTE WILL BE THERE** | **NO MEMORY WASTE - MEMORY IS ALLOCATED AT RUNTIME AS PER GIVEN INPUT** |
| **ACCEPTS ANY LANGUAGE ALONG WITH ENGLISH** | **ACCEPTS ANY LANGUAGE ALONG WITH ENGLISH** |

1. **WHAT IS THE DIFFERENCE BETWEEN CHAR AND VARCHAR?**
2. **WHAT IS THE DIFFERENCE BETWEEN CHAR AND NCHAR?**
3. **WHAT IS THE DIFFERENCE BETWEEN VARCHAR AND NVARCHAR?**

CREATETABLE STUDENTS

(

SID INT,

SNAME VARCHAR(20),

SAGE TINYINT,

JOINDATE DATETIME,

FEES MONEY

)

**CLICK ON EXECUTE BUTTON OT PRESS F5 KEY.**

**TO SEE THE TABLE DATA:**

SELECT\*FROM STUDENTS

**CREATE TABLE, ALTER TABLE, DROP TABLE,**

**CREATE VIEW, ALTER VIEW, DROP VIEW…..**

**DML OPERATIONS: DATA MANIPULATION LANGUAGE**

**INSERT, UPDATE, DELETE**

**INSERT TO TABLE**

**SYNTAX:**

**INSERT INTO TABLENAME VALUES (..,…,…..)**

**HERE *INTO* IS OPTIONAL**

INSERT INTO STUDENTS VALUES

(1,'VENKAT','1234567890',30,'1-12-2012'),

(2,'TEJASWI','1234567890',25,'2-12-2012'),

(3,'BHAVNI','1234567890',24,'3-12-2012'),

(4,'THANU','1234567890',32,'4-12-2012'),

(5,'AKHILA','1234567890',23,'5-12-2012'),

(6,'USHA','1234567890',22,'6-12-2012'),

(7,'PADMA','1234567890',27,'7-12-2012'),

(8,'SHEWTHA','1234567890',26,'8-12-2012'),

(9,'SIDDARTH','1234567890',21,'9-12-2012'),

(10,'MAHITHA','1234567890',24,'10-12-2012'),

(11,'MINATI','1234567890',22,'11-12-2012'),

(12,'RADHIKA','12345678',23,'12-1-2013')

SELECT \* FROM STUDENTS

INSERT INTO STUDENTS(STID,STNAME,STMOBILE,JOINDATE)VALUES

(13,'MANOJ','456123525',1-13-2013)

SELECT \* FROM STUDENTS

INSERT STUDENTS VALUES

(14,'SUNIL','14567890',33,'2-12-2012')

SELECT \* FROM STUDENTS

**DELETE SYNTAX:**

**DELETE FROM TABLENAME WHERE <CONDITION>**

**NOTE: IF WE DON’T PASS CONDITION THEN TOTAL TABLE DATA WILL BE DELETED**

**EX: DELETE FROM TABLENAME – TATOL DATA WILL BE DELETED.**

--LET US DELETE TOTAL TABLE DATA ONCE..

DELETE FROM STUDENTS

SELECT \* FROM STUDENTS

--EXECUTE THE ABOVE INSERT STATEMENTS TO LOAD THE DATA

DELETE FROM STUDENTS WHERE STNAME='MANOJ' AND STID=13

SELECT \* FROM STUDENTS

**UPDATE:**

SYNTAX:

UPDATE TABLE TABLENAME SET COLX=VALUE

WHERE COLY=VALUE..

UPDATE STUDENTS SET STNAME='BHAVANI JADALA'

WHERE STID=3

SELECT \* FROM STUDENTS

UPDATE STUDENTS SET STMOBILE='8553132092'

WHERE STID=1 AND STNAME='VENKAT'

SELECT \* FROM STUDENTS

--DISPLAY ALL STUDENTS INFORMATION

SELECT\*FROM STUDENTS

--DISPLAY STUDENST INFO WHOSE AGE MORE THAN 28

SELECT\*FROM STUDENTS WHERE SAGE>28

--DISPLAY ALL SIDS, SNAMES

SELECTSID,SNAME FROM STUDENTS

--DISPLAY ALL SIDS, SNAMES, SAGE WHOSE AGE MORE THAN 28

SELECT STID,STNAME,STAGE FROM STUDENTS

WHERE STAGE>28

**NULL: UNKNOWN VALUE.**

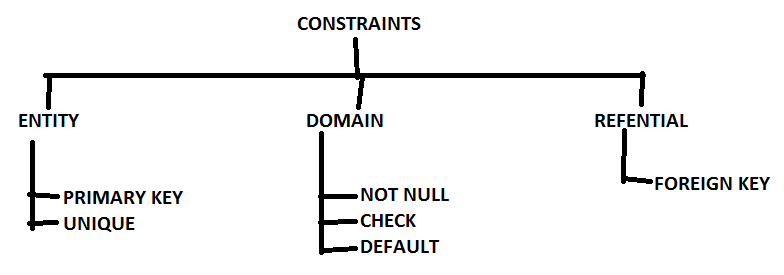
INSERTINTO STUDENTS VALUES(15,'VARUN', '2014-10-10',NULL, NULL)

--DISPLAY STUDENTS INFO WHOSE AGE IS UNKNOWN

SELECT\*FROM STUDENTS WHERE SAGE ISNULL

**Data Integrity/constraints**

**The concept of enforcing business rules into database tables can be called as data integrity.**

****

**Types of Data Integrity:**

**Entity integrity**

**Domain integrity**

**Referential integrity**

**Entity integrity:**

**This can be implemented with the help of primary key and unique.**

**Primary key: NOTHING BUT COLLECTION OF ONE OR MORE COLUMNS USED TO IDENTIFY THE ROW.**

**This is used to identify the entire row.**

**Primary key is defined for a column or for combination of columns (composite key).**

**Primary key column don’t allow duplicate values.**

**Primary key column don’t allow null values.**

**A table can have only one primary key column.**

**Once primary key is created, in the background clustered index is created on the same column.**

**A table can have only one clustered index.**

**A table can have only one primary key. Primary key can be defined on a column then it is primary, if primary key is defined on more than column i.e combination of column then it is called as composite key.**

**Unique:**

**This is also used to identify the rows.**

**It allows all the data for once including null value.**

**A table can have more than one unique column.**

**Once unique constraint is created, in the background unique non-clustered index is created on the same column.**

Create table BRANCHES

(

branchId int PRIMARY KEY IDENTITY(1,1),

bname varchar(20),

bmgrName varchar(20)

)

Select \* from BRANCHES

INSERT [dbo].[BRANCHES] VALUES

('Bommasandra', 'Abdul'),

('BTM', 'Kareem'),

('Bommanahalli', 'Ram'),

('Jayanagar', 'Rahul'),

('Banashanakari', 'Sudha'),

('Madiwala', 'Naren'),

('RTNagar', 'Neeraj'),

('Vijayanagar', 'Vijay'),

('JP nagar', 'raju'),

('Domlur', 'manju'),

('Sankarmat', 'manju'),

('Dodda bengaluru', 'kishore'),

('Chennai', 'Ramanatha'),

('BTM', 'Suresh'),

('BTM', 'Rosha'),

('BTM', 'Maju')

SELECT\*FROM BRANCHES

create table emp\_aug

(

empid int primary key identity(1,1),

ename varchar(20),

empmobile varchar(20) unique

)

insert into emp\_aug values ('sreekanth','8553132092')

select \* from emp\_aug

insert into emp\_aug values ('manoj','8553132092') --error

insert into emp\_aug values ('manoj','8553132033')

insert into emp\_aug values ('thanu','5896548654')

select \* from emp\_aug

delete from emp\_aug

--(3 row(s) affected)

select \* from emp\_aug

insert into emp\_aug values ('sreekanth','8553132092')

insert into emp\_aug values ('manoj','8553132093')

insert into emp\_aug values ('thanu','5896548654')

select \* from emp\_aug

------truncate

truncate table emp\_aug --Command(s) completed successfully.

insert into emp\_aug values ('sreekanth','8553132092')

insert into emp\_aug values ('manoj','8553132093')

insert into emp\_aug values ('thanu','5896548654')

select \* from emp\_aug

insert into emp\_aug values

('kkk','998565965'),

('aziza',null),

('Namrata','56565656565')

select \* from emp\_aug

Domain Integrity:

**Domain** is nothing but list of possible values in a column. It can be implemented with the help of below:

1. NOT NULL
2. DEFAULT
3. CHECK
4. **Not null:**

To restrict null values in a column.

While inserting, row value must be provided for not null column.

Not null constraint can be mixed with any other constraints.

Ex

ename varchar(20) not null

Emobile varchar(20) not null

Emobile varchar(20) unique not null

1. **DEFAULT:**

To provide value for a column.

Studentage tinyint default(20)

1. **CHECK:**

Check constraint is used to check the values in a column with user defined condition.

Customerbalance money check(Customerbalance>=500)

Gender char(1) check(gender=’M’ or gender=’F’)

Create table students\_new

(

Stid int primary key identity(1,1),

Stname varchar(20)not null,

Stage tinyint default(20),

Stmobile varchar(20)unique notnull,

Stfees money check(stfees>5000)

)

select\*from students\_new

insert into students\_new values('ssss',null,'8585858585',5000)

select \* from students\_new

insert into students\_new(Stname,Stmobile,Stfees)values('aaaa','456565656',5000)

select \* from students\_new

insert into students\_new values

(null,25,'12355445',6000) --insert fails stname should not null

insert into students\_new values

('Sindhu',25,'12355445',6000)

select \* from students\_new

insert into students\_new values

('Chandu',26,'12355445',7000) -- insert fails due to mobile num duplicate - unique

insert into students\_new values

('Chandu',26,null,7000) --insert fails mobile num should not null

insert into students\_new values

('Chandu',26,'12355666',7000)

select \* from students\_new

insert into students\_new values

('Bindhu',28,'22255666',4500) --insert fails due to fees is less than 5000 - check

insert into students\_new values

('Bindhu',28,'22255666',5500)

select \* from students\_new

1. **Referential integrity**

Rdbms – relational database management system.

It is nothing but concept of maintaining relationships among the tables.

This can be implemented using ‘foreign key’

Relationship is nothing but dependency among the tables.

There are 3 types of relationships:

1. One to one relationship
2. One to many relationship or many to one relationship
3. Many to many relationship
4. **One to one relationship:**

**A row in one table associated with one row in other table.**

Create table employee\_1\_1

(

Eid int primary key identity(1,1),

Ename varchar(20),

Emanager varchar(20)

)

INSERT INTO employee\_1\_1 VALUES

('VENKAT','JACOB'),

('BHAVANI','NANDU')

create table emp\_details\_1\_1

(

Eid int foreign key references employee\_1\_1(Eid) unique,

Ename varchar(20),

passport varchar(10),

ssn int,

Dlincense varchar(20),

maritalstatus varchar(10)

)

INSERT INTO emp\_details\_1\_1 VALUES

(1,'VENKAT','UU7U7U7',2336556,'DDDDDD','M')

--TRY TO INSERT THE SAME RECORD ONE MORE TIME

INSERT INTO emp\_details\_1\_1 VALUES

(1,'VENKAT','UU7U7U7',2336556,'DDDDDD','M')

**Master child**

****

1. **One to Many relationship or many to one relationship:**

A row in one table associated with many rows in other table.

Create table department

(

deptid int primary key identity(1,1),

deptname varchar(20) not null

)

Insert into department values('HR'),('Finance'),('Sales'),('Marketing')

select\*from department

create table employees\_1\_m

(

empid int primary key identity(1,1),

ename varchar(20)not null,

egae tinyint,

emobile varchar(20)unique,

esalary money,

deptid int foreign key references department(deptid)

)

--1,2,3,4

Insert into employees\_1\_m values

('sreekanth',28,'8553132092',5000,2),

('Aziza',22,'898989898',6000,1),

('Reshma',25,'4564564567',7000,1),

('Mehdi',24,'7897897958',6500,2),

('Durga',26,'6546546545',7200,3),

('Namrata',23,'3213213212',6400,4),

('William',25,'3693693696',8000,3),

('Astha',24,'1471471474',7600,4),

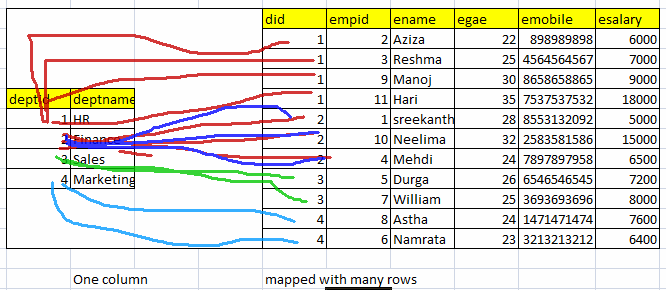
('Manoj',30,'8658658865',9000,1),

('Neelima',32,'2583581586',15000,2),

('Hari',35,'7537537532',18000,1)

select did,\*from employees orderby 1

select did,\*from employees orderby 1



1. **Many to many relationship:**

Many rows in one table associated with many rows in other table.

This can be achieved with the help of linking table or bridge table or associative table

create table authors

(

authorid int primary key identity(1,1),

authname varchar(20)

)

INSERTINTO authors VALUES

('SREEKANTH'),('RASHMI'),('MANOJ')

SELECT\*FROM authors

create table BOOKS

(

BOOKid int primary key identity(1,1),

BOOKTITLE varchar(20)

)

INSERT INTO BOOKS VALUES

('SQL'),('JAVA'),('.NET'),('ETL TESTING')

SELECT\*FROM BOOKS

CREATE TABLE BOOKS\_AUTHORS

(

AUTHORID INT FOREIGN KEY REFERENCES AUTHORS(AUTHORID),

BOOKID INT FOREIGN KEY REFERENCES BOOKS(BOOKID)

)

INSERTINTO BOOKS\_AUTHORS VALUES

(1,1),

(1,2),

(1,4),

(2,2),

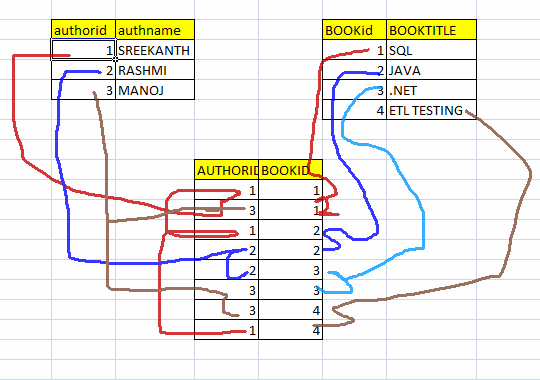
(2,3),

(3,1),

(3,3),

(3,4)

SELECT\*FROM BOOKS\_AUTHORS ORDERBY 2



**ASSIGNMENT – THERE IS SCRIPT FOR ONE TO ONE, ONE TO MANY, MANY TO MANY RELATIONSHIPS.**

**CREATE THOSE TABLES**

**INSERT THAT DATA**

**AND VERIFY THE OUT PUT.**

**ALTER TABLE: - TO ALTER THE STRUCTURE**

CREATE TABLE WORKER

(

WORKID INT,

WORKNAME VARCHAR(20)

)

**ALTER --- ADD COLUMN, REMOVE COLUMN, CHANGE DATATYPE OF A COLUMN**

--ADD COLUMNS

ALTER TABLE WORKER

ADD AGE INT

SELECT \* FROM WORKER

--ALTER DATATYPE OF COLUMN

ALTER TABLE WORKER

ALTER COLUMN AGE TINYINT

--DROP COLUMNS

ALTER TABLE WORKER

DROP COLUMN AGE

--DROP TABLE - IF WE WANT TO DELETE A TABLE WE USE DROP COMMAND

--SYNTAX:

DROP TABLE TABLENAME

DROP TABLE WORKER

**OR WE CAN RIGHT CLICK ON THE TABLE AND CLICK ON DELETE OPTION.**

**NORMALIZATION:**

**THE PROCESS OF APPLYING NORMAL FORMS SEQUENTIALLY ON THE DATABASE DESIGN IS CALLED AS NORMALIZATION.**

1. **FIRST NORMAL FORM**
2. **SECOND NORMAL FORM**
3. **THIRD NORMAL FORM**
4. **BCNF – BOYCE CODD NORMAL FORM**
5. **FOURTH NORMAL FORM**
6. **FIFTH NORMAL FORM**
7. **SIXTH NORMAL FORM**

**WE NEED TO IMPLEMENT FIRST THREE NORMAL FORMS IN ORDER TO MAKE OUR DATABASE AS FULLY NORMALIZED.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CUSTOMER** |  | **PRODUCTS** |  | **ORDERS** |  |
| CUSTNO(PK) |  | PRODiD(PK) |  | ORDERNO | (PK) |
| CUSTNAME |  | DESCRIPTION |  | PRODID |
| ADDRESS |  | UNITPRICE |  | QUANTITY |  |
|  |  | STOCK |  | TOTALPRICE |  |
|  |  |  |  | ORDERDATE |  |
|  |  |  |  | CUSTONO(FK) |  |

1. **FIRST NORMAL FORM:**

**TO SAY THAT A TABLE IS IN FIRST NORMAL FORM, IT SHOULD FOLLOW:**

**NO MULTI VALUE COLUMNS IN A TABLE I.E A COLUMN SHOULD NOT HAVE MULTIPLE VALUES.**

**EX: IT SHOULD NOT BE AS BELOW:**

|  |  |  |
| --- | --- | --- |
| **CUSTNO** | **CUSTNAME** | **ADDRESS** |
| 1 | PAYAL | H.NO,STREETNAME,CITY,STATE |
| 2 | MANOJ | H.NO,STREETNAME,CITY,STATE |
| 3 | SREEKK | H.NO,STREETNAME,CITY,STATE |

**SOLUTION:**

**CUSTOMER TABLE IS NOT IN FIRST NORMAL FORM.**

**IF THE TABLE IS NOT FOLLOWING A NORMAL FORM, WE HAVE TO CUSTOMIZE THE TABLE.**

**AFTER IMPLEMENTING 1ST NF, THE TABLE IS LIKE BELOW:**

|  |
| --- |
| **CUSTOMER** |
| CUSTNO(PK) |
| CUSTNAME |
| H.NO |
| STREETNAME |
| CITY |
| STATE |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CUSTNO** | **CUSTNAME** | **HNO** | **STREETNAME** | **CITY** | **STATE** |
| 1 | PAYAL | HNO | STREETNAME | CITY | STATE |
| 2 | MANOJ | HNO | STREETNAME | CITY | STATE |
| 3 | SREEKK | HNO | STREETNAME | CITY | STATE |

1. **SECOND NORMAL FORM:**

**RULES:**

1. **TABLE SHOULD BE IN 1ST NF.**
2. **NO PARTIAL DEPENDENCY.**

**NO PARTIAL DEPENDENCY: EVERY NON KEY COLUMN SHOULD DEPEND ON COMPLETE PRIMARY KEY COLUMN.**

**NON-KEY : WHICH IS NOT A PRIMARY KEY OR FOREIGN KEY COLUMN.**

**PARTIAL DEPENDENCY: A NON-KEY COLUMN DEPENDS ON PART OF PRIMARY KEY COLUMN IS CALLED PARTIAL DEPENDENCY.**

|  |  |
| --- | --- |
| **ORDERS** |  |
| ORDERNO | (PK) |
| PRODID |
| QUANTITY | NON-KEY |
| TOTALPRICE |
| ORDERDATE |
| CUSTONO(FK) |

**HERE ORDERDATE IS NOT COMPLETELY DEPENDING ON PRIMARY KEY (ORDERNO AND PRODID), HERE WE NEED TO SEPARATE THE TABLE INTO TWO.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **ORDERS** |  | **ORDER\_DETAILS** |  |
|  | ORDERNO |  | ORDERNO | (PK) |
| NON-KEY | ORDERDATE |  | PRODID |
| CUSTNO(FK) |  | QUANTITY | NON-KEY |
|  |  |  | TOTALPRICE |

1. **THIRD NORMAL FORM:**

**RULES:**

1. **IT SHOULD BE IN 2NF.**
2. **NO TRANSITIVE DEPENDENCY**

**TRANSITIVE DEPENDENCY:**

**A NON KEY COLUMN DEPENDS ON OTHER NON-KEY COLUMN.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| EMP\_DEPT |  |  |  |  |  |
| EMPNO(PK) | ENAME | ESAL | DEPTNO | DNAME |  |
| 1 | AA | 50000 | 10 | ACCOUNTS | DUPLICATE DATA |
| 2 | BB | 60000 | 10 | ACCOUNTS |  |
| 3 | CC | 70000 | 20 | HR |  |
| 4 | DD | 80000 | 20 | HR |  |
| 5 | EE | 90000 | 30 | SALES |  |
| 6 | FF | 40000 | 30 | SALES |  |

**HERE DNAME IS DEPENDING ON DEPTNO**

**THIS IS NOT IN 3NF (DNAME AND DEPTNO ARE NON KEY COLUMNS), HERE THIS IS FOLLOWING TRANSITIVE DEPENDENCY – ELIMINATE IT**

**SEPERATE THE DATA INTO TWO TABLES**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| EMP |  |  |  |  | DEPT |  |
| EMPNO(PK) | ENAME | ESAL | DEPTNO(FK) |  | DEPTNO(PK) | DNAME |
| 1 | AA | 50000 | 10 |  | 10 | ACCOUNTS |
| 2 | BB | 60000 | 10 |  | 20 | HR |
| 3 | CC | 70000 | 20 |  | 30 | SALES |
| 4 | DD | 80000 | 20 |  |  |  |
| 5 | EE | 90000 | 30 |  |  |  |
| 6 | FF | 40000 | 30 |  |  |  |

CREATE TABLE dbo.CUSTOMERS(

custId int primary key identity(1,1),

cname varchar(20)NOT NULL,

cage tinyint NULL,

cbalance money NULL,

cmobile varchar(20) NULL,

cbranchid INT FOREIGN KEY REFERENCES BRANCHES(BRANCHID),

),

INSERT [dbo].[CUSTOMERS]([cname], [cage], [cbalance], [cmobile], [cbranchid])VALUES

(N'sreekk', 28, 6000.0000,N'96325698665', 4),

(N'suresh', 29, 7000.0000,N'8325698665', 5),

(N'Ram', 35, 35000.0000,N'9999999999', 3),

(N'Neeraj', 30, 30000.0000,N'8888888888', 7),

(N'sudha', 38, 40000.0000,N'7777777777', 5),

(N'kareem', 36, 37000.0000,N'6666666666', 2),

(N'Rahul', 33, 48000.0000,N'5555555555', 4),

(N'shalini', 22, 7000.0000,N'6666545454', 2),

(N'thushara', 23, 10000.0000,N'9166545454', 4),

(N'Sindhu', 24, 8000.0000,N'8417036266', 1),

(N'Manoj', 29, 75000.0000,N'998989898', 5)

SELECT\*FROM CUSTOMERS

IF MORE THAN ONE COLUMN IS SAID AS PRIMARY KEY THEN IT IS CALLED AS CONDIDATE KEY / COMPOSITE KEY.

CONDIDATE KEY IS THE COLLECITION OF ONE OR MORE COLUMNS USED TO IDENTIFY A ROW.

**DDL, DML, QUERIES.**

**RENAME TABLE:**

SP\_RENAME'OLD TABLE NAME','NEW TABLE NAME'

sp\_rename 'students\_new','students\_latest'

sp\_rename 'students\_latest','students\_new'

SP\_RENAME'TABLENAME.OLD\_COLNAME','NEW COL NAME'

sp\_rename 'students\_new.stname','fullname'

sp\_rename 'students\_new.fullname','Stname'

**SQL LANGUAGE WAS DEVELOPED BY IBM LATER APPROVED BY ANSI.**

**COMMENTS –**

**MULTI LINE COMMENTS /\* \*/**

**OPERATORS:**

**ARTHIMETIC OPERATORS → +,-,\*,/,%**

**LOGICAL OPERATORS →AND , OR, NOT**

**SINGLE VALUE COMPAISON OPERATORS → =, < ,>,,>=, <=, IS**

**MULTI VALUE COMPAISON OPERATORS → IN, ANY, ALL**

**BITWISE OPERTORS→&, |**

**BETWEEN OPERATOR**

**LIKE OPERATOR**

**SET OPERATORS → UNION, UNION ALL, INTERSECT, EXCEPT**

**BUILT IN FUNCTIONS:**

1. **MATHEMATICAL FUNCTIONS**
2. **STRING FUNCTIONS**
3. **DATETIME FUNCTIONS**
4. **AGGREGATE FUNCTIONS**
5. **MATHEMATICAL FUNCTIONS:**

SELECT ABS(-4.38)

SELECT SQUARE(5)

SELECT SQRT(49)

SELECT POWER(3,4)

SELECT PI()

SELECT COS(0)

SELECT SIN(90)

SELECT CEILING(5.34)-- UPPER BOUNDARY VALUE

SELECT FLOOR(5.34)-- LOWER BOUNDARY VALUE

1. **STRING FUNCTIONS:**

SELECT ASCII('B')

SELECT ASCII('Z')

SELECT ASCII('a')

SELECT ASCII('z')

SELECT CHAR(98)

SELECT CHAR(85)

SELECT LEN('I AM A TRAINER AND I TEACH SQL SERVER CLASSES')--length of string

SELECT UPPER('sqlserver')

SELECT LOWER('SQLSERVER')

SELECT LTRIM(' SQLSERVER')

SELECT RTRIM('SQLSERVER ')

SELECT LTRIM(RTRIM(' SQLSERVER '))

SELECT LEFT('SQLSERVER',3)

SELECT RIGHT('SQLSERVER',6)

SELECT SUBSTRING('MICROSOFT SQL SERVER',11,3)

SELECT SUBSTRING('I AM A TRAINER AND I TEACH SQL SERVER CLASSES',22,5)

SELECT REVERSE('SQLSERVER')

SELECT REPLACE('SQLSERVER','E','I')

SELECT REPLACE('SQLSERVER','ER','I')

1. **DATE FUNCTIONS:**

select cast(getdate() as date) todays date only

SELECT CAST(GETDATE() AS DATE) AS TEST

SELECT CAST(GETDATE() AS TIME) AS TEST

SELECT GETDATE()-- todays date and time

SELECT YEAR(GETDATE())

SELECT MONTH(GETDATE())

SELECT DAY(GETDATE())

--DATEPART -- PART OF DATE YEAR, MONTH, DAY

SELECT DATEPART(YYYY,GETDATE())

SELECT DATEPART(MM,GETDATE())

SELECT DATEPART(DD,GETDATE())

--DATENAME -- NAME OF THE DATE

SELECT DATENAME(MM,GETDATE())

SELECT DATENAME(DW,GETDATE())

--DATE ADD - ADD/SUB DAYS/MONTHS/YEARS

SELECT DATEADD(YY,3,GETDATE())

SELECT DATEADD(MM,3,GETDATE())

SELECT DATEADD(DD,3,GETDATE())

SELECT DATEADD(YY,-3,GETDATE())

SELECT DATEADD(MM,-3,GETDATE())

SELECT DATEADD(DD,-3,GETDATE())fJOIN

--DATEDIFF --

SELECT DATEDIFF(YY,'12-05-1986',GETDATE())

SELECT DATEDIFF(MM,'12-05-1986',GETDATE())

SELECT DATEDIFF(DD,'12-05-1986',GETDATE())

1. **AGGREGATE FUNCTIONS:**

**AGGREGATE FUNCTIONS ARE APPLIED ON LIST OF VALUES AND THE OUTPUT WILL BE A SIGNLE ROW OR VALUE.**

SELECT \* FROM CUSTOMERS

SELECT COUNT(\*) FROM CUSTOMERS -- TO KNOW ROW COUNT

SELECT COUNT(1) FROM CUSTOMERS

SELECT \* FROM STUDENTS

SELECT COUNT(STID) FROM STUDENTS --14

SELECT COUNT(STMOBILE) FROM STUDENTS --14

SELECT COUNT(STAGE) FROM STUDENTS --13

**NOTE**: ALL AGGREGATE FUNCTIONS DOESNOT CONSIDER NULL VALUE.

SELECT \* FROM CUSTOMERS

SELECT DISTINCT CAGE FROM CUSTOMERS

SELECT COUNT(DISTINCT CAGE) FROM CUSTOMERS

SELECT \* FROM CUSTOMERS

SELECT SUM(CBALANCE) FROM CUSTOMERS

SELECT AVG(CBALANCE) FROM CUSTOMERS

SELECT MIN(CBALANCE) FROM CUSTOMERS

SELECT MAX(CBALANCE) FROM CUSTOMERS

**NOTE**: SUM, AVG, MIN, MAX ARE THE AGGREGATE FUNCTIONS WHICH CAN BE APPLIED ON ONLY NUMERICAL COLUMNS.

COUNT FUNCTION CAN BE APPLIED ON ANY COLUMN BUT THE OUTPUT IS A NUMBER.

DISTINCT IS A KEY WORD WHICH IS USED TO SHOW THE VALUES FOR ONE TIME.

--BETWEEN OPERATOR

SYNTAX:

COLUMNNAME BETWEEN LOWERVALUE AND UPPERVALUE

--DISPLAY CUSTOMERS WHOSE AGE IN THE RANGE OF 25 TO 35

SELECT \* FROM CUSTOMERS WHERE CAGE>=25 AND CAGE<=35 ORDER BY CAGE

SELECT \* FROM CUSTOMERS WHERE CAGE BETWEEN 25 AND 35 ORDER BY CAGE

--LIKE OPERATOR

**IT CAN BE USED FOR PATTERN MATCHING, IT CAN BE APPLIED ONLY STRING COLUMN.**

--DISPLAY CUSTOMERS WHOSE NAME STARTS WITH S

SELECT \* FROM CUSTOMERS WHERE CNAME LIKE 'S%'

--DISPLAY CUSTOMERS WHOSE NAME STARTS WITH S OR K

SELECT \* FROM CUSTOMERS WHERE CNAME LIKE 'S%' OR CNAME LIKE 'K%'

SELECT \* FROM CUSTOMERS WHERE CNAME LIKE '[SK]%'

--DISPLAY CUSTOMERS WHOSE NAME STARTS WITH S OR K OR R

SELECT \* FROM CUSTOMERS WHERE CNAME LIKE 'S%' OR CNAME LIKE 'K%' OR CNAME LIKE 'R%'

SELECT \* FROM CUSTOMERS WHERE CNAME LIKE '[SKR]%'

--DISPLAY CUSTOMERS WHOSE NAME DOES NOT STARTS WITH S OR K

SELECT \* FROM CUSTOMERS WHERE CNAME NOT LIKE '[SK]%'

SELECT \* FROM CUSTOMERS WHERE CNAME LIKE '[^SK]%'

--DISPLAY CUSTOMERS WHOSE NAME CONATINS H IN THE SECOND POSITION

SELECT \* FROM CUSTOMERS

SELECT\* FROM CUSTOMERS WHERE CNAME LIKE '\_H%'

--DISPLAY CUSTOMERS WHOSE NAME CONATINS A IN THE SECOND POSITION

SELECT \* FROM CUSTOMERS

SELECT\* FROM CUSTOMERS WHERE CNAME LIKE '\_A%'

--DISPLAY CUSTOMERS WHOSE NAME CONATINS H IN THE THIRD POSITION

SELECT\* FROM CUSTOMERS WHERE CNAME LIKE '\_\_H%'

--DISPLAY CUSTOMERS WHOSE NAME STARTS WITH ANY CHARACTER BETWEEN A TO M

SELECT \* FROM CUSTOMERS WHERE cname LIKE '[A-M]%'

SELECT SYNTAX:

SELECT COL1, COL2,… FROM TABLENAME

WHERE

GROUP BY

HAVING – CAN BE USED ONLY WHEN WE USE GROUP BY

ORDER BY

SELECT \* FROM CUSTOMERS

ORDER BY CAGE ASC

SELECT \* FROM CUSTOMERS

ORDER BY CAGE DESC

SELECT \* FROM CUST\_NEW ORDERBY 2

SELECT DISTINCT CNAME FROM CUST\_NEW

SELECT CNAME, CAGE,COUNT(CNAME) FROM CUST\_NEW

GROUPBY CNAME,CAGE

**—IDENTIFY THE DUPLICATE DATA**

**WE IDENTIFY THE DUPLICATES WITH THE HELP OF GROUP BY**

**RULE: THE COLUMNS WHICH ARE USED IN SELECT STATEMENT SHOULD BE PART OF GROUP BY CLAUSE**

**GROUP BY CAN BE USED ALONE**

**HAVING IS DEPENDENT ON GROUP BY – HAVING CANNOT BE USED ALONE**

**RULES:**

**1. THE COLUMNS USED IN SELECT SHOULD BE PART OF GROUP BY CLAUSE.**

**2. IF WE ARE USING AGGREGATE FUNCTION(S) IN SELECT ALONG WITH OTHER COLUMNS THEN IT IS MANADATORY TO HAVE GROUP BY CLAUSE.**

**3. IF THERE ARE ANY AGGREGATE FUNCTIONS IN THE SELECT, THOSE SHOULD NOT BE PART OF GROUP BY CLAUSE.**

**WHERE CLAUSE DOESNT WORK ON AGGREGATE FUNCTIONS. WE HAVE HAVING CLAUSE WHICH IS USED TO FILTER ON AGGREGATE FUNCTIONS ONLY. THIS CAN ONLY BE USED WHEN GROUP BY IS USED.**

**HAVING CANNOT BE USED ALONE**

**WHEN EVER WE USE GROUP BY IF REQUIRED WE CAN USE HAVING CLAUSE.**

**SELECT , ,**

**FROM TABLE**

**WHERE.... – WE CANNOT USE FOR CONDITIONS ON AGGREGATE FUNCTIONS**

**GROUP BY...**

**HAVING ... - CAN BE USED ONLY ON AGGREGATE FUNCTION**

**ORDER BY**

**SELECT ..**

**FROM ..**

**HAVING -- NO THIS DEPENDINT ON GROUP BY**

**WHEN EVER YOU USE GROUP BY YOU CAN USE HAVING OR YOU CANNOT USE HAVING .**

**HAVING - MAKE SURE ALREADY GROUP BY IS THERE..**

**WHERE SAL>5000**

**WHERE COUNT(JOB)>1 --- THIS WRONG SYNTAX - WHERE CANNOT BE APPLIED ON FUNCTIONS**

**CREATE A NEW TABLE CUSTOMERS\_NEW, MAKE SURE THAT THERE ARE FEW DUPLICATE CUSTOMER NAMES IN THE TABLE.**

**APLLY BELOW GROUP BY QUERY:**

SELECT CNAME, COUNT(CNAME) FROM CUSTOMERS\_NEW

GROUP BY CNAME

SELECT \* FROM CUSTOMERS\_NEW ORDER BY CNAME

SELECT DISTINCT CNAME FROM CUSTOMERS\_NEW

SELECT CNAME, COUNT(CNAME) COUNT\_CNAME FROM CUSTOMERS\_NEW

GROUP BY CNAME

HAVING COUNT(CNAME)>1

ORDER BY 2

**RULES:**

**1. THE COLUMNS WHICH WERE IN SELECT LIST SHOULD BE THERE IN GORUPBY**

**HAVING CAN BE USED ONLY WHEN WE USE GROUP BY**

**HAVING CANNOT BE USED ALONE BUT GROUP BY CAN BE USED ALONE**

--QUERY TO SHOW DUPLICATE RECORDS

SELECT CAGE, COUNT(CAGE) AGE\_COUNT FROM CUSTOMERS

GROUPBY CAGE

HAVINGCOUNT(CAGE)>1

**ALIAS:**

**TABLE→ CUSTOMER C / CUSTOMER AS C**

**COLUMN→ CNAME CN/CNAME AS CN**

**JOINS:**

**JOINS ARE USED TO RETRIEVE DATA FROM MULTIPLE TABLE COLUMNS. IN SQL SERVER, WE HAVE BASICALLY 3 TYPES OF JOINS:**

1. **CROSS JOIN**
2. **INNER JOIN**
3. **OUTER JOIN**
4. **CROSS JOIN:**

**CROSS = MULTIPLICATION = CARTESIAN**

**A JOIN WITHOUT CONDITION IS CALLED AS CROSS JOIN.**

**CROSS JOIN PRODUCES RESULT SET WHERE EVERY ROW IN FIRST TABLE JOIN WITH EVERY ROW IN SECOND TABLE.**

**ASSUME 1ST TABLE HAS 5 RECORDS**

**2ND TABLE HAS 3 RECORDS**

**CROSS JOIN = 5\*3=15 RECORDS.**

**SYNTAX:**

**SELECT \* FROM TABLENAME\_1 CROSS JOIN TABLENAME\_2**

**SELECT \* FROM TABLENAME\_1 , TABLENAME\_2**

SELECT \* FROM CUSTOMERS --11

SELECT \* FROM branches --16

SELECT 11\*16

SELECT \* FROM CUSTOMERS CROSS JOIN branches

SELECT \* FROM BRANCHES CROSS JOIN CUSTOMERS

SELECT \*FROM CUSTOMERS , branches

1. **INNER JOIN:**

**SYNTAX:**

**SELECT \* OR (COLUMNS) FROM**

**TABLE1**

**<JOINTYPE>**

**TABLE 2**

**ON <CONDTION>**

**<JOINTYPE>**

**TABLE 3**

**ON <CONDTION>**

**---**

**WHERE**

**GROUP BY**

**HAVING**

**ORDERBY**

**RULES OF JOINS:**

1. **TABLES SHOULD HAVE RELATED (COMMON) COLUMN .**
2. **COMMON COLUMN DATATYPES SHOULD BE SAME.**
3. **JOINING TABLES PRODUCES TEMPORARY RESULT SET WITH COMBINED STRUCTURE OF BOTH TABLES.**

**INNER JOIN IS DEFAULT TYPE OF JOIN. INNER JOIN PRODUES RESULT SET WHICH CONTAINS ONLY MATCHED RECORDS.**

**WE GET ONLY MATCHED DATA ACCORDING TO THE JOIN CONDITION FROM THE TABLES WHICH INVOLVED IN INNER JOIN.**

(WE CAN GET ALL TABLE BY EXECUTING THIS QUERY

SELECT \*  
FROM sys.Tables

select table\_name,column\_name from information\_schema.columns  
where column\_name like '%esalary%')

SELECT C.custID,C.cname,B.bname FROM CUSTOMERS C

INNER JOIN

BRANCHES B

ON

C.Cbranchid=B.branchid

--DISPLAY CUSTID, CNAME, CAGE, BNAME FROM CUSTOMERS AND BRANCHES TABLE

SELECT C.CUSTID, C.CNAME, C.CAGE, B.BNAME FROM CUSTOMERS C

JOIN

branches B

ON

C.branchid=B.branchId

--DISPLAY DATA WHERE MANAGERS ARE ACTING AS CUSTOMERS

SELECT \* FROM CUSTOMERS C --11

INNER JOIN

BRANCHES B --16

ON

C.cname=B.bmgrname

SELECT DPC.EnglishProductCategoryName,DPSC.EnglishProductSubcategoryName,

DP.EnglishProductName, FS.SalesAmount

FROM DimProductCategory DPC

JOIN DimProductSubcategory DPSC

ON DPSC.ProductCategoryKey=DPC.ProductCategoryKey

JOIN DimProduct DP

ON DP.ProductSubcategoryKey=DPSC.ProductSubcategoryKey

INNER JOIN FactInternetSales FS

ON FS.ProductKey=DP.ProductKey

ORDER BY 1

**ASSIGNMENT IS BELOW (DO AS I DID IN THE CLASS)**

SELECT \* FROM authors

SELECT \* FROM BOOKS

SELECT \* FROM BOOKS\_AUTHORS

CREATE TABLE BOOKS\_SALES

(

BOOKID INT FOREIGN KEY REFERENCES BOOKS(BOOKID) UNIQUE,

PRICE MONEY

)

INSERT INTO BOOKS\_SALES VALUES

(1, 500),

(2, 450),

(3, 600),

(4, 400)

SELECT \* FROM BOOKS\_SALES

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

--SALES BASED ON BOOKS

SELECT B.BOOKid,B.BOOKTITLE, SUM(BS.PRICE) SALES FROM BOOKS B INNER JOIN BOOKS\_SALES BS

ON B.BOOKid=BS.BOOKID

GROUP BY B.BOOKid,B.BOOKTITLE

--SALES BASED ON BOOKS FOR AUTHORS\_BOOKS

SELECT B.BOOKid,B.BOOKTITLE,SUM(BS.PRICE) SALES FROM BOOKS B

INNER JOIN BOOKS\_AUTHORS BA ON BA.BOOKID=B.BOOKid

INNER JOIN BOOKS\_SALES BS ON BS.BOOKID=B.BOOKID

GROUP BY B.BOOKid,B.BOOKTITLE

ORDER BY B.BOOKTITLE

--SALES BASED ON AUTHORS FOR AUTHORS\_BOOKS

SELECT A.authorid,A.authname, SUM(BS.PRICE) SALES FROM authors A

INNER JOIN BOOKS\_AUTHORS BA ON A.authorid=BA.AUTHORID

INNER JOIN BOOKS\_SALES BS ON BA.BOOKID=BS.BOOKID

GROUP BY A.authorid,A.authname

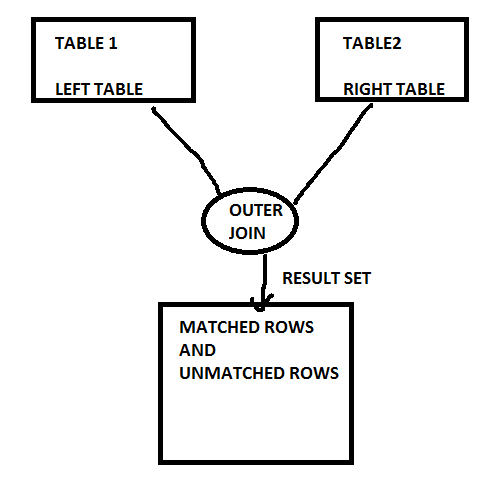
ORDER BY A.authname

1. **OUTER JOIN:**

**OUTER JOIN PRODUCES RESULT SET WHICH CONTAIN MATCHED ROWS AND UNMATCHED ROWS.**

**WE HAVE 3 TYPES OF OUTER JOINS:**

1. **LEFT OUTER JOIN**
2. **RIGHT OUTER JOIN**
3. **FULL OUTER JOIN**

****

1. **LEFT OUTER JOIN:**

**RESULT SET CONTAINS MATCHED ROWS AND UNMATCHED ROWS FROM LEFT TABLE.**

**FOR UNMATCHED ROWS - RIGHT TABLE DATA WILL BE FILLED WITH NULLS.**

**TOTAL DATA FROM LEFT TABLE + MATCHED DATA FROM RIGHT TABLE**

**+ FOR NOT MATCHING IN RIGHT TABLE – FILLED WITH NULLS.**

SELECT \* FROM CUSTOMERS C

LEFT OUTER JOIN

branches B

ON

C.Cbranchid=B .BRANCHID

--DISPLAY CUSTID, CNAME, CAGE, BNAME FROM CUSTOMERS AND BRANCHES TABLE

SELECT C.CUSTID, C.CNAME, C.CAGE, B.BNAME FROM

CUSTOMERS C LEFTJOIN branches B ON C.Cbranchid=B.branchId

SELECT \* FROM BRANCHES B LEFT JOIN CUSTOMERS C ON C.branchid=B.branchid

SELECT \* FROM CUSTOMERS C LEFT JOIN BRANCHES B ON C.cname=B.bmgrname

SELECT \* FROM BRANCHES B LEFT JOIN CUSTOMERS C ON C.cname=B.bmgrname

1. **RIGHT OUTER JOIN:**

**RESULT SET CONTAINS MATCHED ROWS AND UNMATCHED ROWS FROM RIGHT TABLE.**

**FOR UNMATCHED ROWS - LEFT TABLE DATA WILL BE FILLED WITH NULLS.**

**TOTAL DATA FROM RIGHT TABLE + MATCHED DATA FROM LEFT TABLE**

**+ FOR NOT MATCHING IN LEFT TABLE – FILLED WITH NULLS.**

SELECT\* FROM CUSTOMERS C

RIGHT OUTER JOIN

branches B

ON

C.Cbranchid=B .BRANCHID

SELECT C.CUSTID, C.CNAME, C.CAGE, B.BNAME

FROM CUSTOMERS C RIGHT JOIN branches B

ON

C.Cbranchid=B.branchId

SELECT \* FROM CUSTOMERS C RIGHT OUTER JOIN branches B

ON

C.cname=B.bmgrName

SELECT \* FROM branches B RIGHT OUTER JOIN CUSTOMERS C

ON

C.cname=B.bmgrName

1. **FULL OUTER JOIN:**

**RESULT SET CONTAINS MATCHED ROWS AND UNMATCHED ROWS FROM LEFT TABLE AND RIGHT TABLE.**

**FOR UNMATCHED ROWS - LEFT TABLE AND RIGHT TABLE DATA WILL BE FILLED WITH NULLS.**

**TOTAL DATA FROM RIGHT TABLE +TOTAL DATA FROM LEFT TABLE + MATCHING DATA + UNMATCHED DATA IS FILLED WITH NULLS.**

SELECT \* FROM CUSTOMERS C

FULL OUTER JOIN

branches B

ON

C.Cbranchid=B .BRANCHID

SELECT C.CUSTID, C.CNAME, C.CAGE, B.BNAME

FROM CUSTOMERS C FULL JOIN branches B

ON

C.Cbranchid=B.branchId

SELECT\*FROM CUSTOMERS C FULL OUTER JOIN branches B

ON

C.cname=B.bmgrName

SELECT\*FROM branches B FULL OUTER JOIN CUSTOMERS C

**DISPLAY ALL THE EMPNO’S, ENAMES AND CORRESPONDING DEPTNAMES, DLOCATIONS.**

**DISPLAY EMPNO’S , EMPNAME AND ALL DNAMES, DLOCATIONS.**

**DISPLAY ALL EMPNO’S, EMPNAME AND ALL DNAMES, DLOCATIONS.**

**EQUI JOIN → IF JOIN CONDITION HAS = SIGN.**

**NON EQUI JOIN→ IF JOIN CONDITION IS HAVING OTHER THAN = SIGN THEN IT IS NON EQUI JOIN.**

**NATURAL JOIN→ WHILE JOINING TABLES, ONCE WE ELIMINATE DUPLICATE COLUMNS IN THE OUTPUT IT CAN BE CALLED AS NATURAL JOIN.**

**SELF JOIN→ THE CONCEPT OF JOINING A TABLE WITH ITSELF IS CALLED AS SELF JOIN.**

**SUB QUERIES:**

**QUERY INSIDE THE QUERY.**

**GENERALLY A SELECT STATEMENT NESTED WITH OTHER SELECT STATEMENT IS CALLED SUBQUERY.**

**SELECT \* FROM TABLENAME WHERE COLUMNNAME IN (SELECT COLUMN FROM TABLE2)**

**OUTER QUERY --INNER QUERY**

**WE HAVE TWO TYPES OF SUBQUERIES:**

**1. NESTED SUBQUERY**

**2.CORELATED SUBQUERY**

**OPERATIONS:**

|  |  |
| --- | --- |
| **INNER QUERY** | **OPERATOR** |
| SINGLE SCALAR VALUE | SINGLE VALUE COMPARISON OPERATOR  =, >, <, <= >=, IS |
| LIST OF VALUES | LIST OPERATOR (IN, ANY, ALL) |

**1. NESTED SUB-QUERY:**

**IN A SUBQUERY, OUTER QUERY DEPENDS ON RESULT OF INNER QUERY IS CALLED NESTED SUB-QUERY**

**EXECUTION PROCESS:**

**1ST INNER QUERY IS EXECUTED THEN BASED ON OUTPUT OF INNER QUERY, OUTER QUERY WILL EXECUTE.**

**DISPLAY CUSTOMERINFO WHO ARE HAVING 1ST HIGHEST CUSTOMER BALANCE.**

SELECT MAX(CBALANCE) FROM CUSTOMERS

SELECT \* FROM CUSTOMERS WHERE cbalance=75000

SELECT \* FROM CUSTOMERS WHERE cbalance=(SELECT MAX(CBALANCE) FROM CUSTOMERS)

**DISPLAY CUSTID, CNAME WHO ARE HAVING BALANCE MORE THAN AVG BALANCE**

SELECT AVG(cbalance) FROM CUSTOMERS

SELECT \* FROM CUSTOMERS WHERE cbalance>27545.4545

SELECT \* FROM CUSTOMERS

WHERE cbalance>(SELECT AVG(cbalance) FROM CUSTOMERS)

**CREATE TABLE DEPARTMENT AS DISCUSSED**

**CREATE TABLE EMPLOYEE AS DISCUSSED (FK WITH DEPID)**

**DISPLAY EMPINFO WHO ARE WORKING ACCOUNTS DEPART.**

**LIST OPERATORS:**

**THESE ARE USED TO PERFORM COMPARISON WITH THE LIST OF VALUE.**

1. **IN → DIRECTLY USED**
2. **ANY → USED IN COMBINATION WITH SINGLE VALUE COMPARISON OPERATORS.**
3. **ALL → USED IN COMBINATION WITH SINGLE VALUE COMPARISON OPERATORS.**

**ANY(8000,12000,15000)**

* **8000 OR 12000 OR 15000**

**ALL (8000,12000,15000)**

* **8000 AND 12000 AND 15000**

**DISPLAY CUSTOMER INFO WHO ARE PART OF ‘Banashanakari’ BRANCH**

**‘RTNagar’.**

select \* from BRANCHES where bname='Banashanakari' or bname='RTNagar'

select \* from CUSTOMERS where

branchid in (select branchid from BRANCHES where bname='Banashanakari' or bname='RTNagar')

**OR CAN BE ACHIVED AS BELOW**

select \* from CUSTOMERS where

branchid in (select branchid from BRANCHES where bname in ('Banashanakari','RTNagar'))

**OR CAN BE ACHIVED AS BELOW**

select \* from CUSTOMERS where

branchid =ANY (select branchid from BRANCHES where bname in ('Banashanakari','RTNagar'))

**DISPLAY EMPINFO WHO ARE GETTING SALARY MORE THAN ATLEAST ONE SALARY OF DEPTNO 4**

**DISPLAY EMPINFO WHO ARE GETTING SALARY LESS THAN ALL SALARIES OF DEPT 2**

**2. CORRELATED SUBQUERY.**

**IN A SUBQUERY , IF INNER QUERY DEPENDS ON OUTER QUERY THEN IT IS CALLED AS CORRELATED SUBQUERY.**

**EXECUTION PROCESS:**

**OUTER QUERY EXECUTES 1ST AND THE RESULT OF OUTER QUERY IS PASSED TO INNER QUERY, BASED ON OUTER QUERY RESULT INNER QUERY WILL BE EXECUTED.**

**DISPLAY TOP 2 SALARIES LIST FROM EMP TABLE**

**DISPLAY EMPINFO WHO ARE GETTING 2ND HIGHEST SALARY**

**DISPLAY EMPINFO WHO ARE GETTING 66TH HIGHEST SALARY**

CREATE TABLE DEPARTMENT

(DEPTID INT PRIMARY KEY IDENTITY(1,1),

DNAME VARCHAR(20)

)

INSERTINTO DEPARTMENT VALUES

('ACCOUNTS'),

('ADMIN'),

('SECURITY'),

('HR'),

('SALES AND MARKETTING'),

('TELLERS')

CREATE TABLE EMPLOYEE

(

EID INT PRIMARY KEY IDENTITY(1,1),

ENAME VARCHAR(20),

ESALARY MONEY,

DID INT FOREIGN KEY REFERENCES DEPARTMENT(DEPTID)

)

INSERTINTO EMPLOYEE VALUES

('SREEKANTH',30000,1),

('PAYAL',40000,1),

('MANOJ',50000,1),

('SINDHU',35000,2),

('HARI',45000,2),

('VIVEK',55000,2),

('SWETHA',33000,3),

('KRISHNA',43000,3),

('PAVANI',53000,3),

('ASHOK',34000,4),

('KIRAN',44000,4),

('RAHU',54000,4),

('VINAY',36000,5),

('RAJ',46000,5),

('RANJITH',56000,5)

SELECT\*FROM DEPARTMENT

SELECT\*FROM EMPLOYEE

**BLIND RULE TO GET HIGHEST SALARIES:**

**SELECT \* FROM TABLENAME T1 WHERE (N-1)=**

**( SELECT COUNT(DISTINCT COL ) FROM TABLENAME T2**

**WHERE T2.COL>T1.COL )**

**DISPLAY EMPLOYEES WHO ARE GETTING 2ND HIGHEST SALARY.**

SELECT\*FROM EMPLOYEE E1 WHERE 1=

(SELECTCOUNT(DISTINCT ESALARY)FROM EMPLOYEE E2 WHERE E2.ESALARY>E1.ESALARY)

--10TH HIGHEST SALARY (10-1)

SELECT\*FROM EMPLOYEE E1 WHERE 9=

(SELECTCOUNT(DISTINCT ESALARY)FROM EMPLOYEE E2 WHERE E2.ESALARY>E1.ESALARY)

**BLIND RULE TO GET LEAST SALARIES:**

**SELECT \* FROM EMPLOYEE E1 WHERE (N-1) =**

**(SELECT COUNT(DISTINCT E2.ESALARY) FROM EMPLOYEE E2**

**WHERE E2.SALARY<E1.SALARY)**

**DISPLAY EMPLOYEES WHO ARE GETTING 2ND LEAST SALARY.**

--2ND LEAST SALARY

SELECT\*FROM EMPLOYEE E1 WHERE 1=

(SELECTCOUNT(DISTINCT ESALARY)FROM EMPLOYEE E2 WHERE E2.ESALARY<E1.ESALARY)

--5TH LEAST SALARY

SELECT\*FROM EMPLOYEE E1 WHERE 4=

(SELECTCOUNT(DISTINCT ESALARY)FROM EMPLOYEE E2 WHERE E2.ESALARY<E1.ESALARY)

**INLINE QUERY:**

**ONCE WE SUBMIT THE SQL STATEMENT DIRECTLY TO THE DATABASE SERVER FROM THE CLIENT THAT WE CALLED AS INLINE QUERIES.**

**DRAWBACKS OF INLINE QUERIES:**

1. **INLINE QUERIES CAN PERFORM SINGLE OPERATION AT A TIME.**
2. **INLINE QUERIES ARE TO BE COMPLIED EVERYTIME WHEN THEY ARE EXECUTED WHICH IS TIME TAKING.**
3. **LESS PERFORMANCE.**

**TO OVERCOME THIS SQL SERVER SUPPORTS OBJECTS LIKE:**

1. **VIEWS**
2. **STORED PROCEDURES**
3. **USER DEFINED FUNCTIONS**
4. **VIEWS:**
5. **VIEWS ARE NOTHING BUT A LOGICAL REPRESENTATION BASED ON ONE OR MORE TABLES.**
6. **VIEW CONTAINS ONLY SELECT STATEMENT (IT STORES SELECT QUERY).**
7. **VIEW IS ALSO CALLED AS VIRTUAL TABLE.**
8. **VIEW DOESN’T CONTAIN DATA IN IT.**

**SNYTAX:**

**CREATE VIEW VIEWNAME**

**[WITH ENCRYPTION]**

**AS**

**SELECT STATEMENT..**

**HERE IF, WE USE ‘WITH ENCRYPTION’ WE CAN’T ALTER THE VIEW.**

**WITH ENCYPTION IS ALWAYS OPTIONAL AND IS NOT PREFERRED TO USE.**

**VIEW CONTAINS SELECT STATEMENT WHICH REFERS SIMPLE SELECT STATEMENT OR JOINS, AGGREGATE FUNCTIONS , SUB QUERIES…**

**VIEW CANNOT REFER TEMPARORY TABLES.**

**WE CAN UPDATE A VIEW ALSO.**

**CREATE A VIEW TO WORK WITH EMPNO, EMPNAME OF EMP TABLE.**

CREATE VIEW VW\_EMPLOYEE

AS

SELECT EID,ENAME FROM EMPLOYEE

SELECT \* FROM VW\_EMPLOYEE

**IF THE DATA IS MODIFIED IN TABLE THAT WILL REFLECT IN THE VIEW.**

**ALTER VIEW VW\_EMPLOYEE TO ADD A COLUMN ESALARY TO IT.**

ALTER VIEW VW\_EMPLOYEE

AS

SELECT EID,ENAME,ESALARY FROM EMPLOYEE

SELECT\*FROM VW\_EMPLOYEE

**CREATE A VIEW TO SELECT ALL COLUMNS FROM CUSTOMERS**

CREATE VIEW VW\_CUSTOMERS

AS

SELECT \* FROM CUSTOMERS

SELECT \* FROM VW\_CUSTOMERS

**DELETE OR DROP VIEW:**

**DROP THE VIEW VW\_CUSTOMERS**

**GOT O OBJECT EXPLORER→ DATABASES→ SELECT OUR DB→ VIEWS→ RIGHT CLICK ON A VIEW → CLICK ON DELETE.**

**OR**

DROP VIEW VW\_CUSTOMERS

**CREATE A VIEW FOR BRANCHES USING WITH ENCRYPTION.**

**LATER TRY TO ALTER IT.**

CREATE VIEW VW\_BRANCHES

WITH ENCRYPTION

AS

SELECT\*FROM branches

SELECT\*FROM VW\_BRANCHES

**ALTER VIEW VW\_BRANCHES**

ALTERVIEW VW\_BRANCHES

WITHENCRYPTION

AS

SELECT\*,GETDATE() DATEVAL FROM branches

SELECT\*FROM VW\_BRANCHES

**NOTE: FOR THE NORMAL VIEWS WE CAN SEE THE SCRIPT WHERE FOR THE ENCRYPTED VIEW WE CANNOT SEE THE SCRIPT AS IT IS ENCRYPTED.**

**--CREATE A VIEW BASED ON EMP AND DEPT TABLES AND SELECT EMPID, ENAME, ESALARY, DEPTID, DNAME**

**--AND GET ONLY MATCHED DATA BASED ON DEPTID**

SELECT\*FROM DEPARTMENT

SELECT\*FROM EMPLOYEES

SELECT EMPID,ENAME,ESALARY, DEPTID, DEPTNAME FROM DEPARTMENT D

INNERJOIN

EMPLOYEES E

ON

E.EDID=D.DEPTID

CREATEVIEW VW\_EMP\_DEPT

AS

SELECT EMPID,ENAME,ESALARY,DEPTID,DEPTNAME FROM DEPARTMENT D

INNERJOIN

EMPLOYEES E

ON

E.EDID=D.DEPTID

SELECT\*FROM VW\_EMP\_DEPT

-- CREATE A VIEW TO SEE HIGHEST SALARY EMPLOYEE INFORMATION

SELECTMAX(ESALARY)FROM EMPLOYEES

SELECT\*FROM EMPLOYEESWHERE ESALARY=18000

SELECT\*FROM EMPLOYEESWHERE ESALARY=(SELECTMAX(ESALARY)FROM EMPLOYEE)

CREATEVIEW CW\_MAXSALARY

AS

SELECT\*FROM EMPLOYEE WHERE ESALARY=(SELECTMAX(ESALARY)FROM EMPLOYEE)

SELECT\*FROM CW\_MAXSALARY

**T-SQL PROGRAMMING CONSTRUCTS:**

**VARIABLE: ITS USED TO STORE DATA TEMPORARILY.**

**VARIABLE SAYS THE DATA IN ME VARIES.**

**2 TYPES OF VARIABLES:**

1. **GLOBAL VARIABLES**
2. **LOCAL VARIABLES**
3. **GLOBAL VARIABLES:**

**THESE ARE PREDEFINED VARIABLES CREATED FOR EACH CONNECTION TO STORE SYSTEM LEVEL INFORMATION. THESE ARE READ ONLY.**

**GLOBAL VARIABLES ARE ALSO CALLED AS SYSTEM FUNCTIONS.**

**EX:** SELECT@@ERROR **-- IT STORES ERROR INFORMATION OF RECENT T-SQL STATEMENT.**

SELECT@@VERSION **– IT STORES THE CURRENT SQL SERVER VERSION.**

SELECT@@SERVERNAME **– IT STORES THE SYSTEMNAME WHERE SQL SERVER IS INSTALLED.**

SELECT@@SERVICENAME **– THIS IS NOTHING BUT SQL SERVER SERVICE.**

1. **LOCAL VARIABLES:**

**THESE ARE USER DEFINED VARIABLES WHICH CAN BE READ, WRITE.**

**SYNTAX TO DECLARE:**

**DECLARE @VARIABLENAME DATATYPE**

**SET @VARABLENAME=VALUE**

**PRINT @VARIABLENAME**

**PRINT ‘MESSAGE’ + @VARIABLENAME**

**HERE + IS USED FOR CONCATENATION.**

**EX:**

DECLARE @X VARCHAR(20)-- DECLARATION

SET @X='I AM A TRAINER'-- ASSIGNATION

PRINT @X -- OUTPUT

PRINT'VARIABLE VALUE IS - '+ @X

DECLARE @EMPNO INT, @ENAME VARCHAR(20)

SET @EMPNO=1001

SET @ENAME='SREEKANTH'

PRINT @EMPNO

PRINT @ENAME

**EX:**

DECLARE @EMPNO INT, @ENAME VARCHAR(20), @ESALARY MONEY

SELECT @EMPNO=1001,@ENAME='SREEKANTH',@ESALARY=85000

SELECT @EMPNO EID ,@ENAME ENAME ,@ESALARY ESAL

**SET STATEMENT CAN BE USED TO ASSIGN ONE VARIABLE AT A TIME BUT SELECT STATEMENT CAN BE USED TO ASSIGN MULTIPLE VARIABLES AS WELL AS READ THE DATA FROM MULTIPLE VARIABLES.**

**OUTPUT OF SET WILL BE IN FORM OF LIST (OF VALUES)**

**OUTPUT OF SELECT WILL BE IN FORM OF TABLE.**

**LOOPING:**

**IN T-SQL WE HAVE ONLY ONE LOOPING STATEMENT I.E WHILE**

**SYNTAX:**

**WHILE (CONDITION)**

**BEGIN**

**----**

**----**

**END**

**CONDITIONAL STATEMENTS:**

1. **IF … ELSE**
2. **CASE STATEMENT**

**IF..ELSE:**

**SYNTAX:**

**IF(CONDITION)**

**BEGIN**

**----**

**----**

**END**

**ELSE**

**BEGIN**

**----**

**----**

**END**

**WRITE A T-SQL SCRIPT TO INSERT EVEN NUMBERS BETWEEN 1 TO 100 IN A TABLE .**

CREATE TABLE EVENNOS

(

EVENNO INT

)

SELECT\*FROM EVENNOS

**--PROGRAM**

DECLARE @i INT

SET @i=1

WHILE(@i<=100)-- THIS WILL LOOP TILL 1 TO 100

BEGIN

IF(@i%2=0)-- VALIDATING FOR EVEN NUMBER

BEGIN

INSERT INTO EVENNOS VALUES(@i)

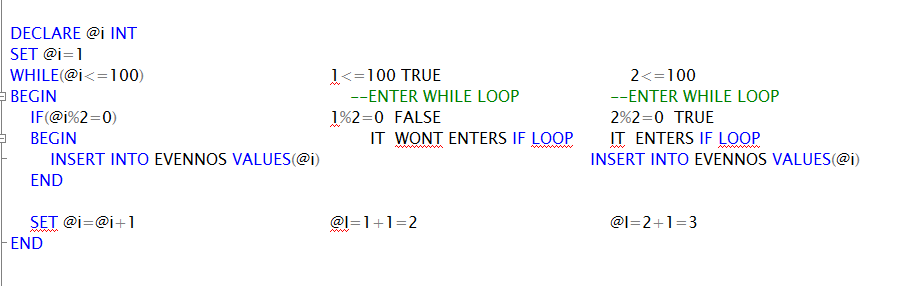
END

SET @i=@i+1 --THIS VALUE IS INCREMENTED TILL THE WHILE CONDITION IS MET

END

SELECT\*FROM EVENNOS

**EXECUTION PROCESS:**



DECLARE @X INT

SET @X=2

WHILE(@X<=100)

BEGIN

IF(@X%2=0)

BEGIN

INSERT INTO EVENNOS VALUES(@X)

END

SET @X=@X+2

END

DECLARE @X INT

SET @X=2

WHILE(@X<=100)

BEGIN

INSERT INTO EVENNOS VALUES(@X)

SET @X=@X+2

END

**SELECT.. CASE:**

**SYNTAX:**

**SELECT COL1, COL2, CASE EXPRESSION**

**WHEN VALUE1 THEN [STMT]**

**WHEN VALUE2 THEN [STMT]**

**---**

**---**

**ELSE [STMT]**

**END**

**FROM TABLENAME**

**WHERE <CONDITION>**

DECLARE @i INT

SET @i=3

SELECT CASE @i

WHEN 1 THEN'FIRST MESSGAE'

WHEN 2 THEN'SECOND MESSGAE'

WHEN 3 THEN'THIRD MESSAGE'

ELSE'OTHER OUTPUT'

END VARIABLEVALUE

CREATE TABLE EMP\_GENDER

(

ENAME VARCHAR(20),

GENDER CHAR(1)

)

INSERTINTO EMP\_GENDER VALUES

('SREEKANTH','M'),

('KISHORE','M'),

('SINDHU','F'),

('MANASA','F'),

('SURI','M')

SELECT\*FROM EMP\_GENDER

--WHERE EVER M IS THERE SHOW AS MALE AND F IS THERE SHOW AS FEMALE

SELECT ENAME,CASE GENDER

WHEN'M'THEN'MALE'

WHEN'F'THEN'FEMALE'

ELSE'OTHERS'

END GENDER

FROM EMP\_GENDER

SELECT ENAME,CASE GENDER

WHEN'M'THEN'MALE'

ELSE'FEMALE'

END GENDER

FROM EMP\_GENDER

**TYPE CONVERSION:**

**THE PROCESS OF CONVERTING FROM ONE TYPE OF DATA TO OTHER TYPE OF DATA.**

**CHANGING THE DATA TYPE OF VALUE.**

1. **CAST(EXPRESSION AS TARGET TYPE)**
2. **CONVERT(TARGET, EXPRESSION, STYLE)**

**-- STYLE IS OPTIONAL PARAMETER**

**STYLE – IT IS USED TO WORK WITH DIFFERENT DATETIME FORMATS.**

**0-21 – YY**

**100-135 – YYYY**

DECLARE @P INT

SET @P=45

PRINT'P VALUE USING CAST IS - '+CAST(@P ASCHAR)

PRINT'P VALUE USING CONVERT IS - '+CONVERT(CHAR(2),@P)

--DISPLAY ONLY DATE OF CURRENT DATETIME

SELECTGETDATE()

SELECTCONVERT(CHAR(10),GETDATE(), 101)

SELECTCONVERT(CHAR(10),GETDATE(), 102)

SELECTCONVERT(CHAR(10),GETDATE(), 103)

SELECTCONVERT(CHAR(10),GETDATE(), 104)

SELECTCONVERT(CHAR(10),GETDATE(), 105)

SELECTCONVERT(CHAR(12),GETDATE(), 106)

SELECTCONVERT(CHAR(14),GETDATE(), 107)

--DISPLAY ONLY TIME OF CURRENT DATETIME

SELECTCONVERT(CHAR(8),GETDATE(), 108)--21:56:39

**STORED PROCEDURES:**

**DEF: STORED PROCEDURE IS NOTHING BUT SET OF PRE-COMPILED T-SQL STATEMENTS STORED IN DATABASE SERVER.**

**SYNTAX:**

**CREATE PROCEDURE PROCEDURE\_NAME(PARAMS)**

**AS**

**BEGIN**

**----**

**----**

**END**

--CREATE A STORE PROCEDURE WHICH RETURNS EMPNO, ENAME, DID AND DEPTNAME FOR ALL EMPLOYEES

SELECT \* FROM DEPARTMENT

SELECT \* FROM EMPLOYEES

SELECT E.EID,E.Ename,D.DEPTID,D.DNAME FROM EMPLOYEES E

INNER JOIN DEPARTMENT D ON E.EID=D.DEPTID

CREATE PROCEDURE USP\_EMPDATA

AS

BEGIN SELECT E.EID,E.Ename,D.DEPTID,D.DNAME FROM EMPLOYEES E

INNER JOIN DEPARTMENT D ON E.EID=D.DEPTID

END

EXEC USP\_EMPDATA

USP\_EMPDATA

**PASSING PARAMETERS:**

**SP’S CAN BE CREATED WITH MAXIMUM 1024 INPUT PARAMETERS AND WITH MAXIMUM 1024 OUTPUT PARAMETERS.**

**SYNTAX:**

**PARAMETERNAME TYPE(SIZE)**

**PARAMETERNAME1 TYPE(SIZE) OUTPUT**

--CREATE A PROCEDURE WHICH CALCUALTES AVG OF GIVEN TWO NUMBERS

10+20

SELECT (10+20)/2

DECLARE @X INT, @Y INT, @Z INT

SET @X=10

SET @Y=20

SET @Z=(@X+@Y)/2

PRINT'AVG VALUE IS - '+CAST(@Z AS CHAR)

GO

CREATE PROCEDURE USP\_CALCAVG

(@X INT, @Y INT)

AS

BEGIN

DECLARE @Z INT

SET @Z=(@X+@Y)/2

PRINT'AVG VALUE IS - '+CAST(@Z AS CHAR)

END

EXEC USP\_CALCAVG205,445

USP\_CALCAVG 205,445

**DIRECTION OF PARAMETERS:**

**BY DEFAULT IT IS INPUT DIRECTION. IF WE WANT TO SHOW THE VARIABLE VALUE AS OUTPUT, WE NEED TO SET IT AS OUTPUT DIRECTION.**

CREATE TABLE USERS

(

USERNAME VARCHAR(25) PRIMARY KEY,

PASSWORD CHAR(8)

)

INSERT INTO USERS VALUES

('SREEKANTH','SREE123'),

('MANOJ','MANOJ123'),

('PAYAL','PAYAL123'),

('SURI','S123'),

('DATTA','D123'),

('KISHORE','K123')

SELECT\*FROM USERS

-- CREATE A PROCEDURE WHICH RETURNS PASSWORD AND LENGTH OF PASSWORD FOR GIVEN USERNAME

SELECT\*FROM USERS

SELECT \* FROM USERS WHERE USERNAME='SREEKANTH'

SELECT PASSWORD,LEN(PASSWORD) LEN\_PWD FROM USERS WHERE USERNAME='KISHORE'

DECLARE @USERNAME VARCHAR(25)

DECLARE @PASSWORD VARCHAR(20)

DECLARE @LENGTH\_PWD INT

SET @USERNAME='KISHORE'

--THE BELOW USED TO ASSIGN DATA TO VARIABLES

SELECT @PASSWORD=[PASSWORD],

@LENGTH\_PWD=LEN([PASSWORD])

FROM USERS

WHERE USERNAME=@USERNAME

--TO SHOW OUPUT

SELECT @PASSWORD, @LENGTH\_PWD

GO

CREATE PROCEDURE USP\_GETPWD

(

@USERNAME VARCHAR(25),

@PASSWORD VARCHAR(20) OUTPUT,

@LENGTH\_PWD INT OUTPUT

)

AS

BEGIN

--THE BELOW USED TO ASSIGN DATA TO VARIABLES

SELECT @PASSWORD=PASSWORD,

@LENGTH\_PWD=LEN(PASSWORD)

FROM USERS

WHERE USERNAME=@USERNAME

--TO SHOW OUPUT

SELECT @PASSWORD PASSWORD, @LENGTH\_PWD LENGTH\_PWD

END

--EXECUTE

DECLARE @PWD VARCHAR(10), @LEN\_PWD INT

EXEC USP\_GETPWD'MANOJ', @PWD OUTPUT, @LEN\_PWD OUTPUT

**ASSIGNMENT --CREATE A PROCEDURE WHICH RETURNS NO.OF ROWS IN USERS TABLE.**

-- CREATE A PROCEDURE WHICH WILL VERIFY GIVEN USERNAME AND PASSWORD VALID OR NOT

--IF VALID PRINT SUCCESSFULL ELSE LOGIN FAILED.

--INPUT - USERNAME AND PASSWORD

--OUPUT DISPLAYED - LOGIN SUCCESSFUL OR LOGIN FAILED

SELECT \* FROM USERS

DECLARE @UNAME VARCHAR(20), @PWD VARCHAR(20)

SET @UNAME='SURI'

SET @PWD='S123'

IF((SELECT COUNT(\*) FROM USERS WHERE USERNAME=@UNAME AND PASSWORD=@PWD)>0)

BEGIN

PRINT 'LOGIN IS SUCCESSFUL'

END

ELSE

BEGIN

PRINT 'LOGIN IS FAILED - USERNAME/PASSWORD IS WRONG'

END

GO

CREATE PROCEDURE USP\_VALIDATE\_LOGIN\_CREDENTIALS

(

@UNAME VARCHAR(20),

@PWD VARCHAR(20)

)

AS

BEGIN

IF((SELECT COUNT(\*) FROM USERS WHERE USERNAME=@UNAME AND PASSWORD=@PWD)>0)

BEGIN

SELECT 'LOGIN IS SUCCESSFUL' LOGIN\_STATUS

END

ELSE

BEGIN

SELECT 'LOGIN IS FAILED - USERNAME/PASSWORD IS WRONG' LOGIN\_STATUS

END

END

EXEC USP\_VALIDATE\_LOGIN\_CREDENTIALS 'SURI','S123'

**--CREATE A PROCEDURE WHICH RETURNS EMPNAME,ESALARY,EAGE,DNAME FROM EMP AND DEPT TABLES FOR THE GIVEN EMPID**

**--INPUT - EMPID**

**--OUTPUTS - ENAME, ESALARY,EAGE, DNAME**

**PROCEDURE TO INSERT ROWS INTO MULTIPLE TABLE**

**CREATE A PROCEDURE TO INSERT ROWS INTO EMP\_MAIN AND EMP\_PHONE TABLES.**

SELECT\*FROM EMPLOYEES

SELECT EMPID,ENAME,EGAE AS EAGE,ESALARY INTO EMP\_MAIN FROM EMPLOYEES

SELECT\*FROM EMP\_MAIN

CREATE TABLE EMP\_PHONE

(

EID INT FOREIGN KEY REFERENCES EMPLOYEES(EMPID),

EMP\_PHONE VARCHAR(20)

)

INSERT INTO EMP\_PHONE VALUES

(1,'8553132202'),

(2,'9999999999'),

(3,'8888888887'),

(4,'7777777777'),

(8,'9898989898')

SELECT \* FROM EMP\_PHONE

SELECT\*FROM EMP\_MAIN

GO

CREATE PROCEDURE USP\_INSERTDATA

(

@EID INT,

@ENAME VARCHAR(20),

@EAGE TINYINT,

@ESALARY MONEY,

@EMP\_PHONE VARCHAR(20)

)

AS

BEGIN

INSERT INTO EMP\_MAIN VALUES (@ENAME,@EAGE,@ESALARY)

INSERT INTO EMP\_PHONE VALUES(@EID,@EMP\_PHONE)

END

EXEC USP\_INSERTDATA 9,'KKKK',28,98066,'3345454545'

--DROP A PROCEDURE

DROPPROCEDURE PROCNAME

**USER DEFINED FUNCTIONS:**

**USER DEFINED FUNCTIONS ARE NOTHING BUT SET OF PRE-COMILED TRANSACT SQL STATEMENTS STORED IN DATABASE.**

**SP AND UDF**

|  |  |
| --- | --- |
| **STORED PROCEDURES** | **USER DEFINED FUNCTIONS** |
| SP CONTAINS ALL DDL/DML STATEMENT TO PERFORM OPRATIONS WITH TABLES | UDF CONTAINS ONLY SELECT STATEMENT TO PERFORM OPERATIONS WITH TABLES |
| SP CAN INPUT AND OUTPUT PARAMETERS | UDF CAN HAVE ONLY INPUT PARAMETERS |
| RETURN STATEMENT CAN RETURN NUMERIC VALUES IN SP | RETURN STATEMENT CAN RETURN SCALAR OR SET OF ROWS IN UDF |
| SP CANNOT BE CALLED DIRECTLY IN DML STATEMENTS, IT IS EXECUTED. | UDF CAN BE CALLED DIRECTLY IN DML STATEMENTS. |

**SYNATAX:**

**CREATE FUNCTION FUNCTIONAME**

**(**

**<INPUT PARAMS>**

**)**

**RETURN RETURNTYPE**

**AS**

**BEGIN**

**------**

**------**

**RETUN SCALAR VALUE OR SET OF ROWS**

**END**

**SELECT DBO.FUNCTIONNAME(INPUT PARAMS)**

**TYPES OF USER DEFINED FUNCTIONS:**

1. **SCALAR UDF**
2. **INLINE UDF**
3. **MULTI STATEMENT TABLE UDF**
4. **SCALAR UDF:**

**WHICH RETURNS SINGLE SCALAR VALUE .**

**EX:**

--CREATE A FUNCTION WHICH RETURNS DEPTNAME OF GIVEN EMPNO.

SELECT D.deptname FROM employees E

INNERJOIN

department D

ON

E.Edid=D.deptid

WHERE empid=9

GO

CREATE FUNCTION UDF\_GETDNAME

(

@ENO INT

)

RETURNS VARCHAR(20)

AS

BEGIN

DECLARE @S VARCHAR(20)

SELECT @S=D.deptname FROM employees E

INNERJOIN

department D

ON

E.Edid=D.deptid

WHERE empid=@ENO

RETURN @S

END

SELECT DBO.UDF\_GETDNAME(9)

1. **INLINE FUNCTION:**

**WHICH RETURNS SET OF ROWS IS CALLED INLINE UDF.**

**INLINE UDF CAN HAVE ONLY ONE STATEMENT THAT TO BE RETURN STATEMENT.**

--CREATE A FUNCTION WHICH RETURNS EMPID,ENAME AND SALARIES OF GIVEN DEPTNO.

SELECT empid,ename,esalary FROM EMPLOYEES where edid=2

CREATE FUNCTION USF\_GET\_EMPDATA

(

@DPTNO INT

)

RETURNS TABLE

AS

RETURN

(SELECT empid,ename,esalary FROM EMPLOYEES where edid=@DPTNO)

--CALLING FUNCTION

SELECT\*FROM DBO.USF\_GET\_EMPDATA(2)

1. **MULTISTATEMENT TABLE VALUE UDF:**

**WHICH RETURNS SET OF ROWS WITH MULTIPLE STATEMENTS.**

**RETURN TYPE WILL BE TABLE VARIABLE.**

**TABLE VARIABLES:**

**TABLE VARIABLE ARE USED TO STORE LIST OF VALUES ( 1D OR 2D) TEMPORARILY.**

DECLARE @X INT-- VARIABLE OF TYPE INTEGER

DECLARE @Y VARCHAR()-- VARIABLE OF TYPE CHAR

DECLARE @TAB TABLE-- VARIBALE OF TYPE TABLE

**EX:**

**DECLARE @TAB TABLE**

**(**

**COL1 INT,**

**COL2 VARCHAR**

**)**

**PREPARE ONE DIMENSIONAL ARRY TO STORE COLLECTION OF STRINGS.**

DECLARE @STR\_TAB TABLE

(

NAME VARCHAR(20)

)

INSERTINTO @STR\_TAB VALUES

('SREEKANTH'),

('KISHORE'),

('SURI'),

('DATTA')

SELECT\*FROM @STR\_TAB

**LIFE TIME OF TABLE VARIABLE IS AS LONG AS SCRIPT IS EXECUTED.**

**TABLE VARIABLES CANT MAINTAIN THE RELATION BETWEEN OTHER TABLE VARIBALES.**

**EX: PREPARE 2D ARRAY TO STORE SET OF VALUES**

DECLARE @EMP TABLE

(

EID INT PRIMARY KEY IDENTITY(1,1),

ENAME VARCHAR(20)

)

INSERT INTO @EMP VALUES

('AAA'),

('BBB'),

('CCC')

SELECT \* FROM @EMP

**TABLE VARIABLES EXISTED IN THE MEMORY AS LONG AS SCRIPT IS EXECUTED.**

**WE CAN’T MAINTAIN RELATIONSHIP USING FOREIGN KEY.**

**COPYING DATA BETWEEN TABLES:**

**TO IMPORT DATA FROM ONE OR MORE TABLES INTO ANOTHER TABLE, T-SQL PROVIDES 2 STATEMENTS:**

1. **SELECT \* INTO**
2. **INSERT INTO SELECT**
3. **SELECT \* INTO:**

**IT AUTOMATICALLY CREATES A DESTINATION TABLE STRUCTURE WITHOUT CONSTRAINTS AND COPIES ROWS FROM ONE OR MORE SOURCE TABLES.**

**SYNTAX:**

**SELECT \* INTO DESTINATION\_TABLE**

**FROM SOURCE\_TABLE**

SELECT\*FROM EMPLOYEES

SELECT\*INTO EMP\_NEW FROM EMPLOYEES

SELECT\*FROM EMP\_NEW

SELECT EMPID,ENAME INTO EMP\_NEW1 FROM EMPLOYEES

SELECT\*FROM emp\_new1

SELECT \* INTO EMP\_DEPT FROM EMPLOYEES E

INNERJOIN

DEPARTMENT D

ON

E.EDID=D.DEPTID

SELECT\*FROM EMP\_DEPT

SELECT \* FROM BRANCHES

--THIS IS USED TO CREATE A NEW TABLE FROM THE EXISTING TABLE AND COPIES TOTAL DATA

SELECT \* INTO BRANCHES\_NEW FROM BRANCHES

--THIS IS USED TO CREATE A NEW TABLE FROM THE EXISTING TABLE AND COPIES SPECIFIED DATA

SELECT \* INTO BRANCHES\_TODAY FROM BRANCHES WHERE branchid IN (1,2,3,5,7,8,10,11)

--THIS IS USED TO CREATE A NEW TABLE FROM THE EXISTING TABLE AND DONT COPY DATA

SELECT \* INTO BRANCHES\_STRUCTURE FROM BRANCHES WHERE 1=0

SELECT \* FROM BRANCHES\_STRUCTURE

1. **INSERT INTO SELECT:**

**THIS IS LIKE A BULK INSERT OF DATA TO THE EXISTING TABLE FROM OTHER EXISTING DATA.**

**SYNTAX:**

**INSERT INTO DESTINARION\_TABLENAME (COLNAMES)**

**SELECT \* (COLNAMES) FROM SOURCE\_TABLE**

**(WHERE..)**

**EX:**

**PREPARE DESTINATION TABLE STRUCTURE:**

**NOTE: FOR INSERT INTO, WE SHOULD HAVE DESTINATION TABLE DEFINED.**

CREATE TABLE EMPTEST

(

EID INT,

ENAME VARCHAR(20),

ESALARY MONEY

)

SELECT \* FROM EMPTEST

SELECT\*FROM EMPLOYEES

INSERT INTO EMPTEST(EID,ENAME,ESALARY)

SELECT EMPID,ENAME,ESALARY FROM EMPLOYEES

SELECT\*FROM EMPTEST

--CREATE A FUNCTION WHICH RETURNS EID,ENAME AND EDSEK OF GIVEN DPNO.

SELECT EMPID,ENAME,ESALARY FROM EMPLOYEES WHERE EDID=@DID

CREATE FUNCTION UDF\_GETEMPDATA\_DNO

(

@DID INT

)

RETURNS @E TABLE

(EID INT,

ENAME VARCHAR(20),

ESAL MONEY

)

AS

BEGIN

INSERT INTO @E(EID,ENAME,ESAL)

SELECT EMPID,ENAME,ESALARY FROM EMPLOYEES WHERE EDID=@DID

RETURN

END

--CALLING FUNCTION

SELECT \* FROM dbo.UDF\_GETEMPDATA\_DNO(3)

**DROP FUNCTION OR DELETE FUNCTION:**

**CREATE THE SAME FUNCTION AGAIN AND NAME IT AS ‘**UDF\_GETEMPDATA\_DNO\_DUPLICATE’

DROP FUNCTION UDF\_GETEMPDATA\_DNO\_DUPLICATE

**TEMPORARY TABLES:**

**THESE ARE USED TO STORE SET OF ROWS TEMPORARILY.**

**TEMPORARY TABLES MEMORY IS ALLOCATED IN TEMPDB DATABASE.**

**TWO TYPES :**

1. **LOCAL TEMPORARY TABLES (#TABLENAME)**
2. **GLOBAL TEMPORARY TABLES (##TABLENAME)**

**TEMPORARY TABLES EXISTS UNTIL WE CLOSE THE CONNECTION. TEMPORARY TABLES CAN MAINTAIN RELATIONSHIPS.**

**EX:**

**CONNECTION1:**

CREATE TABLE #EMP

(

EID INT PRIMARY KEY,

ENAME VARCHAR(20)

)

SELECT \* FROM #EMP

INSERT INTO #EMP VALUES

(1,'AAA'),

(2,'BBB'),

(3,'CCC')

SELECT \* FROM #EMP

**CONNECTION2:**

SELECT \* FROM #EMP

**THIS IS NOT ACCESSABLE IN CONNECTION2.**

**EX: GLOBAL TEMPORARY TABLES**

**CONNCETION1:**

CREATE TABLE ##EMP

(

EID INT PRIMARY KEY,

ENAME VARCHAR(20)

)

SELECT \* FROM ##EMP

INSERT INTO ##EMP VALUES

(1,'AAA'),

(2,'BBB'),

(3,'CCC')

SELECT \* FROM ##EMP

**CONNECTION2:**

SELECT \* FROM ##EMP

**THE GLOBAL TEMPORARY TABLE IS ACCESSIBLE**

**NOTE: SCOPE OF BOTH THE GLOBAL AND LOCAL TEMPORARY TABLES WORK ONLY TILL THE CONNECTION WHERE THE TEMP TABLES ARE CREATED THAT CONNECTION IS AVAILABLE.**

**TRANSACTIONS AND LOCKS:**

**TRANSACTION IS NOTHING BUT A LOGICAL UNIT OF WORK WHICH CONTAINS SET OF OPERATIONS.**

**EVERY TRANSACTION FOLLOWS “ACID” PROPERTIES:**

**A - AUTOMOCITY**

**C - CONSISTENCY**

**I – ISOLATION**

**D – DURABILITY**

**AUTMOCITY: THIS GUARANTEES THAT TRANSACTION COMITTED IF ALL THE OPERATIONS ARE SUCCESSFUL.**

**TRANSACTION ROLLBACK IF ANY OPERATION FAILED.**

**CONSISTENCY: CONSISTENCY IS NOTHING BUT PROVIDING CORRECT DATA TO THE TRANSACTION.**

**THIS CAN BE IMPLEMENTED WITH THE HELP OF LOCKING.**

**LOCKING MECHANISM ELIMATES THE CONCURRENCY PROBLEMS AND PROVIDES CONSISTENCY. LOCKS CAN BE IMPLEMENTED AT DIFFERENT LEVELS.**

**DATABASE LEVEL**

**TABLE LEVEL**

**PAGE LEVEL**

**ROW LEVEL**

**WE HAVE FOUR TYPES OF LOCKS:**

1. **SHARED LOCK (S) – WHICH CAN BE APPLIED FOR SELECT STATEMENT.**
2. **UPDATE LOCK(U) - WHICH CAN BE APPLIED FOR UPDATE STATEMENT.**
3. **EXCLUSIVE LOCK(X) – ANY DML STATEMENT.**

**ONCE A EXCLUSIVE LOCK IS APPLIED ON ONE TRANSACTION THEN WE NEED TO WAIT UNTIL IT IS COMPLETED IN ORDER TO START ANOTHER TRANSACTION.**

1. **INTENT LOCK(I) – A TRANSACTION APPLYING LOCK AT HIGH LEVEL AND OTHER TRANSACTION APPLYING LOCK AT LOW LEVEL IS CALLED INTENT LOCK.**

**HIGH LEVEL – TABLE LEVEL**

**LOW LEVEL – ROW LEVEL**

**SQL SERVER INTERNALLY CONTAINS THE ‘LOCK MANAGER’**

**TO MONITOR LOCK INFORMATION WE CAN USE ‘SP\_LOCK’**

**ISOLATION: ONE TRANSACTION FAILURE SHOULD NOT EFFECT THE OTHER TRANSACTIONS IN THE SYSTEM.**

**DURABILITY: DURABILITY GUARANTEES THAT RESULT OF EVERY COMITTED TRANSACTION SHOULD BE PERMANENTLY SAVED IN THE DATABASE TABLES.**

**COMITTED MEANS RESULT IS PERMANENT IN THE TABLE.**

**ROLLBACK MEANS TABLE IS NOT EFFECTED AFTER THE OPERATION.**

**TYPES OF TRANSACTIONS:**

1. **IMPLICIT TRANSACTION:**

**EVERY STATEMENT CREATES ONE TRANSACTION IMPLICITLY.**

**IMPLICIT TRANSACTIONS ARE AUTOMATICALLY COMMITED.**

1. **EXPLICIT TRANSACTION**

**EXPLICIT TRANSACTIONS ALLOWS US TO PERFORM GROUP OF OPERATIONS IN SINGLE TRANSACTION.**

**TP PERFROM EXPLICIT TRANSACTIONS WE CAN USE TCL – TRANSACTION CONTROL LANGUAGE.**

**BEGIN TRANSACTION**

**--CODE**

**SAVE TRANSACTION**

**--CODE**

**ROLLBACK TRANSACTION**

**--IT WILL UNDO THE TRANSACTION OPERATION RESULT**

**COMMIT TRANSACTION**

**--THIS WILL SAVE THE TRANSACTION OPERATIONS TO THE DATABASE.**

**SYNTAX:**

**BEGIN TRANSACTION**

**------**

**------**

**SAVE POINT**

**----**

**----**

**COMMIT/ROLLBACK TRANSACTION**

--PREPARE EXPLICIT TRANSACTION TO TRANSFER GIEVN AMOUNT FROM ONE ACCOUNT TO OTHER ACCOUNT

--FOR EVERY ACCOUNT MINIMUM BALANCE IS 5000

CREATE TABLE ACCOUNTS

(

ACCNO INTPRIMARYKEY,

BALANCE MONEY

)

INSERTINTO ACCOUNTS VALUES(101,75000)

INSERTINTO ACCOUNTS VALUES(102,5000)

SELECT\*FROM ACCOUNTS

CREATE PROCEDURE USP\_AMOUNT\_TRANSFER

(

@ACC1 INT,

@ACC2 INT,

@AMOUNT MONEY,

@STATUS VARCHAR(50) OUTPUT

)

AS

BEGIN

BEGIN TRANSACTION

UPDATE ACCOUNTS SET BALANCE=BALANCE-@AMOUNT WHERE ACCNO=@ACC1

UPDATE ACCOUNTS SET BALANCE=BALANCE+@AMOUNT WHERE ACCNO=@ACC2

--VERIFY BALANCE OF ACCNO1

IF(SELECT BALANCE FROM ACCOUNTS WHERE ACCNO=@ACC1)<5000

BEGIN

SET @STATUS='TRANSACTION FAILED'

ROLLBACKTRANSACTION

END

ELSE

BEGIN

SET @STATUS='TRANSACTION SUCCESSFUL'

COMMIT TRANSACTION

END

END

--EXECUTION

DECLARE @STATUS VARCHAR(50)

EXEC USP\_AMOUNT\_TRANSFER101,102,70000, @STATUS OUTPUT

PRINT @STATUS

SELECT\*FROM ACCOUNTS

--EXECUTION

DECLARE @STATUS VARCHAR(50)

EXEC USP\_AMOUNT\_TRANSFER101,102,2000, @STATUS OUTPUT

PRINT @STATUS

SELECT\*FROM ACCOUNTS

**DEADLOCK:**

**DEAD LOCK S A SITUATION WHERE ONE TRANSACTION CONFLITS WITH OTHER TRANSACTION WITH THE SYSTEM.**

**DBA PROVIDES TWO TYPES OF SOLUTIONS FOR DEADLOCK:**

1. **RESTART THE TRANSACTIONS.**
2. **EXECUTE THE TRANSACTIONS SEQUENTIALLY.**

**TRIGGERS:**

**TRIGGERS ARE NOTHING BUT A SPECIAL KIND OF STORED PROCEDURE WHICH WILL BE EXECUTED AUTOMATICALLY BASED ON USER EVENTS (DATABASE EVENETS).**

**THESE TRIGGERS ARE CATEGORIZED INTO TWO TYPES:**

1. **DML TRIGGERS (EVENTS)- insert , update, delete**
2. **DDL TRIGGERS (EVENTS) –create, alter, drop**

**DDL – DATA DEFINITION LANGUAGE**

**DML – DATA MANIPULATION LANGUAGE**

**DCL – DATA CONTROL LANGUAGE**

**DRL – DATA RETREIVAL LANGUAGE**

**TCL – TRANSACTION CONTROL LANGUAGE**

**DML TRIGGERS: TRIGGERS WHICH ARE CREATED ON TABLES/VIEWS ARE CALLED AS DML TRIGGERS.**

**SYNTAX:**

**CREATE TRIGGER TRIGGER\_NAME**

**ON TABLENAME/VIEWNAME**

**AFTER <INSERT,DELETE,UPDATE>**

**OR**

**INSTEAD OF<INSERT,DELETE,UPDATE>**

**AS**

**BEGIN**

**--CODE**

**END**

**DML TRIGGERS ARE CLASSIFIED INTO TWO SUBTYPES:**

1. **AFTER TRIGGER**
2. **INSTEAD OF TRIGGER**

|  |  |
| --- | --- |
| **AFTER TRIGGER** | **INSTEAD OF TRIGGER** |
| 1. THESE TRIGGERS ARE CREATED ON TABLES. | 1. THESE TRIGGERS ARE CREATED ON TABLES AS WELL AS VIEWS. |
| 2. THERE IS NO LIMIT FOR NO.OF TRIGGERS ON A TABLE. | 2. ONLY 3 TRIGGERS WE CAN CREATE TO MAX. |
| 3. SINGLE AFTER TRIGGER CAN BE EXECUTED FOR MULTIPLE OPERATIONS. | 3. SINGLE INSTEAD OF TRIGGERS CAN BE EXECUTED FOR SINGLE OPERATION ONLY. |

**ADVANTAGES OF TRIGGERS:**

1. **ENFORCE USER DEFINED DATA INTEGRITY.**
2. **CHECKING COMPLEX BUSINESS LOGIC.**
3. **POPULATING AUDIT TABLES AUTOMATICALLY.**
4. **PROVIDING SECURITY FOR THE TABLES.**

**MAGIC TABLES:**

**WHILE PERFORMING DML OPERATIONS ON TABLES, IMPLICIT TRANSACTIONS CREATED AND TEMPORARY MAGIC TABLE GETS CREATED.**

**THE STRUCTURE OF MAGIC TABLE WILL BE SAME AS ACTUAL TABLE.**

**COMMIT OR ROLLBACK DOES NOT GOES TO THE ORIGINAL TABLE.**

**BELOW ARE THE MAGIC TABLES CREATED UPON DML OPERATIONS:**

**INSERT → INSERTED**

**DELETE → DELETED**

**UPDATE→ DELETED – FOR OLD ROWS**

**INSERTED – FOR NEW ROWS.**

1. **ENFORCE USER DEFINED DATA INTEGRITY WITH THE HELP OF TRIGGER:**

**EX: CREATE A TRIGGER TO VERIFY WHETHER USER INSERTING SALARY MORE THAN 6000 OR NOT.**

**IF GIVEN SALARY IS 6000 OR LESS THAN 6000 THEN OUTPUT SHOULD BE ‘SALARY SHOULD BE GREATER THAN 6000’**

SELECT\*FROM EMP\_TRIGGER

CREATETRIGGER TRIG\_CHECKSALARY

ON EMP\_TRIGGER

AFTERINSERT

AS

BEGIN

IF(SELECT SALARY FROM INSERTED )<6000

PRINT'SALARY SHOULD BE GREATER THAN 6000'

ROLLBACKTRANSACTION

END

--TESTING

INSERTINTO EMP\_TRIGGER VALUES ('RAJASHREE',5,5000)

1. **CHECKING COMPLEX BUSINSESS LOGIC WITH THE HELP OF TRIGGER:**

**CREATE A TRIGGER TO RESTRICT THE DELETION OF JOB CATEGORY “MANAGER” FROM EMPLOYEE TABLE.**

CREATE TABLE EMP

(

EID INT IDENTITY(1,1),

ENAME VARCHAR(20),

EAGE INT,

EDESIGNATION VARCHAR(30),

ESALARY MONEY

)

--DELETE FROM EMP\_DEL\_TRIGGER WHERE DESIGNATION='MANAGER'

CREATETRIGGER TRIG\_DELETE

ON EMP

AFTERDELETE

AS

BEGIN

IF(SELECTCOUNT(\*)FROM DELETED WHERE EDESIGNATION='MANAGER')>0

BEGIN

PRINT'MANAGER RECORD CANNOT BE DELETED'

ROLLBACKTRANSACTION

END

END

--TESTING

SELECT\*FROM EMP\_DEL\_TRIGGER

DELETEFROM EMP\_DEL\_TRIGGER WHERE DESIGNATION='MANAGER'

DELETEFROM EMP\_DEL\_TRIGGER WHERE EID=5

DELETEFROM EMP\_DEL\_TRIGGER WHERE EID=12

**THE SAME CAN BE IMPLEMENTED AS BELOW:**

CREATE TRIGGER TRIGGER\_DELETE\_EMP

ON EMP

AFTER DELETE

AS

BEGIN

IF(SELECT DISTINCT EDESIGNATION FROM DELETED)='MANAGER'

BEGIN

PRINT 'WE CANNOT DELETE RECORDS WHOSE DESIGNATION IS MANAGER'

ROLLBACK TRANSACTION

END

END

1. **POPULATING TRIGGER AUDIT TABLES/RELATED TABLES AUTOMATICALLY.**

CREATE TABLE EMP\_AUDIT

(

EMPNO INT,

ENAME VARCHAR(20),

USERINFO VARCHAR(20),

TIMEINFO DATETIME,

OPCODE CHAR(3)

)

SELECT\*FROM EMP\_AUDIT

SELECT\*FROM EMP

CREATE TRIGGER TRIGGER\_AUDIT\_EMP\_AUDIT

ON EMP

AFTER INSERT, DELETE

AS

BEGIN

INSERT INTO EMP\_AUDIT (EMPNO,ENAME,USERINFO,TIMEINFO,OPCODE)

SELECT EID,ENAME, SUSER\_NAME(), GETDATE(),'INS' FROM inserted

INSERT INTO EMP\_AUDIT (EMPNO,ENAME,USERINFO,TIMEINFO,OPCODE)

SELECT EID,ENAME, SUSER\_NAME(),GETDATE(),'DEL' FROM deleted

END

INSERT INTO EMP VALUES

('AAA',25,'SSE',50000),

('BBB',29,'TECH LEAD',50000),

('CCC',35,'MANAGER',50000),

('DDD',38,'ARCHITECT',50000),

('EEE',40,'SE',50000)

SELECT \* FROM EMP

SELECT\*FROM EMP\_AUDIT

DELETE FROM EMP WHERE EID=1

**DDL TRIGGERS:**

**TRIGGERS WHICH ARE CREATED AT DATABASE LEVEL OR SERVER LEVEL CAN BE CALLED AS DDL TRIGGERS.**

1. **PROVIDING SECURITY FOR THE TABLES.**

**CREATE A TRIGGER ON DATABASE LEVEL WHICH RESTRICTS ANY USER OF CREATING TABLES / DROPPING TABLES.**

CREATETRIGGER TRIG\_SECURITY

ONDATABASE

AFTER CREATE\_TABLE, DROP\_TABLE

AS

BEGIN

PRINT'YOU CANNOT CREATE OR DROP TABLES IN THIS DATABASE'

ROLLBACKTRANSACTION

END

--

CREATE TABLE ABC

(

XYZ VARCHAR(20)

)

DROPTABLE EMP\_AUDIT

**ENABLE OR DISABLE TRIGGER IN ORDER TO WORK ON ALL THE OPERATIONS:**

**TO IMPROVE THE PERFORMANCE WE DISABLE THE TRIGGERS.**

**ALTER TABLE TABLENAME**

**ENABLE/DISABLE**

**TRIGGER TRIGGERNAME (OR ALL)**

ALTERTABLE EMP\_DEL\_TRIGGER

DISABLE

TRIGGER TRIG\_DELETE

**DROP TRIGGER:**

DROPTRIGGER TRIG\_DELETE

**IF WE DROP THE TABLE – ALL THE ASSOCIATED TRIGGERS WILL ALSO BE DROPPED.**

**CURSORS:**

**CURSOR IS NOTHING BUT A TEMPORARY MEMORY ON DATABASE SERVER WHICH STORES RESULTSET TEMPORARILY.**

**CURSORS MEMORY IS ALLOCATED IN TEMPDB DATABASE.**

**CURSOR ALLOWS US TO PROCESS THE RESULT/DATA ROW BY ROW.**

**TO WORK WITH CURSORS, WE HAVE TO FOLLOW 5 STEPS IN T-SQL:**

**STEP1: DECLARING CURSOR**

**SYNTAX: DECLARE CURSORNAME CURSOR**

**[LOCAL/GLOBAL]**

**NAVIGATIONTYPE → SCROLL**

* **FORWARD-ONLY**

**CURSOR TYPE :**

**STATIC**

**KEYSET**

**FAST-FORWARD**

**DYNAMIC**

**LOCK TYPE:**

**READONLY**

**SCROLL LOCK**

**OPTIMISTIC LOCK**

**FOR**

**SELECT …. STMT**

**STEP2: ---OPEN THE CURSOR**

**OPEN CURSORNAME**

**STEP3: ---FETCH THE ROWS**

**FETCH(FIRST|LAST|PRIOR|NEXT)**

**(ABSOLUTE <POSITION>)**

**(RELATIVE <POSITION>)**

**FROM CURSORNAME**

**INTO TEMPVARIABLES**

**STEP4 : ---CLOSE THE CURSOR**

**CLOSE CURSORNAME**

**STEP5: ---DEALLOCATE THE CURSOR**

**DEALLOCATE CURSORNAME**

--PREPARE CURSOR SCRIPT TO PROCESS "FIRST ROW AND LAST ROW" OF EMPLOYEES TABLE

SELECT\*FROMEMPLOYEES

--STEP1:

DECLAREEMPCURSORCURSOR

LOCAL

SCROLL

DYNAMIC

OPTIMISTIC

FOR

SELECTENO,ENAMEFROMEMPLOYEES

OPENEMPCURSOR

DECLARE@IDINT,@NAMEVARCHAR(20)

FETCHFIRST

FROMEMPCURSOR

INTO@ID,@NAME

IF(@@FETCH\_STATUS=0)

PRINT'FIRST EMPNO IS : '+CAST(@IDASCHAR)+' EMP NAME IS : '+@NAME

ELSE

PRINT'FIRST RECORD FETCHING FAILED'

FETCHLAST

FROMEMPCURSOR

IF(@@FETCH\_STATUS=0)

PRINT'LAST EMPNO IS : '+CAST(@IDASCHAR)+' EMP NAME IS : '+@NAME

ELSE

PRINT'LAST RECORD FETCHING FAILED'

CLOSEEMPCURSOR

DEALLOCATEEMPCURSOR

@@FETCH\_STATUS-- STORES STATUS OF FETCH STATEMENT

0 --- SUCCESSFUL

1 OR 2 -FAILED

1 -- FETCH THE ROW DELETED BY SOME OTHER USER.

2 -- FETCH THE ROW BUT IT IS USED BY SOME OTHER USER.

**PROPERTIES OF CURSOR:**

1. **SCOPE OF CURSOR**
2. **NAVIGATION TYPE**
3. **CURSOR TYPE**
4. **LOCK TYPE**
5. **SCOPE OF CURSOR:**

**SCOPE MAY BE LOCAL OR GLOBAL, ONCE WE DECLARE THE CURSOR AS LOCAL INSIDE THE STORED PROCEDURE THEN THE CURSOR CAN BE ACCESSIBLE ONLY WITHIN THAT STORED PROCEDURE ONLY.**

**ONCE WE DECLARE THE CURSOR AS GLOBAL INSIDE THE STORED PROCEDURE THEN THE CURSOR CAN BE ACCESSIBLE OUTSIDE THE PROCEDURE EXECUTION ALSO.**

1. **NAVIGATION TYPE:**

**FORWARD-ONLY|SCROLL**

**FORWARD-ONLY → PROCESSING ONLY IN FORWARD DIRECTION.**

**SCROLL → PROCESSING IN ANY DIRECTION.**

1. **CURSOR TYPES (TYPES OF CURSORS**
2. **STATIC**

**STATIC CURSORS TAKE MORE MEMORY. ANY MODIFICATIONS IN THE TABLE ARE NOT REFLECTED IN THE STATIC CURSOR RESULT.**

1. **KEYSET**

**IT TAKES LESS MEMORY, DELETIONS, UPDATIONS IN THE TABLE ARE REFLECTED IN THE KEYSET CURSOR.**

1. **FAST-FORWARD**

**IT IS SAME AS KEYSET CURSOR, FAST-FORWARD CURSOR ALLOWS US TO PROCESS THE RESULT IN ONLY FORWARD DIRECTION AND HERE IT GIVES HIGH PERFORMANCE.**

1. **DYNAMIC**

**ANY MODIFICATIONS IN THE TABLE WILL BE DYNMICALLY REFLECTED IN DYNAMIC CURSOR AND IT GIVES LESS PERFORMANCE.**

1. **LOCK TYPES:**
2. **READ ONLY [SHARED LOCK]**

**WE CAN PERFORM ONLY READ OPERATIONS FROM THE CURSOR.**

1. **SCROLL LOCK [EXCLUSIVE LOCK]**

**WE CAN PERFORM MODIFICATION IN THE CURSOR RESULT PROVIDED ONLY ONE USER IS ALLOWED AT A TIME TO MODIFY.**

1. **OPTIMISTIC LOCK[UPDATE LOCK]**

**WE CAN PERFORM MODIFICATION IN THE CURSOR RESULT PROVIDED MULTIPLE USERS ARE ALLOWED AT A TIME TO MODIFY CURSOR RESULT.**

**ABSOULTE OR RELATIVE POSITION:**

**ABSOLUTE POSITION ALWAYS CALCUALTED FROM BEGINNING**

**RELATIVE POSITION ALWAYS CALCUALTED FROM CURRENT POSITION.**

**RELATIVE POSITION MAY BE +VE OR –VE**

--FETCH ABSOLUTE 50 FROM EMPCURSOR

DECLARE@IDINT,@NAMEVARCHAR(20)

FETCH ABSOLUTE 50 FROMEMPCURSOR

INTO @ID,@NAME

CLOSE EMPCURSOR

DEALLOCATE EMPCURSOR

--CREATE A SP AND PREPARE CURSOR SCRIPT TO PROCESS ALL THE ROWS FROM EMPLOYEE TABLE

CREATEPROCEDUREUSP\_GETEMP\_CURSOR

AS

BEGIN

DECLAREEMPCURSORCURSOR

LOCAL

SCROLL

DYNAMIC

OPTIMISTIC

FOR

SELECTENO,ENAMEFROMEMPLOYEES

OPENEMPCURSOR

DECLARE@IDINT,@NAMEVARCHAR(20)

FETCHFIRST

FROMEMPCURSOR

WHILE(@@FETCH\_STATUS=0)

BEGIN

PRINT'EMPNO IS : '+CAST(@IDASCHAR)+' EMP NAME IS : '+@NAME

FETCHNEXTFROMEMPCURSORINTO@ID,@NAME

END

CLOSEEMPCURSOR

DEALLOCATEEMPCURSOR

END

**TO FIND OUT CURRENT POSITION OF CURSOR POINTER:**

**SYNTAX : CURRENT OF CURSORNAME**

**EX:** **FETCH ABSOLUTE(50) FROM EMPCURSOR INTO @ID,@NAME**

**DELETE FROM EMPLOYEES**

**WHERE CURRENT OF EMPCURSOR**

**FETCH ABSOLUTE(50) FROM EMPCURSOR INTO @ID,@NAME**

**UPDATE EMPLOYEES SET SAL=85000**

**WHERE CURRENT OF EMPCURSOR**

--PREPARE CURSORS SCRIPT TO DELETE FIRST ROW AND UPDATE THE LAST ROW OF EMPLOYEES TABLE

CREATEPROCEDUREUSP\_EMP\_CURSOR

AS

BEGIN

DECLAREEMP\_CURSORCURSOR

GLOBAL--ACCESSIBLITY

SCROLL

DYNAMIC

OPTIMISTIC

FOR

SELECTENO,ENAMEFROMEMPLOYEES

END

EXECUSP\_EMP\_CURSOR

OPENEMP\_CURSOR

DECLARE@IDINT,@NAMEVARCHAR(20)

FETCHFIRSTFROMEMP\_CURSORINTO@ID,@NAME

DELETEFROMEMPLOYEES

WHERECURRENTOFEMP\_CURSOR

FETCHLASTFROMEMP\_CURSORINTO@ID,@NAME

UPDATEEMPLOYEESSETSALARY=851111

WHERECURRENTOFEMP\_CURSOR

--F5

CLOSEEMP\_CURSOR

DEALLOCATEEMP\_CURSOR

--F5

**INDEXES:**

**INDEX IS NOTHING BUT ORDERED LIST OF VALUES TAKEN FROM ONE OR MORE COLUMNS OF A TABLE AND ORGANIZED INTO B-TREE STRUCTURE.**

**IN SQL SERVER WE HAVE TWO TYPES OF INDEXES:**

1. **CLUSTERED INDEX**
2. **NON-CLUSTERED INDEX.**

**SYNTAX:**

**CREATE [UNIQUE]**

**<CLUSTERED/NONCLUSTURED> INDEX <INDEXNAME>**

**ON TABLENAME (COLUMN NAME)**

**NOTE: WE CAN CREATE AN INDEX ON SINGLE COLUMN – SIMPLE INDEX.**

**WE CAN CREATE AN INDEX ON MULTIPLE COLUMNS – COMPOSITE INDEX.**

CREATE TABLE STUDS

(

STID INT,

SNAME VARCHAR(20),

SAGE TINYINT,

STMOBILE VARCHAR(20)UNIQUE,

FEES MONEY

)

SELECT \* FROM STUDS

--

CREATE NONCLUSTERED INDEX IDX\_MOBILE ON STUDS(SNAME)

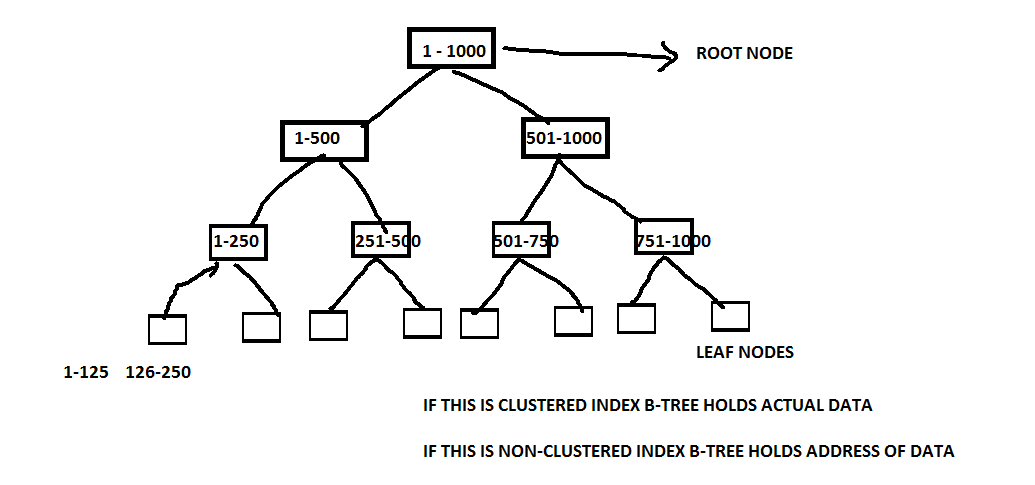
CREATE CLUSTERED INDEX IDX\_SID ON STUDS(STID)

**TO LIST OUT THE INDEXES INFO OF A TABLE:**

SP\_HELPINDEX 'STUDS'

|  |  |
| --- | --- |
| **CLSUTERED INDEX** | **NON CLSUTERED INDEX** |
| A TABLE CAN HAVE ONLY ONE CLUSTERED INDEX | A TABLE CAN HAVE 249 NON CLUSTERED INDEXES |
| LEAF PAGES IN THE B-TREE CONTAINS ACTUAL ROWS | LEAF PAGES IN THE B-TREE CONTAINS ADDRESS OF ROWS |
| ACTUAL ROWS ARE ARRANGED IN SAME ORDER AS INDEX | ACTUAL ROWS MAY NOT BE ARRANGED IN THE SAME ORDER AS INDEX |

**HERE WHEN WE TRY TO SEARCH FOR VALUE 200, WE CAN SEE HOW FAST WE CAN FIND THE DATA.**

****

**CLUSTERED INDEX GIVES BETTER PERFORMANCE THEN NON-CLUSTERED INDEX.**

**ADVANTAGES :**

1. **FAST RETRIVAL OF DATA FROM THE TABLES.**
2. **IMPROVES PERFORMANCE WHILE SEARCHING THE TABLE.**

**SYNTAX:**

**SELECT QUERY……**

**SET OPERATOR**

**SELEC QUERY…**

**UNION – COMBINATION OF TABLES DATA WITHOUT DUPLICATES.**

**UNION ALL - COMBINATION OF TABLES DATA ALONG WITH DUPLICATES.**

**INTERSECT – COMMON DATA FROM BOTH TABLES.**

**EXCEPT – DATA EXCEPT FROM THE BOTTOM TABLE.**

CREATE TABLE EMP\_US

(

EID INT,

ENAME VARCHAR(20)

)

CREATE TABLE EMP\_INDIA

(

EMPID INT,

EMPNAME VARCHAR(20)

)

INSERT INTO EMP\_US VALUES

(1,'BALU'),

(2,'RAJU'),

(3,'KUMAR')

INSERT INTO EMP\_INDIA VALUES

(4,'HARI'),

(1,’BALU’),

(5,'RAJESH')

SELECT\*FROM EMP\_US

SELECT\*FROM EMP\_INDIA

--DISPLAY ALL EMPLOYEES INFORMATION

SELECT EMPID,EMPNAME FROM EMP\_NEPAL

UNION

SELECT EID,ENAME FROM EMP\_US ORDER BY EMPID

SELECT \* FROM EMP\_NEPAL

SELECT \* FROM EMP\_US

--DISPLAY ALL EMPLOYEES INFORMATION INCLUDING DUPLICATES

SELECT EMPID,EMPNAME FROM EMP\_NEPAL

UNION ALL

SELECT EID,ENAME FROM EMP\_US ORDER BY EMPID

--DISPLAY EMPLOYEES WHO ARE WORKING IN BOTH PLACES

SELECT EID,ENAME FROM EMP\_US

INTERSECT

SELECT EMPID,EMPNAME FROM EMP\_NEPAL ORDER BY EID

--DISPLAY EMPLOYEES WHO ARE WORKING IN ONLY INDIA BIT IN US

SELECT EMPID,EMPNAME FROM EMP\_NEPAL

EXCEPT

SELECT EID,ENAME FROM EMP\_US

--DISPLAY EMPLOYEES WHO ARE WORKING IN ONLY US BIT IN INDIA

SELECT EID,ENAME FROM EMP\_US

EXCEPT

SELECT EMPID,EMPNAME FROM EMP\_NEPAL

**CTE:**

**CTE IS CALLED AS COMMON TABLE EXPRESSIONS.**

**IT IS A TEMPORARY ALIAS NAME FOR RESULT OF A QUERY.**

**SCOPE OF CTE IS TILL THE SCIRPT GETS EXECUTED.**

**SYNTAX:**

**WITH CTENAME (COLUMNNAMES)**

**AS**

**(**

**SELECT QUERY**

**)**

**SELECT \* FROM CTENAME**

**EX:**

WITH EMP\_DATA(EID,ENAME,ESALARY)

AS

(

SELECT EMPID,ENAME,ESALARY FROM employees\_1\_m

WHERE esalary>5000

)

SELECT \* FROM EMP\_DATA

WITH EMP\_DATA

AS

(

SELECT \* FROM employees\_1\_m

WHERE esalary>5000

)

SELECT EMPID,ESALARY FROM EMP\_DATA

**SYNONYMS:**

**SYNONYMS ARE NOTHING BUT PERMANENT ALIAS NAME STORED IN DATABASE.**

**SYNONYMS CAN BE CREATED FOR DIFFERENT DATABASE OBJECTS LIKE TABLES, VIEWS, STORED PROCEDURES..**

**SYNTAX:**

**CREATE SYNONYM SYNONYMNAME**

**FOR OBJECT**

CREATE SYNONYM SYN\_CUST

FOR CUSTOMERS

**Practical SQL Interview Questions:**

In this part, we will see SQL practice questions which contain both complex SQL queries interview questions and basic SQL Interview Questions. Let’s see important SQL queries for interview

**76. How to get unique records from a table?**  
By using DISTINCT keyword, we can get unique records from a table

|  |  |
| --- | --- |
| 1 | SELECT DISTINCT Col1, Col2 from Table1 |

**77. What is the command used to fetch the first 5 characters of a string?**  
Some of the ways to fetch the first 5 characters of a string are as follows:

|  |  |
| --- | --- |
| 1  2 | SELECT RIGHT(EmpName,5) AS EmployeeName FROM Employee  SELECT SUBSTRING(EmpName,1,5) AS EmployeeName FROM Employee |

**78. How to add new Employee details in an Employee\_Details table with the following details**  
**Employee\_Name: John, Salary: 5500, Age: 29?**

|  |  |
| --- | --- |
| 1 | INSERT into Employee\_Details (Employee\_Name, Salary, Age) VALUES (‘John’, 5500 , 29); |

[View Detailed Post](https://www.softwaretestingmaterial.com/sql-insert-query/)

**79. How to add a column ‘Salary’ to a table Employee\_Details?**

|  |  |
| --- | --- |
| 1 | ALTER TABLE Employee\_Details ADD (Salary); |

[View Detailed Post](https://www.softwaretestingmaterial.com/sql-alter/)

**80. How to change a value of the field ‘Salary’ as 7500 for an Employee\_Name ‘John’ in a table Employee\_Details?**

|  |  |
| --- | --- |
| 1 | UPDATE Employee\_Details set Salary = 7500 where Employee\_Name = ‘John’; |

[View Detailed Post](https://www.softwaretestingmaterial.com/sql-update-query/)

**81. Write an SQL Query to select all records from the table?**

|  |  |
| --- | --- |
| 1 | Select \* from table\_name; |

[View Detailed Post](https://www.softwaretestingmaterial.com/sql-select-query/)

**82. How To Get List of All Tables From A DataBase?**To view the tables available on a particular DataBase

|  |  |
| --- | --- |
| 1  2  3  4 | USE TestDB  GO  SELECT \* FROM sys.Tables  GO |

**83. Define SQL Delete statement.**  
The SQL Delete statement is used to delete records from a table.

|  |  |
| --- | --- |
| 1 | DELETE FROM table\_name WHERE some\_column=some\_value; |

[View Detailed Post](https://www.softwaretestingmaterial.com/sql-delete/)

**84. Write the command to remove all Players named Sachin from the Players table.**

|  |  |
| --- | --- |
| 1 | DELETE from Players WHERE Player\_Name = ‘Sachin’ |

**85. How to fetch values from *TestTable1* that are not in *TestTable2*without using NOT keyword?**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | --------------  | TestTable1 |  --------------  |     11     |  |     12     |  |     13     |  |     14     |  -------------- |



|  |  |
| --- | --- |
| 1  2  3  4  5  6 | --------------  | TestTable2 |  --------------  |     11     |  |     12     |  -------------- |

By using the except keyword



|  |  |
| --- | --- |
| 1 | SELECT \* FROM TestTable1 EXCEPT SELECT \* FROM TestTable2; |

**86. How to get each name only once from an employee table?**  
By using the DISTINCT keyword, we could get each name only once.



|  |  |
| --- | --- |
| 1 | SELECT DISTINCT employee\_name FROM employee\_table; |

**87. How to rename a column in the output of SQL query?**  
By using SQL AS keyword



|  |  |
| --- | --- |
| 1 | SELECT column\_name AS new\_name FROM table\_name; |

**88. What is the order of SQL SELECT?**  
Order of SQL SELECT statement is as follows

SELECT, FROM, WHERE, GROUP BY, HAVING, ORDER BY.

**89. How to display the current date in SQL?**

In SQL, there is a built-in function called GetDate() which helps to return the current date.



|  |  |
| --- | --- |
| 1 | SELECT GetDate(); |

**90. Write an SQL Query to find an Employee\_Name whose Salary is equal or greater than 5000 from the below table Employee\_Details.**



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | |  Employee\_Name   |  Salary|  -----------------------------  |  John            |  2500  |  |  Emma            |  3500  |  |  Mark            |  5500  |  |  Anne            |  6500  |  ----------------------------- |

Syntax:



|  |  |
| --- | --- |
| 1 | SELECT Employee\_Name FROM Employee\_Details WHERE Salary>=5000; |

Output:



|  |  |
| --- | --- |
| 1  2  3  4  5 | |  Employee\_Name   |  Salary|  -----------------------------  |  Mark            |  5500  |  |  Anne            |  6500  |  ----------------------------- |

**91. Write an SQL Query to find list of Employee\_Name start with ‘E’ from the below table**



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | |  Employee\_Name   |  Salary|  -----------------------------  |  John            |  2500  |  |  Emma            |  3500  |  |  Mark            |  5500  |  |  Anne            |  6500  |  ----------------------------- |

Syntax:



|  |  |
| --- | --- |
| 1 | SELECT \* FROM Employee\_Details WHERE Employee\_Name like 'E%'; |

Output:



|  |  |
| --- | --- |
| 1  2  3  4 | |  Employee\_Name   |  Salary|  -----------------------------  |  Emma            |  3500  |  ----------------------------- |

**92. Write SQL SELECT query that returns the FirstName and LastName from Employee\_Details table.**



|  |  |
| --- | --- |
| 1 | SELECT FirstName, LastName FROM Employee\_Details; |

**93. How to rename a Table?**



|  |  |
| --- | --- |
| 1 | SP\_RENAME TABLE 'SCOREBOARD', 'OVERALLSCORE' |

To rename Table Name & Column Name



|  |  |
| --- | --- |
| 1  2 | sp\_rename OldTableName,NewTableName  sp\_rename 'TableName.OldColumnName', 'NewColumnName' |

**94. How to select all the even number records from a table?**  
To select all the even number records from a table:



|  |  |
| --- | --- |
| 1  2 | Select \* from table where id % 2 = 0 |

**95. How to select all the odd number records from a table?**  
To select all the odd number records from a table:



|  |  |
| --- | --- |
| 1 | Select \* from table where id % 2 != 0 |

**96. What is the SQL CASE statement?**  
SQL Case statement allows embedding an if-else like clause in the SELECT statement.

**97. Can you display the result from the below table TestTable based on the criteria*M,m* as *M* and *F, f* as *F* and *Null* as *N* and *g, k, I* as *U***



|  |  |
| --- | --- |
| 1 | SELECT Gender from TestTable |



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | |  Gender  |  ------------  |     M    |  |     F    |  |   NULL   |  |     m    |  |     f    |  |     g    |  |     H    |  |     i    |  ------------ |

By using the below syntax we could achieve the output as required.



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | SELECT Gender,  case  when Gender='i' then 'U'  when Gender='g' then 'U'  when Gender='H' then 'U'  when Gender='NULL' then 'N'  else upper(Gender)  end as newgender from TestTable GROUP BY Gender |

**98. What will be the result of the query below?**



|  |  |
| --- | --- |
| 1 | select case when null = null then 'True' else 'False' end as Result; |

This query returns “False”. In the above question, we could see null = null is not the proper way to compare a null value. To compare a value with null, we use IS operator in SQL.

So the correct way is as follows



|  |  |
| --- | --- |
| 1 | select case when null is null then 'True' else 'False' end as Result; |

**99. What will be the result of the query below?**



|  |  |
| --- | --- |
| 1 | select case when null is null then 'Queries In SQL Server' else 'Queries In MySQL' end as Result; |

This query will returns “Queries In SQL Server”.

**100. How do you update F as M and M as F from the below table TestTable?**



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | |  Name    |  Gender  |  ------------------------  |  John    |     M    |  |  Emma    |     F    |  |  Mark    |     M    |  |  Anne    |     F    |  ------------------------ |

By using the below syntax we could achieve the output as required.



|  |  |
| --- | --- |
| 1 | UPDATE TestTable SET Gender = CASE Gender WHEN 'F' THEN 'M' ELSE 'F' END |

**101. Describe SQL comments?**Single Line Comments: Single line comments start with two consecutive hyphens (–) and ended by the end of the line  
Multi-Line Comments: Multi-line comments start with /\* and end with \*/. Any text between /\* and \*/ will be ignored.

**102. What is the difference between NVL function, IFNULL function, and ISNULL function?**  
These three functions work in the same way. These functions are used to replace NULL value with another value. Oracle developers use NVL function, MySQL developers use IFNULL function and SQL Server developers use ISNULL function.  
Assume, some of the values in a column are NULL.  
If you run below statement, you will get result as NULL



|  |  |
| --- | --- |
| 1 | SELECT col1 \* (col2 + col3) FROM Table1 |

Suppose any of the value in col3 is NULL then as I said your result will be NULL.

To overcome this we use NVL() function, IFNULL() function, ISNULL() Function.

***ORACLE:***



|  |  |
| --- | --- |
| 1 | SELECT col1 \* (col2 + NVL(col3,0)) FROM Table1 |

***MySQL:***



|  |  |
| --- | --- |
| 1 | SELECT col1 \* (col2 + IFNULL(col3,0)) FROM Table1 |

Also, you can use the COALESCE() function



|  |  |
| --- | --- |
| 1 | SELECT col1 \* (col2 + COALESCE(col3,0)) FROM Table1 |

***SQL Server:***



|  |  |
| --- | --- |
| 1 | SELECT col1 \* (col2 + ISNULL(col3,0)) FROM Table1 |

**103. What is Database Testing?**  
It is AKA back-end testing or data testing.  
Database testing involves in verifying the integrity of data in the front end with the data present in the back end. It validates the schema, database tables, columns, indexes, stored procedures, triggers, data duplication, orphan records, junk records. It involves in updating records in a database and verifying the same on the front end.

**104. What is the difference between GUI Testing and Database Testing?**

* GUI Testing is AKA User Interface Testing or Front-end testing  
  Database Testing is AKA back-end testing or data testing.
* GUI Testing deals with all the testable items that are open to the user to interaction such as Menus, Forms etc.  
  Database Testing deals with all the testable items that are generally hidden from the user.
* The tester who is performing GUI Testing doesn’t need to know Structured Query Language  
  The tester who is performing Database Testing needs to know Structured Query Language
* GUI Testing includes invalidating the text boxes, check boxes, buttons, drop-downs, forms etc., majorly the look and feel of the overall application  
  Database Testing involves in verifying the integrity of data in the front end with the data present in the back end. It validates the schema, database tables, columns, indexes, stored procedures, triggers, data duplication, orphan records, junk records. It involves in updating records in a database and verifying the same on the front end.

**SSIS Interview Questions and Answers for Experienced and Freshers**

Here we are publishing series of posts on SSIS Interview questions with answers for experienced and freshers . Below is the series 1.

**Q. Define SSIS?**

**Ans:**

SQL Server Integration Services — commonly known as SSIS is the new platform that was introduced in SQL Server 2005, for data transformation and data integration solutions. This replaced the DTS in SQL Server 2000.

**Q. Name a few SSIS components?**

**Ans**:

* Integration Services Projects
* Integration Services Packages
* Control Flow Elements
* Data Flow Elements
* Integration Services Connections
* Integration Services Variables
* Integration Services Event Handlers
* Integration Services Log Providers

**Q. What is a project and Package in SSIS?**

**Ans:**

Project is a container for developing packages. Package is nothing but an object. It implements the functionality of ETL — Extract, Transform and Load — data.

**Q. What are the 4 elements (tabs) that you see on a default package designer in BIDS?**

**Ans:**

Control Flow, Data Flow, event Handler and package explorer. (Parameters – 2012 Data Tools)

**Q.** **What is a Control flow and Data Flow elements in SSIS?**

**Ans:**

**Control Flow:**

Control flow element is one that performs any function or provides structure or control the flow of the elements. There must be at least one control flow element in the SSIS package. In SSIS a workflow is called a control-flow. A control-flow links together our modular data-flows as a series of operations in order to achieve a desired result.

A control flow consists of one or more tasks and containers that execute when the package runs. To control order or define the conditions for running the next task or container in the package control flow

**Data Flow:**

All ETL tasks related to data are done by data flow elements. It is not necessary to have a data flow element in the SSIS package. A data flow consists of the sources and destinations that extract and load data, the transformations that modify and extend data, and the paths that link sources, transformations, and destinations. Before you can add a data flow to a package, the package control flow must include a Data Flow task. The Data Flow task is the executable within the SSIS package that creates, orders, and runs the data flow. A separate instance of the data flow engine is opened for each Data Flow task in a package.

**Q. What are the 3 different types of control flow elements in SSIS?**

**Ans:**

* Structures provided by Containers
* Functionality provided by Tasks
* Precedence constraints that connect the executables, containers, and tasks into an ordered control flow.

**Q. What are the 3 data flow components in SSIS?**

**Ans:**

* Source
* Transformation
* Destination

**Q. What are connections and connection managers in SSIS?**

**Ans:**

Connection as its name suggests is a component to connect to any source or destination from SSIS — like a sql server or flat file or lot of other options that SSIS provides. Connection manager is a logical representation of a connection.

**Q. What is the use of Check Points in SSIS?**

**Ans:**

SSIS provides a Checkpoint capability which allows a package to restart at the point of failure.

**Q. What are the command line tools to execute SQL Server Integration Services packages?**

**Ans:**

**DTSEXECUI –** When this command line tool is run a user interface is loaded in order to configure each of the applicable parameters to execute an SSIS package.

**DTEXEC –** This is a pure command line tool where all of the needed switches must be passed into the command for successful execution of the SSIS package.

**Q. Can you explain the SQL Server Integration Services functionality in Management Studio?**

**Ans:**

You have the ability to do the following:

* Login to the SQL Server Integration Services instance
* View the SSIS log
* View the packages that are currently running on that instance
* Browse the packages stored in MSDB or the file system
* Import or export packages
* Delete packages
* Run packages

**Q. Can you name some of the core SSIS components in the Business Intelligence Development Studio you work with on a regular** **basis** **when** **building an SSIS package?**

**Ans:**

* Connection Managers
* Control Flow
* Data Flow
* Event Handlers
* Variables window
* Toolbox window
* Output window
* Logging
* Package Configurations

**Q. Name Transformations available in SSIS?**

**Ans:**

**DATACONVERSION:** Converts columns data types from one to another type. It stands for Explicit Column Conversion.

**DATAMININGQUERY:** Used to perform data mining query against analysis services and manage Predictions Graphs and Controls.

**DERIVEDCOLUMN:** Create a new (computed) column from given expressions.

**EXPORTCOLUMN:** Used to export a Image specific column from the database to a flat file.

**FUZZYGROUPING:** Used for data cleansing by finding rows that are likely duplicates.

**FUZZYLOOKUP:** Used for Pattern Matching and Ranking based on fuzzy logic.

**AGGREGATE:** It applies aggregate functions to Record Sets to produce new output records from aggregated values.

**AUDIT:** Adds Package and Task level Metadata: such as Machine Name, Execution Instance, Package Name, Package ID, etc..

**CHARACTERMAP:** Performs SQL Server column level string operations such as changing data from lower case to upper case.

**MULTICAST:** Sends a copy of supplied Data Source onto multiple Destinations.

**CONDITIONALSPLIT:** Separates available input into separate output pipelines based on Boolean Expressions configured for each output.

**COPYCOLUMN:** Add a copy of column to the output we can later transform the copy keeping the original for auditing.

**IMPORTCOLUMN:** Reads image specific column from database onto a flat file.

**LOOKUP:** Performs the lookup (searching) of a given reference object set to a data source. It is used for exact matches only.

**MERGE:** Merges two sorted data sets into a single data set into a single data flow.

**MERGEJOIN:** Merges two data sets into a single dataset using a join junction.

**ROWCOUNT:** Stores the resulting row count from the data flow / transformation into a variable.

**ROWSAMPLING:** Captures sample data by using a row count of the total rows in dataflow specified by rows or percentage.

**UNIONALL:** Merge multiple data sets into a single dataset.

**PIVOT:** Used for Normalization of data sources to reduce anomalies by converting rows into columns

**UNPIVOT:** Used for de-normalizing the data structure by converts columns into rows in case of building Data Warehouses.

**SSIS Interview Questions and Answers for Experienced and Fresher’s**

**Here we are publishing series of posts on SSIS Interview questions and answers Part 2 for experienced and freshers** **. Below is the series 2.**

**Q. What is a breakpoint in SSIS?**

**Ans:**

A breakpoint is a stopping point in the code. The breakpoint can give the Developer\DBA an opportunity to review the status of the data, variables and the overall status of the SSIS package.  
Breakpoints are setup in BIDS. In BIDS, navigate to the control flow interface. Right click on the object where you want to set the breakpoint and select the ‘Edit Breakpoints…’ option.

**Q. Can you name 5 or more of the native SSIS connection managers?**

**Ans:**

* **OLEDB connection** – Used to connect to any data source requiring an OLEDB connection (i.e., SQL Server)
* **Flat file connection –** Used to make a connection to a single file in the File System. Required for reading information from a File System flat file
* **ADO.Net connection –** Uses the .Net Provider to make a connection to SQL Server 2005 or other connection exposed through managed code (like C#) in a custom task
* **Analysis Services connection –** Used to make a connection to an Analysis Services database or project. Required for the Analysis Services DDL Task and Analysis Services Processing Task
* **File connection –** Used to reference a file or folder. The options are to either use or create a file or folder
* **Excel**
* **FTP**
* **HTTP**
* **MSMQ**
* **SMO**
* **SMTP**
* **SQL Mobile**
* **WMI**

**Q. How do you eliminate quotes from being uploaded from a flat file to SQL Server?**   
**Ans:**

In the SSIS package on the Flat File Connection Manager Editor, enter quotes into the Text qualifier field then preview the data to ensure the quotes are not included.

**Q. Can you name 5 or more of the main SSIS tool box widgets and their functionality?**

**Ans:**

* ActiveX Script Task
* Analysis Service Processing Task
* Analysis Services Execute DDL Task
* Backup Database Task
* Bulk Insert Task
* CDC Control Task
* Check Database Integrity Task
* Data Flow Task
* Data Mining Query Task
* Data Profiling Task
* Execute DTS 2000 Package Task – Till 2008
* Execute Package Task
* Execute Process Task
* Execute SQL Server Agent Job Task
* Execute SQL Task
* Execute T-SQL Statement Task
* Expression Task
* File System Task
* For Loop Container
* Foreach Loop Container
* FTP Task
* History Cleanup Task
* Maintenance Cleanup Task
* Message Queue Task
* Notify operator Task
* Rebuild Index Task
* Reorganize Index Task
* Script Task
* Send Mail Task
* Sequence Container
* Shrink Datbase Task
* Transfer Database Task
* Transfer error message
* Transfer Jobs Task
* Transfer Logins Task
* Transfer Mastor Stored Procedures Task
* Transfer SQL Server Object Task
* Update Ststistics Task
* Web Service Task
* WMI Datareader Task
* WMI Event Watcher Task
* XML Task

**Q. Can you explain one approach to deploy an SSIS package?**

**Ans:**

* One option is to build a deployment manifest file in BIDS, then copy the directory to the applicable SQL Server then work through the steps of the package installation wizard
* A second option is using the dtutil utility to copy, paste, rename, delete an SSIS Package
* A third option is to login to SQL Server Integration Services via SQL Server Management Studio then navigate to the ‘Stored Packages’ folder then right click on the one of the children folders or an SSIS package to access the ‘Import Packages…’ or ‘Export Packages…’option.
* A fourth option in BIDS is to navigate to File | Save Copy of Package and complete the interface.

**Q. Can you explain how to setup a checkpoint file in SSIS?**

**Ans:**   
The following items need to be configured on the properties tab for SSIS package:

**CheckpointFileName** – Specify the full path to the Checkpoint file that the package uses to save the value of package variables and log completed tasks. Rather than using a hard-coded path, it’s a good idea to use an expression that concatenates a path defined in a package variable and the package name.

**CheckpointUsage** – Determines if/how checkpoints are used. Choose from these options: Never (default), If Exists, or Always. Never indicates that you are not using Checkpoints. “If Exists” is the typical setting and implements the restart at the point of failure behavior. If a Checkpoint file is found it is used to restore package variable values and restart at the point of failure. If a Checkpoint file is not found the package starts execution with the first task. The Always choice raises an error if the Checkpoint file does not exist.

**SaveCheckpoints** – Choose from these options: True or False (default). You must select True to implement the Checkpoint behavior.

**Q. Would you recommend using “Check Points” in SSIS?**

**Ans:**

As per my experience I could say “NO” as there are compatibility issues with various options hence using checkpoints may give unpredictable results. Checkpoints doesn’t work properly when a SSIS package contains

* Complex logic
* Iterations/Loops
* Transactions Enabled
* “Object” type variables
* Parallel execution

Checkpoints works fine when the package is having straightforward control flow with a single thread.

**Q. Can you explain different options for dynamic configurations in SSIS?**

**Ans:**

* Use an XML file
* Use custom variables
* Use a database per environment with the variables
* Use a centralized database with all variables

**Q. How do you upgrade an SSIS Package?**

**Ans:**

Depending on the complexity of the package, one or two techniques are typically used:

* Recode the package based on the functionality in SQL Server DTS.
* Use the Migrate DTS 2000 Package wizard in BIDS and then recode any portion of the package that is not accurate

**Q. Can you name five of the Perfmon counters for SSIS and the value they provide?**

**Ans:**

**SQLServer: SSIS Service**

**SSIS Package Instances** – Total number of simultaneous SSIS Packages running

**SQLServer: SSIS Pipeline**

**BLOB bytes read** – Total bytes read from binary large objects during the monitoring period.

**BLOB bytes written** – Total bytes written to binary large objects during the monitoring period.

**BLOB files in use** – Number of binary large objects files used during the data flow task during the monitoring period.

**Buffer memory:** The amount of physical or virtual memory used by the data flow task during the monitoring period.

**Buffers in use** – The number of buffers in use during the data flow task during the monitoring period.

**Buffers spooled** – The number of buffers written to disk during the data flow task during the monitoring period.

**Flat buffer memory** – The total number of blocks of memory in use by the data flow task during the monitoring period.

**Flat buffers in use** – The number of blocks of memory in use by the data flow task at a point in time.

**Private buffer memory** – The total amount of physical or virtual memory used by data transformation tasks in the data flow engine during the monitoring period.

**Private buffers in use** – The number of blocks of memory in use by the transformations in the data flow task at a point in time.

**Rows read** – Total number of input rows in use by the data flow task at a point in time.

**Rows written** – Total number of output rows in use by the data flow task at a point in time.

**Q. How do you handle errors in ssis?**

**Ans:**

When a data flow component applies a transformation to column data, extracts data from sources, or loads data into destinations, errors can occur. Errors frequently occur because of unexpected data values.

Errors typically fall into one the following categories:

**Data conversion errors:** occurs if a conversion results in loss of significant digits, the loss of insignificant digits, and the truncation of strings. Data conversion errors also occur if the requested conversion is not supported.

**Expression evaluation errors**: occurs if expressions that are evaluated at run time perform invalid operations or become syntactically incorrect because of missing or incorrect data values.

**Lookup errors**: occurs if a lookup operation fails to locate a match in the lookup table.

Many data flow components support error outputs, which let you control how the component handles row-level errors in both incoming and outgoing data. You specify how the component behaves when truncation or an error occurs by setting options on individual columns in the input or output.

**Q. How do you do Logging in SSIS?**

**Ans:**

* SSIS includes logging features that write log entries when run-time events occur and can also write custom messages.
* The Integration Services log providers can write log entries to text files, SQL Server Profiler, SQL Server, Windows Event Log, or XML files.
* Logs are associated with packages and are configured at the package level. Each task or container in a package can log information to any package log. The tasks and containers in a package can be enabled for logging even if the package itself is not.

To enable logging in a package:

* In Business Intelligence Development Studio, open the Integration Services project that contains the package you want.
* On the SSIS menu, click Logging.
* Select a log provider in the Provider type list, and then click Add.

**Q. Demonstrate how you would suggest using configuration files in packages. Would you consider it a best practice to create a configuration file for each connection manager or one for the entire package?**

**Ans:**

There should be a single configuration file for each connection manager in your packages that stores their connection string information. So if you have 6 connection managers then you have 6 config files. You can use the same config file across all your packages that use the same connections.

If you have a single config file that stores all your connection managers then all your packages must have contain the connection managers that are stored in that config file. This means you may have to put connection managers in your package that you don’t even need.

**Q. Demonstrate how checkpoints work in a package.**

**Ans:**

When checkpoints are enabled on a package if the package fails it will save the point at which the package fails. This way you can correct the problem then rerun from the point that it failed instead of rerunning the entire package. The obvious benefit to this is if you load a million record file just before the package fails you don’t have to load it again.

**Q. Demonstrate how transactions work in a package.**

**Ans:**

If transactions are enabled on your package and tasks then when the package fails it will rollback everything that occurred during the package. First make sure MSDTC (Microsoft Distributed Transaction Coordinator) is enabled in the Control Panel -> Administrative Tools -> Component Services. Transactions must be enabled not only on the package level but also on each task you want included as part of the transaction. To have the entire package in a transaction set Transaction Option at the package level to Required and each task to Supported.

**Q. If you have a package that runs fine in Business Intelligence Development Studio (BIDS) but fails when running from a SQL Agent Job what would be your first guess on what the problem is?**

**Ans:**

The account that runs SQL Agent Jobs likely doesn’t have the needed permissions for one of the connections in your package. Either elevate the account permissions or create a proxy account.

To create a proxy account you need to first create new credentials with the appropriate permissions. Next assign those credentials to a proxy account. When you run the job now you will select Run As the newly created proxy account.

**Q. What techniques would you consider to add auditing to your packages? You’re required to log when a package fails and how many rows were extracted and loaded in your sources and destinations.**

**Ans:**

I like to create a database that is designated for package auditing. Track row counts coming from a source and which actually make it to a destination. Row counts and package execution should be all in one location and then optionally report off that database.

There are also third party tools that can accomplish this for you (Pragmatic Works BI xPress).

**Q. Demonstrate or whiteboard techniques you would use to for CDC (Change Data Capture)? Tell how you would write a package that loads data but first detects if the data already exists, exists but has changes, or is brand new data for a destination.**

**Ans:**

For small amounts of data I may use the Slowly Changing Dimension. More often than not the data is too large to use in such a slow transform.

I prefer to do a lookup on the key of the target table and rows that don’t match are obviously new rows that can be inserted. If they do match it’s possible they are updates or duplicates. Determine this by using a conditional split comparing rows from the target to incoming rows. Send updates to a staging table that can then be updated in an Execute SQL Task.

Explain that putting updates in a staging table instead of updating using the OLE DB Command is much better for performance because the Execute SQL Task performs a bulk operation.

**Q. Explain what breakpoints are and how you would use them.**

**Ans:**  
Breakpoints put pauses in your package. It’s a great tool for debugging a package because you can place a breakpoint on a task and it will pause the package based on execution events.

A reason in which I have used breakpoints is when I have a looping container and I want to see how my variables are changed by the loop. I would place a watch window on the package and type the variable name in. Set a break point on the container the stop after each iteration of the loop.

**Q. What are the main components involved in SSIS?**

**Ans:**

* SSIS is not improved version of DTS
* SSIS is completely redesigned and build from ground up using .NET code.
* SSIS is mainly divided into two parts.
* Data Transformation Pipeline (DTP) – Data Flow
* Data Transformation Runtime (DTR) – Control Flow
* In SQL SERVER 7 / 2000 the data flow is stronger than control flow but in SSIS both are in the same level

**Q. What is the work of DTP Engine?**

**Ans:**

* DTP consists of DTP Engine and DTP Object model
* DTP uses Data Adapters to connect source and destination
* DTP engine uses DTP Object Model which is nothing but an API.
* SSIS comes with adapters for SQL Server databases, XML, flat files, and other OLE DB–compliant data sources
* The job of the data adapters is to make connections to the data’s source and destination endpoints
* The job of the transformations is to move and optionally manipulate the data as it’s moved between the source and destination endpoints.

**Q. How the DTR works in SSIS?**

**Ans:**

* The DTR consists of the DTR engine and the DTR components.
* DTR components are objects that enable you to govern the execution of SSIS packages.
* The primary DTR components are packages, containers, and tasks.
* DTR engine stores package layout; runs packages; and provides debugging, logging, and event handling services.
* The DTR is accessed using the DTR object framework. The DTR run-time object framework is the API that supports the Integration Services Import/Export Wizard and the Integration Services Designer in addition to the command-line dtexec tool.

**Q. Can you explain the SSIS Architecture?**

**Ans:**

**Runtime engine**

The Integration Services runtime saves the layout of packages, runs packages, and provides support for logging, breakpoints, configuration, connections, and transactions.

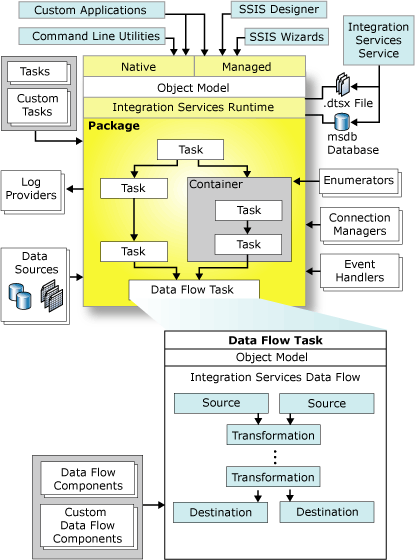
**API or object model**

The Integration Services object model includes managed application programming interfaces (API) for creating custom components for use in packages, or custom applications that create, load, run, and manage packages. Developer can write custom applications or custom tasks or transformations by using any common language runtime (CLR) compliant language.

**Integration Services service:** It is a Windows service, monitors running SSIS packages and manages the storage of packages.

**Data flow:** It contains a data flow engine that manages the data flow components. There are 3 types of

Data Flow components – Source components (which extracts the data from a system), Transformation components (performs transformations, modifications onto the extracted data) and Load components (which simply performs the data loading tasks into the destination systems). Besides the available data flow components, we can write our own custom data flow components to accomplish any custom requirements.



**SSIS – Performance Tuning**

**SSIS Interview Questions and Answers Part 3**

Here we are publishing series of posts on SSIS Interview questions and answers Part 3 for experienced and freshers. Below is the series 3.

**Q. How to quickly load data into sql server table?**

**Ans:**

Fast Load option: This option is not set by default so most developers know this answer as otherwise the load is very slow.

**Q. What are the fast load options available in SSIS?**

**Ans:**

The OLE DB Destination provides more than one way to load data in the destination (5 types of Data Access Mode). Use Fast Load option while loading data into the destination.

* **Data Access Mode** – It allows to define the method to upload data into the destination. The fast load option will use **BULK INSERT** statement instead of **INSERT** statement. If the fast load option is not selected then by default INSERT is used.
* **Keep Identity** – If selected, the identity values of source are preserved and the same are uploaded into the destination table. Else destination table will create its own identity values if there is any column of identity type.
* **Keep Nulls** – If selected, the null values of the source are preserved and are uploaded into the destination table. Else if any column has default constraint defined at destination table and NULL value is coming from the source for that column then in that case, default value will be inserted into the destination table.
* **Table Lock** – If selected, the TABLOCK is acquired on the table during data upload. It is the recommended option if table is not being used by any other application at the time of data upload as it removes the overhead of lock escalation.
* **Check Constraints** – Check constraints will always check for any constraint for the data that is coming through pipeline. It is preferable to uncheck this option if constraint checking is not required. This will reduce the overhead for the pipeline engine.
* **Rows per batch** – **RowsPerBatch** is the number of rows you would want in One Buffer. SSIS automatically sets this property based on the **RowSize** and **MaxBufferRows** property. The number of rows coming from the pipeline per batch can be defined by user. The default value is **-1** if it is kept blank. You can specify the no. of rows as a positive integer (**N**) so that the records will come as small segments or batches, each segment containing **N** no. of rows.
* **Maximum insert commit size** – You can specify the batch size that the OLE DB destination tries to commit during fast load operations; it actually splits up chunks of data as they are inserted into your destination. If you provide a value for this property, the destination commits rows in batches that are the smaller from either (a) the Maximum insert commit size, or (b) the remaining rows in the buffer that is currently being processed.
* **Network limitations:** You can transfer data as fast as your network supports. But use them efficiently; you can customize SSIS to use the maximum bandwidth of your network. You can set the **Packet Size** property of the connection manager to an integer value that suits you. The max value that you can insert is **32767.**

**Q. What are the lookup cache modes available and how to use them?**

**Ans:**

In 2008 we have three different cache modes for lookup transformations.

* Full Cache – Default
* Partial Cache
* No Cache

**Full Cache:**

The database is queried once during the pre-execute phase of the data flow. The entire reference set is pulled into memory. This approach uses the most memory, and adds additional startup time for your data flow. Lookup will not swap memory out to disk, so your data flow will fail if you run out of memory.

* **When to use Full cache mode**
* When you’re accessing a large portion of your reference set
* When you have a small reference table
* When your database is remote or under heavy load, and you want to reduce the number of queries sent to the server

**Partial Cache:**

In this mode, the lookup cache starts off empty at the beginning of the data flow. When a new row comes in, the lookup transform checks its cache for the matching values. If no match is found, it queries the database. If the match is found at the database, the values are cached so they can be used the next time a matching row comes in.

In 2008 there is a new **Miss Cache** feature that allows you to allocate a certain percentage of your cache to remembering rows that had no match in the database. This is useful in a lot of situations, as it prevents the transform from querying the database multiple times for values that don’t exist

* **When to use this cache mode**
* When you’re processing a small number of rows and it’s not worth the time to charge the full cache
* When you have a large reference table
* When your data flow is adding new rows to your reference table

**No Cache:**

As the name implies, in this mode the lookup transform doesn’t maintain a lookup cache (actually, not quite true – we keep the last match around, as the memory has already been allocated). In most situations, this means that you’ll be hitting the database for every row.

* **When to use this cache mode**
* When you’re processing a small number of rows
* When you have non-repeating lookup indexes
* When your reference table is changing (inserts, updates, deletes)
* When you have severe memory limitations

**Q. What are the different types of Transformations in SSIS?**

**Ans:**

**Non-Blocking –** No blocking

**Partial Blocking –** The downstream transformations wait for certain periods, it follows start then stop and start over technique

**Full Blocking:** The downstream has to be waiting till the data has been released from the upstream transformation.

**Non-blocking transformations**

* Audit
* Cache Transform
* Character Map
* Conditional Split
* Copy Column
* Data Conversion
* Derived Column
* Export Column
* Import Column
* Lookup
* Multicast
* OLE DB Command
* Percentage Sampling
* Script Component
* Slowly Changing Dimension

**Partial blocking transformations**

* Data Mining
* Merge
* Merge Join
* Pivot
* Unpivot
* Term Lookup

**Fully Blocking Transformations**

* Aggregate
* Fuzzy grouping
* Fuzzy lookup
* Row Sampling
* Sort
* Term Extraction

If you clearly observe Sort is a fully blocking transformation, so it’s better to sort your data using the SQL command in OLE DB Source instead of using sort transformation. Merge transform requires Sort but not Union All, so use Union All wherever possible.

**Q. Consider a scenario where I am using “Sort” transformation and my requirement is to after sort operation completed I have to remove all duplicate records. Duplicate records are defined based on sort values for example I am sorting result set based on three columns, when these 3 columns are having same values those rows are considered as duplicates. Now my question is which transformation we have to use to ignore all these duplicate records?**

**Ans:**

We need not use any specific transformation to remove duplicate records based on sort columns. There is a feature available at “Sort” transformation itself. We can find an option “Remove duplicate sort values” at the bottom left corner of SORT transformation editor. Just check that box.

**Q. How to avoid the sort transformation in SSIS?**

Ans:

Input datasets are to be in sorted order while dealing with the “Merge” or “Merge Join” transformations. To avoid the sort transformation we can directly use a “Query” with order by clause at data source. But remember we can do that when data source is OLEDB or Excel. When it comes to flat file we don’t have a choice but choose the best way to implement sort transformation.

For example if there is an aggregate required then apply aggregate before applying the sort transformation. If possible load flat file data into stage tables, apply sort at database level and load them at destination. So that we should have two data flows one is to load data from flat files to stage tables and other is to loading data into destination from stage tables hence we can use parallelism property.

**Q.** **How an SSIS package or a data flow knows that the input dataset / source dataset is in sorted order?**

**Ans:**

If the source dataset is in sorted order or we are using a query with order by at source we have to explicitly mention this information at “OLEDB” source.

There are two properties that we need to change at OLEDB source.

1. Open OLEDB source advanced editor

2. Goto tab “Input and Output Properties”

3. Select “OLEDB source OUTPUT”

4. In the properties select the value “True” for the property “IsSorted”

5. Expand Output Column list and select the column name and set “SortKeyPosition” value to one.

6. Repeat the step 5 for all columns in order by cluase by giving appropriate priority

**Q. What data providers supported for OLEDB connection manager for cache option when lookup transformation?**

**Ans:**

SQL Server

Oracle

DB2

**Q. From your customer side one of the architect asked you the below information. “I just wanted to know how many number of execution trees are being created for SSIS package which loads data on daily basis.”**

**How do we know this information?**

**Ans:**  
We can actually use custom log events to capture this information.

The log entry “PipelineExecutionTrees” helps us know about the execution trees created at run time. It includes lots of info for example number of rows stored in a buffer while executing a transformation etc.

For more info please have a look at below link

<http://msdn.microsoft.com/en-us/library/ms345174.aspx>

**Q. Do you know when an execution tree created and when it ends in a dataflow? Simply what is the scope of an execution tree?**

**Ans:**

The work to be done in the data flow task is divided into multiple chunks, which are called execution units, by the dataflow pipeline engine. Each represents a group of transformations. The individual execution unit is called an execution tree, which can be executed by separate thread along with other execution trees in a parallel manner. The memory structure is also called a data buffer, which gets created by the data flow pipeline engine and has the scope of each individual execution tree. An execution tree normally starts at either the source or an asynchronous transformation and ends at the first asynchronous transformation or a destination. During execution of the execution tree, the source reads the data, then stores the data to a buffer, executes the transformation in the buffer and passes the buffer to the next execution tree in the path by passing the pointers to the buffers.  
​​

**Q. While running SSIS package, after 15 min of execution it went to hung state. How you troubleshoot?**

**Ans:**

There are three common reasons that hold / hung the SSIS execution.

1. Resource Bottleneck: Memory / CPU / IO / Network
2. Blocking / Deadlock: Blocking happens at database level or In accessing a file or reading writing variables from script task.
3. Poor Performance query: If SSIS stops at Execute SQL Task look for query using inside the task and tune it.

Looking through above aspects one can identify the issue, based on that we can provide the resolution. If everything looks good but still SSIS is in hung state then check the latest service pack is applied if that’s also passed collect the hung dump file using ADPlus and contact Microsoft support center.

**Q. SSIS 2008 uses all available RAM, and after package completes Memory is not released?**

**Ans:**

This is not actually a problem. You have allowed SQL Server to use x amount of memory, so it does. SQL Server takes that memory as required, up to the limit set, but it does not release it. It can respond to request from OS, again read up on the fine details, but by default once it has got hold of some memory it will keep it even if it is not using it currently. The simple reason is that finding and taking hold of memory is quite expensive to do, so once it has it it keeps it and then any subsequent operations that need memory will have it available much faster. This makes perfect sense when you remember that SQL Server is a service application and more often than not runs on a dedicated machine.

**Q. What is the property “RunInOptimized”? How to set this property?**

**Ans:**

If this property is set to true then the SSIS engine ignore the unused/unmapped columns. Means it does not allocate memory to store data for those columns. At the compilation phase itself SSIS engine identifies what are the columns from source are using across the package, if it finds any columns are neither using nor mapping to destination, it simply ignores all those columns.

We can set this property at two levels “Project Level” and “Package Level”.

Project Level: From project properties → Debugging → RunIn\*\*\*\*\*. By default “FALSE”

Package Level: Can find in DataFlow properties. By default “TRUE”

**Q. Does using “RowCount” transformation affects the package performance?**

**Ans:**

Rowcount component is a synchronous component and it doesn’t actually do anything particularly resource intensive means the performance degradation of your package should be negligible.

We do use this component to capture the number of inserts, deletes and updates from each data-flow and then using “OnPost Execute” event this information would be written to a SQL Server table.

**Q. A SSIS 2008 package has been crashed due to low memory. How to resolve low memory issues with SSIS package?**

**Ans:**

1. Add more memory to the physical machine

2. Run SSIS package on a computer that is not running an instance of SQL Server

3. When SSIS and SQL instance on the same machine, balance the memory allocated to SQL Server instance using “MAX Server Memory” option.

4. Run SSIS package components in series instead of parallel

**Q. How to identify the SSIS processes?**

**Ans:**

SSIS run-time processes include the DTExec.exe process and the DTSHost.exe process.

**Q. How to enable containers continue to run even a task failed inside the container? Suppose you have an application, where we need to loop through some log table based on the IDs & load data into the destination. Now, in this scenario there might be the situation where some of the tasks in foreach loop container may fail. But your requirement is even though the inner tasks fail we should process the other sources which are available with us.**

Ans:

We can do this by updating the propagation property of a task / container to “False”. It means that the loop or sequence container ignores the failure of an internal task.

Assume we have designed a foreach loop container with a dataflow task. As per our requirement DFT is loading 100 files into database if DFT is failed to load 47th file it should skip the error and should continue to load from 48th file.

Steps to accomplish this are:

Select the Data Flow Task and goto eventhandler

Enable the OnError Event handler.

In the Event Handler tab, click on the “Show System Variables”.

Now select the “Propogate” property & change its value to “False”.

This will ensure that the parent control i.e. ForEach loop will not know about the error in the child task.

If incase the foreach loop container is having more than one task, instead of setting the property to all these tasks, add all these tasks to sequence container and change the “Propagate” property of sequence container.

**Note:** When this kind of situation comes to the single task instead of a loop we can actually use a property called “ForceExecutionValue” to “True” and give the value to “Force**d**ExecutionValue”“1”. This means that irrespective of execution result ssis engine forces the outcome to success.

**Q. What is ForceExecution property in SSIS component properties?**

**Ans:**

ForceExecution is a property of Controlflow elements in SSIS. If it is enabled to any of the element then ssis engine follows the execution result as per the given parameters. In other words to control the execution result of any control flow element we can use this property.

ForceExecutionValue: True or False

ForcedExecutionType: <Datatype>

ForcedExecutionValue: <Value>, we usually gives as 1 to make sute its true.

**Q. How to improve the performance of a SSIS package?**

**Ans:**

**1- Utilize parallelism:** It is easy to utilize parallelism in SSIS. All you need to do is to recognize which Data Flow Tasks (DFTs) could be started at the same time and set the control flow constraints of your package in the way that they all can run simultaneously.

**2- Synchronous vs. Asynchronous components:** A synchronous transformation of SSIS takes a buffer, processes the buffer, and passes the result through without waiting for the next buffer to come in. On the other hand, an asynchronous transformation needs to process all its input data to be able to give out any output. This can cause serious performance issues when the size of the input data to the asynchronies transformation is too big to fit into memory and needs to be transferred to HDD at multiple stages.

**3- Execution tree:** An execution tree starts where a buffer starts and ends where the same buffer ends. These execution trees specify how buffers and threads are allocated in the package. Each tree creates a new buffer and may execute on a different thread. When a new buffer is created such as when a partially blocking or blocking transformation is added to the pipeline, additional memory is required to handle the data transformation; however, it is important to note that each new tree may also give you an additional worker thread.

**4-OLE DB Command transformation:** OLE DB Command is a row-by-row transformation, meaning that it runs the command in it on each one of its input rows. This make sit to be damn too slow when the number of the rows goes up. The solution for boosting performance is to stage data into a temporary table and use Execute SQL Task outside that DFT.

**5-SQL Server Destination vs. OLE DB Destination:** There is multiple reason why to use OLE DB Destination and not use SQL Server Destination:

* OLE DB Destination is mostly faster,
* OLE DB Destination is a lot clearer when it fails (The error message is more helpful),
* SQL Server Destination works only when SSIS is installed on the destination server.

**6- Change Data Capture (CDC):** Try to reduce the amount of data to be transferred to the maximum level you can, and do it as close to the source as you can. A Modified On column on the source table(s) helps a lot in this case.

**7- Slowly Changing Dimension (SCD) transformation:** There is only one advice about SSIS’s Slowly Changing Dimension transformation, and that is get rid of it! The reasons are:

* It doesn’t use any cached data, and goes to the data source every single time it is called,
* It uses many OLE DB Command transformations,
* Fast Data Load is off by default on its OLE DB Destination.

**8. Choose the best way in designing Data flow between SQL and SSIS:** Remember SSIS is good at Row by Row operations where AS SQL is not. So depends on the situation design data flow using DFT components instead of executing a query using “Execute SQL Task”.

**9.** Use queries for selecting data rather than selecting a table and checking off the columns you want. This will reduce the initial record set before SSIS gets it rather than ignoring the fields

**10.** Carefully deal with your connections. By default, your connection manager will connect to the database as many times as it wants to. You can set the **RetainSameConnection** property so it will only connect once. This can allow you to manage transactions using an ExecuteSQL task and BEGIN TRAN / COMMIT TRAN statements avoiding the overhead of DTC.

**11.** While running the package with in BIDS ensure you set the package to run in optimized mode.

**12.** While loading data into destination tables it’s helpful to use the “Fast Load option”.

**13.** Wherever possible Consider aggregating and (un)pivotting in SQL Server instead doing it in SSIS package – SQL Server outperforms Integration Services in these tasks;

**14**. **Avoid manipulating large datasets using T-SQL** statements. All T-SQL statements cause changed data to write out to the transaction log *even if you use Simple Recovery Model*.

**15**. For large datasets, **do data sorts at the source** if possible.

**16**. **Use the SQL Server Destination** if you know your package is going to run on the destination server, since it offers roughly 15% performance increase over OLE DB because it shares memory with SQL Server.

**17**. **Increase the network packet size to 32767 on your database connection managers**. This allows large volumes of data to move faster from the source servers.

**18**. If using Lookup transforms, **experiment with cache sizes** – between using a Cache connection or Full Cache mode for smaller lookup datasets, and Partial / No Cache for larger datasets. This can free up much needed RAM.

**19.** Make sure “Lock Options” is using while loading very large datasets as bulk insert happens when it satisfies the below conditions.

a) Destination table is empty

b) Destination database recovery model is either simple or bulk insert

c) When table lock option specified

**20.** **Experiment with the DefaultBufferSize and DefaulBufferMaxRows** properties. You’ll need to monitor your package’s “Buffers Spooled” performance counter using Perfmon.exe, and adjust the buffer sizes *upwards* until you see buffers being spooled (paged to disk), then back off a little.

**21.** Do all setbased, aggregations and sort operations at source or destination using T-SQL.

**22.** If possible always use “NOLOCK” at source and “LOCK” at destination.

**23.** While loading to data warehouses try to disable the indexes while loading.

**Q. Can you explain the settings “Rows Per Batch” and “Maximum Insert Commit Size”?**

**Ans:**

These options are available at “OLEDB destination” in DFT.

**Rows per batch** – The default value for this setting is -1 which specifies all incoming rows will be treated as a single batch. You can change this default behaviour and break all incoming rows into multiple batches. The allowed value is only positive integer which specifies the maximum number of rows in a batch.

**Maximum insert commit size** – The default value for this setting is ‘2147483647’ (largest value for 4 byte integer type) which specifies all incoming rows will be committed once on successful completion. You can specify a positive value for this setting to indicate that commit will be done for those number of records. You might be wondering, changing the default value for this setting will put overhead on the dataflow engine to commit several times. Yes that is true, but at the same time it will release the pressure on the transaction log and tempdb to grow tremendously specifically during high volume data transfers.

**Q. Can you explain the DFT properties “DefaultBufferMaxRows” and “DefaultBufferMaxSize”?**

**Ans:**

The data flow task in SSIS (SQL Server Integration Services) sends data in series of buffers. How much data does one buffer hold? This is bounded by DefaultBufferMaxRows and DefaultBufferMaxSize, two Data Flow properties. They have default values of 10,000 and 10,485,760 (10 MB), respectively. That means, one buffer will contain either 10,000 rows or 10 MB of data, whichever is less.

You can adjust these two properties based on your scenario. Setting them to a higher value can boost performance, but only as long as all buffers fit in memory. In other words, no swapping please!

**Q. How can we connect to Oracle, DB2 and MySQL from SSIS?**

**Ans:**

**Oracle:**

Native OLEDB\Microsoft OLEDB Provider for Oracle

Native .Net providers\ or

.Net providers for OLEDB\

**MySQL:**

.Net Providers \ MySQL Data Provider Or

.Net Providers \ ODBC

**DB2:**

Native OLEDB\Microsoft OLEDB Provider for DB2

Native .Net providers\ ,

.Net providers\ ODBC OR

.Net providers for OLEDB\

**Q. Can’t we do FastLoad using “ADODotNet Destination”?**

**Ans:**

Yes, there is an option called “Use Bulk insert when possible” that needs to be tick at the time of mapping.

**Q. How to check whether SSIS transformations are using memory or spilling to Disk due to huge loads and asynchronous transformations?**

**Ans:**

A great way to check if your packages are staying within memory is to review the SSIS performance counter Buffers spooled, which has an initial value of 0; above 0 is an indication that the engine has started swapping to disk.

**Q. How to find how much of total memory allocated to SSIS and SQL Server?**

**Ans:**

Below are the performance counters which can help us in finding memory details.

**Process / Private Bytes (DTEXEC.exe)**: The amount of memory currently in use by Integration Services.

**Process / Working Set (DTEXEC.exe)**: The total amount of allocated memory by Integration Services.

**SQL Server: Memory Manager / Total Server Memory**: The total amount of memory allocated by SQL Server. Because SQL Server has another way to allocate memory using the AWE API, this counter is the best indicator of total memory used by SQL Server..

**Memory / Page Reads / sec**: Represents to total memory pressure on the system. If this consistently goes above 500, the system is under memory pressure.

**Q. See there is a scenario: We have a package which has to be open using BIDS / SSDT and has to be modified different elements. But from the location where the SSIS has to be open and modified is not having permissions to access the databases hence all connection managers and other location constraints will fail in validation phase and it takes lot of time to validate all of these connections. Do you have any idea how to control this validation phase?**

**Ans:**

Below are the different methods to switch off the package validation.

**Work OffLine:** There is a option called Work Offline. It doesn’t try to locate/validate packages. Once the package is ready then we have to uncheck the option Work Offline from SSI menu.

**Delay Validation:** Set the values to “True” to skip the validation while opening the package. It only applies for executables / control flow elements including package.

**ValidateExternalMetadata:** Property is set to be “True” for disabling the validation for dataflow components.

**SSIS Interview Questions and Answers for Experienced and Freshers**

Here we are publishing series of posts on SSIS Interview questions and answers Part 4 for experienced and freshers. Below is the series 4.

**Q.** **Difference between Union–all and Merge Join?**

**Ans:**

* Merge transformation can accept only two inputs whereas Union all can take more than two inputs
* Data has to be sorted before Merge Transformation whereas Union all doesn’t have any condition like that.

**Q. What is difference between Multicast and Conditional Split?**

**Ans:**

The Multicast transformation distributes its input to one or more outputs. This transformation is similar to the Conditional Split transformation. Both transformations direct an input to multiple outputs. The difference between the two is that the Multicast transformation directs every row to every output, and the Conditional Split directs a row to a single output

**Q. What is the difference between DTS and SSIS?**

**Ans:**

Well, nothing except both the Microsoft SQL Server Products.

Even though both are the ETL tools, we can differentiate if you are asked observations.

|  |  |  |
| --- | --- | --- |
| **S.no** | **DTS** | **SSIS** |
| 1 | Data Transformation Services | Sql Server Integration Services |
| 2 | Using Activex Script | Using Scripting Language |
| 3 | No Deployment wizard | Deployment wizard |
| 4 | Limited Set of Transformation available | Huge of Transformations available |
| 5 | Not Supporting BI Functionality | Completely supporting end to end process of BI |
| 6 | Single Tasks at a time | Multi Tasks run parallel |
| 7 | It is Un managed script | Managed by CLR |
| 8 | DTS can develop thru Enterprise manager | SSIS can thru Business Intelligence Development Studio (BIDS, nothing but new version of VS IDE) |
| 9 | We can deploy only at local server | It can be deployed using multiple server using BIDS |
| 10 | Designer contains Single Pane | SSIS designer contains 4 design panes: |
|  |  | a) Control Flow |
|  |  | b) Data Flow |
|  |  | c) Event Handlers & |
|  |  | d) Package Explorer. |
| 11 | No Event Hander | Event Handler Available |
| 12 | No Solution Explorer | Solution Explorer is available, with packages, connections and Data Source Views (DSV) |
| 13 | Connection and other values are static, not controlled at runtime. | It can be controlled dynamically using configuration |

**Q. What is the difference between Fuzzy Lookup and Fuzzy Grouping?**

**Ans:**

The Fuzzy Grouping task performs the same operations as the Fuzzy Lookup task but instead of evaluating input records against an outside reference table, the input set becomes the reference. Input records are therefore evaluated against other records in the input set and evaluated for similarity and assigned to a group.

**Q. What’s the difference between Control Flow and Data Flow?**

**Ans:**

**Control Flow:**

* Process Oriented
* Doesn’t manage or pass data between components.
* It functions as a task coordinator
* In control flow tasks requires completion (Success.,failure or completion)
* Synchronous in nature, this means, task requires completion before moving to next task. If the tasks are not connected with each other but still they are synchronous in nature.
* Tasks can be executed both parallel and serially
* Three types of control flow elements in SSIS 2005
* **Containers:**Provides structures in the packages
* **Tasks**: Provides functionality in the packages
* **Precedence Constraints**: Connects containers, executables and tasks into an ordered control flow.
* It is possible to include nested containers as SSIS Architecture supports nesting of the containers. Control flow can include multiple levels of nested containers.

**Data Flow**

* Streaming in nature
* Information oriented
* Passes data between other components
* Transformations work together to manage and process data. This means first set of data from the source may be in the final destination step while at the same time other set of data is still flowing. All the transformations are doing work at the same time.
* Three types of Data Flow components
* Sources: Extracts data from the various sources (Database, Text Files etc)
* Transformations: Cleans, modify, merge and summarizes the data
* Destination: Loads data into destinations like database, files or in memory datasets

**Q. What is difference between For Loop and For Each Loop?**

**Ans:**

A for loop will execute the tasks a specified number of times, in other words 10 times, or 25 times, and the number of times is specified in the definition of the container. You can use a variable to specify what that count is.

A for each loop will execute once for each item in the collection of items that it is looking at. A good example would be if users are putting an Excel file into a directory for import into the DB. You cannot tell ahead of time how many will be in the directory, because a user might be late, or there might be more than one file from a given user. When you define the ForEach container, you would tell it to execute for each \*.xls in the directory and it will then loop through, importing each one individually, regardless of how many files are actually there.

**Q. What is the difference between “OLEDB command” transformation and “OLEDB” destination in dataflow?**

**Ans:**

The OLE DB Command is a pretty simple transformation that’s available within a Data Flow that can run a SQL statement that can insert, update, or delete records to, in, or from a desired table. It’s good to keep in mind that this transformation initiates a row-by-row operation, so you may experience some performance limitations when dealing with large amounts of data.

OLEDB destination can use Fast Load options hence perform bulk uploads.

**Q. What is the Difference between merge and Merge Join Transformation?**

**Ans:**

* **Merge Transformation:**
* The data from 2 input paths are merged into one
* Work as UNION ALL
* Metadata for all columns needs to be same
* Use when merging of data from 2 data source
* **Merge Join Transformation:**
* The data from 2 inputs are merged based on some common key.
* Work as JOIN (LEFT, RIGHT OR FULL)
* Key columns metadata needs to be same.
* Use when data from 2 tables having foreign key relationship needs to present based on common key

**Q. What is the difference between “ActiveX Script” and “Script Task”?**

**Ans:**

* We could say “Script Task” is the latest version for the deprecated feature “ActiveX Script”. Both are used to implement extended functionality in SSIS.
* ActiveX script supports VBScript and JScript where as “Script Task supports “VB.Net and C#.Net”.
* “Script Task” is preferable as “ActiveX Script” has been removed in MSSQL 2012.
* Script Task is supported with integrated help, IntelliSense, debugging and can reference external Dotnet assembles.

**Q. What is the difference between “Script Task” and “Script Component”?**

**Ans:**

* Both are used to extend the native functionality of SSIS.
* “Script Task” is to enhance the functionality for control flow where as “Script Component” is to enhance the functionality for Data flow.
* “Script Task” can handle the execution of parts of the package where as “Script Component” can handle the data flow and transformations by processing row by row.

**Q. What is the difference between “Execute SQL Task” and “Execute T-SQL statement” Task?**

**Ans:**

* The Execute T-SQL Statement task takes less memory, parse time, and CPU time than the Execute SQL task, but is not as flexible.
* If you need to run parameterized queries, save the query results to variables, or use property expressions, you should use the Execute SQL task instead of the Execute T-SQL Statement task
* Execute T-SQL Statement task supports only the Transact-SQL version of the SQL language
* Execute SQL task supports many connection types but the Execute T-SQL Statement task supports only ADO.NET

**Q. What is the difference between “Data Conversion” and “Derived Column” transformations?**

**Ans:**

Data Conversion transformation is used o convert the datatype of a column. Same operation can be done using “Derived Column “transformation using typecast but derived column can also be used to add / create a new column by manipulating the existing column based on expressions.

We have to choose “Data Conversion” when the requirement is only intended to change the datatype. In other words “Data Conversion” is introduced just for developer convenience as it’s a direct method where as in “Derived Column” we have to use an expression to change the datatype of a column.

From 2008 in “Derived Column” transformation, datatype and length information is read only, when we create a new column or created from existing , data type would be assigned based on the expression outcome and the datatype is a read-only column.

To change the datatype we have to use “Data Conversion” transformation.

**Q. What is the difference between “Copy Column” and “Derived Column”?**

**Ans:**

Both transformations can add new columns.

Copy column can add new columns only through existing columns but coming to Derived column it can add new columns without any help from existing columns.

Derived Column can assign different transformations and data types to the new columns whereas Copy Column cannot.

Derived Column supports error output whereas Copy Column cannot.

**Q. What is the difference between UNIONALL and MERGE transformations?**

**Ans:**

The Merge transformation combines two sorted datasets into a single dataset. The rows from each dataset are inserted into the output based on values in their key columns.

The Merge transformation is similar to the Union All transformations. Use the Union All transformation instead of the Merge transformation in the following situations:

* The transformation inputs are not sorted.
* The combined output does not need to be sorted.
* The transformation has more than two inputs.

**Q. What is the difference between for loop and for each loop container?**

**Ans:**

The “For Loop Container” executes specified number of times like 10 times, 20 times until the specified condition is met.

The “Foreach Loop Container” runs over an iterator. This iterator can be files from a folder, records from ADO, data from a variable etc.

**Q. How to pass property value at Run time? How do you implement Package Configuration?**

**Ans:**

A property value like connection string for a Connection Manager can be passed to the pkg using package configurations. Package Configuration provides different options like XML File, Environment Variables, SQL Server Table, Registry Value or Parent package variable.

**Q.** **How would you deploy a SSIS Package on production?**

**Ans:**

* Using Deployment Manifest
* Create deployment utility by setting its property as true.
* It will be created in the bin folder of the solution as soon as package is build.
* Copy all the files in the utility and use manifest file to deploy it on the Prod.
* Using import/Export and scheduling a job

**Q. What are the new features added in SQL Server 2008 SSIS?**

**Ans:**

* **Improved Parallelism of Execution Trees**
* **.NET language for Scripting**
* **New ADO.NET Source and Destination Component**
* **Improved Lookup Transformation**
* **New Data Profiling Task**
* **New Connections Project Wizard**
* **DT\_DBTIME2, DT\_DBTIMESTAMP2, and DT\_DBTIMESTAMPOFFSET data types**

**Improved Parallelism of Execution Trees**: The biggest performance improvement in the SSIS 2008 is incorporation of parallelism in the processing of execution tree. In SSIS 2005, each execution tree used a single thread whereas in SSIS 2008, the Data flow engine is redesigned to utilize multiple threads and take advantage of dynamic scheduling to execute multiple components in parallel, including components within the same execution tree

**.NET language for Scripting:** SSIS 2008 is incorporated with new Visual Studio Tool for Application(VSTA) scripting engine. Advantage of VSTA is it enables user to use any .NET language for scripting.

**New ADO.NET Source and Destination Component**: SSIS 2008 gets a new Source and Destination Component for ADO.NET Record sets.

**Improved Lookup Transformation:** In SSIS 2008, the Lookup Transformation has faster cache loading and lookup operations. It has new caching options, including the ability for the reference dataset to use a cache file (.caw) accessed by the Cache Connection Manager. In addition same cache can be shared between multiple Lookup Transformations.

**New Data Profiling Task**: SSIS 2008 has a new debugging aid Data Profiling Task that can help user analyze the data flows occurring in the package. The Data Profiling Task can help users to discover the coerce of these errors by giving better visibility into the data flow.

**New Connections Project Wizard:** One of the main usability enhancements to SSIS 2008 is the new Connections Project Wizard. The Connections Project Wizard guides user through the steps required to create source and destinations.

**DT\_DBTIME2, DT\_DBTIMESTAMP2, and DT\_DBTIMESTAMPOFFSET data types** – facilitate data type mapping to equivalent T-SQL date/time data types introduced in SQL Server 2008. Their primary purpose is to provide support for more accurate time measurements.

**Q. What are Synchronies and Asynchronous transformations in SSIS?**

**Ans:**

**Synchronizes Transformations:**

A synchronous transformation processes incoming rows and passes them on in the data flow one row at a time. Output is synchronous with input, it occurs at the same time. Therefore, to process a given row, the transformation does not need information about other rows in the data set. When a transform can modify the row in place so as to not change the physical layout of the result set, it is said to be a synchronous transformation. The output of a synchronous component uses the same buffer as the input and does not require data to be copied to a new buffer to complete the transformation. Reuse of the input buffer is possible because the output of a synchronous component usually contains the same number of records as the input;

An example of a synchronous transformation is the Data Conversion transformation. For each incoming row, it converts the value in the specified column and sends the row on its way. Each discrete conversion operation is independent of all the other rows in the data set.

**Asynchronous Transformations:**

The output buffer or output rows are not in sync with the input buffer; output rows use a new buffer. In these situations it’s not possible to reuse the input buffer because an asynchronous component can have more, the same or less output records than input records.

* The component has to acquire multiple buffers of data before it can perform its processing. An example is the Sort transformation, where the component has to process the complete set of rows in a single operation.
* The component has to combine rows from multiple inputs. An example is the Merge transformation, where the component has to examine multiple rows from each input and then merge them in sorted order.
* There is no one-to-one correspondence between input rows and output rows. An example is the Aggregate transformation, where the component has to add a row to the output to hold the computed aggregate values.

Asynchronous components can further be divided into the two types described below:

* **Partially Blocking Transformation** – the output set may differ in terms of quantity from the input set. Thus new buffers need to be created to accommodate the newly created set.
* **Blocking Transformation** – a transformation that must hold one or more buffers while it waits on one or more buffers, before it can pass that buffer down the pipeline. All input records must read and processed before creating any output records. For example, a sort transformation must see all rows before sorting and block any data buffers from being passed down the pipeline until the output is generated.

**Note:**

Synchronous components reuse buffers and therefore are generally faster than asynchronous components

**Q.** **Any Idea About execution tree?**

**Ans:**

At run time, the data flow engine breaks down Data Flow task operations into execution trees. These execution trees specify how buffers and threads are allocated in the package. Each tree creates a new buffer and may execute on a different thread.

Execution trees are enormously valuable in understanding buffer usage. They can be displayed for packages by turning on package logging for the Data Flow task

**Q.** **Where are SSIS package stored in the SQL Server?**

**Ans:**

* SQL Server 2000: MSDB..sysdtspackages
* SQL Server 2005: MSDB..sysdtspackages90
* SQL Server 2008: MSDB..sysssispackages

Stores the actual content and the following tables do the supporting roles.

* Sysdtscategories
* sysdtslog90
* sysdtspackagefolders90
* sysdtspackagelog
* sysdtssteplog
* sysdtstasklog

2008:

* sysssispackagefolders
* sysssislog

**Q. How to achieve parallelism in SSIS?**

**Ans:**

Parallelism is achieved using MaxConcurrentExecutable property of the package. Its default is -1 and is calculated as number of processors + 2.

**Q. Differences between dtexec.exe and dtexecui.exe**

**Ans:**

Both dtexec.exe and dtexecui.exe execute SSIS packages in the same manner. The difference is that dtexecui provided a graphical user interface to construct the command line arguments for dtexec. The command string that is generated with dtexecui can be used as command line arguments to dtexec.

**Q. Demonstrate or whiteboard how you would suggest using configuration files in packages. Would you consider it a best practice to create a configuration file for each connection manager or one for the entire package?**

**Ans:**

There should be a single configuration file for each connection manager in your packages that stores their connection string information. So if you have 6 connection managers then you have 6 config files. You can use the same config file across all your packages that use the same connections.

If you have a single config file that stores all your connection managers then all your packages must have contain the connection managers that are stored in that config file. This means you may have to put connection managers in your package that you don’t even need.

**Q. Demonstrate or whiteboard using a loop in a package so each file in a directory with the .txt extension is loaded into a table. Before demonstrating this tell which task/container accomplishes this and which enumerator will be used.**

**Ans:**

This would require a Foreach Loop using the Foreach File Enumerator. Inside the Foreach Loop Editor you need to set a variable to store the directory of the files that will be looped through. Next select the connection manager used to load the files and add an expression to the connection string property that uses the variable created in the Foreach Loop.

**Q. What techniques would you consider to add notification to your packages? You’re required to send emails to essential staff members immediately after a package fails.**

**Ans:**

This could either be set in the SQL Agent when the package runs or actually inside the package you could add a Send Mail Task in the Event Handlers to notify when a package fails.

There are also third party tools that can accomplish this for you (Pragmatic Works BI xPress).

**Q. Have you used SSIS Framework?**

**Ans:**

This is common term in SSIS world which just means that you have templates that are set up to perform routine tasks like logging, error handling etc. Yes answer would usually indicate experienced person, no answer is still fine if your project is not very mission critical.

**Q. How many difference source and destinations have you used?**

**Ans:**

It is very common to get all kinds of sources so the more the person worked with the better for you. Common ones are SQL Server, CSV/TXT, Flat Files, Excel, Access, Oracle, MySQL but also Salesforce, web data scrapping.

**Q. What configuration options have you used?**

**Ans:**

This is an important one. Configuration should always be dynamic and usually is done using XML and/or Environment Variable and SQL Table with all configurations.  
**Q. How do you apply business rules in SSIS (Transformations….Specific calculations but also cleansing)?**

**Ans:**

Some people use SSIS only to extract data and then go with stored procedures only….they are usually missing the point of the power of SSIS. Which allow creating “a flow” and on each step applies certain rules this greatly simplifies the ETL process.

**Q. Give example of handling data quality issues?**

**Ans:**

Data Quality is almost always a problem and SSIS handles it very well. Examples include importing customers from different sources where customer name can be duplicates. For instance you can have as company name: SQL Server Business Intelligence but also SQL Server BI or SQL Server BI LTD or SQL Server BI Limited or inteligence (with one l). There are different ways to handle it. Robust and time consuming is to create a table with or possible scenarios and update it after each update. You can also use fuzzy grouping which is usually easy to implement and will make usually very good decisions but it is not 100% accurate so this approach has to be justified.

Other typical quality issues are nulls (missing values), outliers (dates like 2999 or types like 50000 instead of 5000 especially important if someone is adjusting the value to get bigger bonus), incorrect addresses and these are either corrected during ETL, ignored, re-directed for further manual updates or it fails the packages which for big processes is usually not practiced.

**Q. When to use Stored Procedures?**

**Ans:**

This one is very important but also tricky. ALL SSIS developers have SQL Server background and that is sometime not very good if they use SQL not SSIS approach.

Let’s start with when you typically use SPs. This is for preparing tables (truncate), audit tasks (usually part of SSIS framework), getting configuration values for loops and a few other general tasks.  
During ETL extract you usually type simple SQL because it comes from other sources and usually over complication is not a good choice (make it dynamic) because any changes usually affect the package which has to be updated as well.

During Transformation phase (business rules, cleaning, core work) you should use Transformation tasks not Stored procedures! There are loads of tasks that make the package much easier to develop but also a very important reason is readability which is very important for other people who need to change the package and obviously it reduces risks of making errors. Performance is usually very good with SSIS as it is memory/flow based approach. So when to use Stored Procedures for transformations? If you don’t have strong SSIS developers or you have performance reasons to do it. In some cases SPs can be much faster (usually it only applies to very very large datasets). Most important is have reasons which approach is better for the situation.

**Q. What is your approach for ETL with data warehouses (how many packages you developer during typical load etc)?**

**Ans:**

This is rather generic question. A typical approach (for me) when building ETL is to. Have a package to extract data per source with extract specific transformations (lookups, business rules, cleaning) and loads data into staging table. Then a package do a simple merge from staging to data warehouse (Stored Procedure) or a package that takes data from staging and performs extra work before loading to data warehouse. I prefer the first one and due to this approach I occasionally consider having extract stage (as well as stage phase) which gives me more flexibility with transformation (per source) and makes it simpler to follow (not everything in one go). So to summarize you usually have package per source and one package per data warehouse table destination. There are might be other approach valid as well so ask for reasons.

**Q. What is XMLify component?**

**Ans:**

It is 3rd party free component used rather frequently to output errors into XML field which saves development time.  
**Q. What command line tools do you use with SSIS?**

**Ans:**

dtutil (deployment), dtexec (execution), dtexecui (generation of execution code)

**Q. What is data cleansing?**

**Ans:**

Used mainly in databases, the term refers to identifying incomplete, incorrect, inaccurate, irrelevant, etc. parts of the data and then replacing, modifying, or deleting this dirty data.

**Q. Any Idea what is ETI?**

**Ans:**

Yes! ETI (Error Tolerant Index) is a technique used in Fuzzy Lookup / Fuzzy Grouping for data cleansing operation. The ETI is a decomposition of the field values contained within a reference table of values into smaller tokens is nothing but a match index.

For example, instead of searching for a street address that contains the value “112 Sunny Vail Ln.”, smaller components of the reference value might be used, such as “sunn”, “nyva”, and “112”.

These individual words are called Tokens, and all tokens in a index are divided using some special character and search with the reference table.

http://udayarumilli.com/wp-content/uploads/2015/05/word-image1.png

**Q. What is Fuzzy Lookup? Can you demonstrate it?**

**Ans:**

Fuzzy lookup transformation is data cleaning task that helps to clean the incoming data with the reference table with the actual value. This transformation tries to find the exact or similar value as a result. The result data set is also depends on the fuzzy matching configuration in the fuzzy lookup transformation task. Fuzzy lookup task will be more helpful when you have data typo issues in the source data.

Fuzzy Lookup transformation creates temporary objects, such as tables and indexes in the SQL Server TempDB. So, make sure that the SSIS user account has sufficient access to the database engine to create and maintain this temporary table. Fuzzy lookup transformation has 3 features.

* Defining maximum number of matches to return to output – It starts with 1 and that is the recommended.
* Token delimiters – It has a set of predefined delimiters and we can also add our’s
* Similarity score – It is the fuzzy algorithm input to match the score with the input row and reference row. This value is between 0 and 1. higher the value is the accurate the result. It is usually 0.60 is the best value for similarity score.

**Q. What shape would you use to concatenate two input fields into a single output field?**

**Ans:**

Pivot transformation

**Q. What is the Multicast Shape used for?**

**Ans:**

The Multicast transformation distributes its input to one or more outputs. This transformation is similar to the Conditional Split transformation. Both transformations direct an input to multiple outputs. The difference between the two is that the Multicast transformation directs every row to every output, and the Conditional Split directs a row to a single output

**Q. What types of things can I pass between packages in SSIS?**

**Ans:**

We can pass Variables primarily between packages. Within a variable we can pass them as any type that is available. So if you were to create an object variable, although memory consuming, we could potentially pass a table that is in memory. Granted, in SQL Server 2012 (Denali) this is much, much easier now with parameters. Actually, this was almost a relief in a way. Configuring packages to consume parent variables was a time consuming and in some cases, confusing situation when many variables were in the process.

**Q. How to accomplish incremental loads? (Load the destination table with new records and update the existing records from source (if any updated records are available)**

**Ans:**

There are few methods available:

* You can use **Lookup Transformation** where you compare source and destination data based on some id/code and get the new and updated records, and then use **Conditoional Split** to select the new and updated rows before loading the table. However, I don’t recommend this approach, especially when destination table is very huge and volume of delta is very high.
* Use Execute SQL Task and with Staging table
* Find the Maximum ID & Last ModifiedDate from destination and store in package variables. (Control Flow)
* Pull the new and updated records from source and load to a staging table (A dataload table created in destination database) using above variables.(Data Flow)
* Insert and Update the records using Execute SQL Task (Control Flow)
* Use the feature CDC (Change Data Capture) from SQL Server 2008
* Use Conditional split to split data for Inserts. Updates and Deletes
* For inserts redirect to a OLEDB Destination
* For Updates and Deletes redirect using a OLEDB Command transformation

**Q. How can you enable the CDC for a table?**

**Ans:**

To enable CDC to a table first the feature should be enabled to the corresponding database. Both can be done using the below procs.

exec sys.sp\_cdc\_enable\_db\_change\_data\_capture

sys.sp\_cdc\_enable\_table\_change\_data\_capture

**Q. How can you debug Dataflow?**

**Ans:**

Microsoft Integration Services and the SSIS Designer include features and tools that you can use to troubleshoot the data flows in an Integration Services package.

* SSIS Designer provides data viewers.
* SSIS Designer and Integration Services transformations provide row counts.
* SSIS Designer provides progress reporting at run time.
* Redirect to specified points using error output

**Q. How to debug control flow?**

**Ans:**

* Integration Services supports breakpoints on containers and tasks.
* SSIS Designer provides progress reporting at run time.
* Business Intelligence Development Studio provides debug windows.

**Q. What can you tell me about Ralph Kimball?**

**Ans:**

Ralph Kimball is an author on the topic of data warehousing and BI. He has been regarded as one of the original architects of data warehousing. Kimball has always had the firm belief that data warehouses should fast and understandable. Oh, and he developed this whole methodology of dimensional modeling. There is that. (It’s also probably a good idea to know the basic idea and structure of dimensional modeling)

**Q.** **Are you familiar with Package Configurations?**

**Ans:**  
Yes. Recently I was working on a project where we used the SQL Server Table package configuration to store values for the package parameters. That allowed me to build a GUI for the users to update the package variables each month with new values.  
**Q. Have you ever used the XML package configuration?**

**Ans:**

Yes. In fact, that is the method we use for storing the connection string used by the sql server table package configuration for the project I just mentioned. We have a dev/production environment, so using an xml file with the connection string (and pointing to that XML file from an environment variable) makes it easy to switch between the two servers.

**SSAS Interview Questions**

##### Problem

Dimensions are a very important component of an OLAP Solution/Cube in SQL Server Analysis Services. Hence it is good to have a fair amount of knowledge/understanding on the Dimensions, its properties, and its sub components like Attributes, Hierarchies etc.

##### Solution

In the [first tip](https://www.mssqltips.com/sqlservertip/2637/ssas-interview-questions-part-i-questions-on-basic-concepts-data-sources-and-data-source-views/) of this series, I covered some of the Questions on Basic Concepts, Data Sources, and Data Source Views in SQL Server Analysis Services. And in the [second tip](https://www.mssqltips.com/sqlservertip/2662/sql-server-analysis-services-interview-questions-part-ii--dimensions/), I covered some of the Questions on General Dimension Concepts and Types of Dimensions. In this third tip, I will be covering some of the questions on Dimensions, its properties, and its components like Attributes, Hierarchies etc. within SQL Server Analysis Services.

### **What are Database Dimension and Cube Dimension? What is the difference between them?**

A [Database Dimension](https://www.mssqltips.com/sqlservertutorial/2006/creating-a-dimension/) is a Dimension which exists independent of a Cube. You can see a Database Dimension as a separate entity/file in the Solution Explorer/Project Folder. An analogy, this is pretty similar to a Class in a Programming Language.

A Cube Dimension is an Instance of a Database Dimension. An analogy, this is pretty similar to an Object (Instance of a Class) in a Programming Language.

Here are some of the highlights/differences of Database and Cube Dimensions:

* A Database Dimension is independent of a Cube, but a Cube Dimension is always a part of a Cube.
* One Database Dimension can be added to a Cube more than once ([Role-Playing Dimension](https://www.mssqltips.com/sqlservertip/2662/sql-server-analysis-services-interview-questions-part-ii--dimensions/)) as different instances. In this scenario, each Instance is treated as a separate Cube Dimension.
* A Database Dimension can be added to more than one Cube, whereas a Cube Dimension can belong to a Single Cube at any given point.

### **What is a Linked Dimension? In what scenarios do you use it?**

A [Linked Dimension](https://www.mssqltips.com/sqlservertip/2351/controlling-the-use-of-linked-objects-in-sql-server-analysis-services/) is a Dimension which is based on (Linked To) another Database Dimension which might be either located on the same Analysis Services server as the Linked Dimension or on a different Analysis Services server.

Linked Dimensions can be used when the exact same dimension can be used across multiple Cubes within an Organization like a Time Dimension, Geography Dimension etc.

Here are some of the highlights of a Linked Dimension:

* More than one Linked Dimension can be created from a Single Database Dimension.
* These can be used to implement the concept of [Conformed Dimensions](https://www.mssqltips.com/sqlservertip/2662/sql-server-analysis-services-interview-questions-part-ii--dimensions/).
* For an end user, a Linked Dimension appears like any other Dimension.

### **What are the different ways to create a Time Dimension in Analysis Services?**

Time Dimension is one of the most important and most common type of dimensions as pretty much every metric is analyzed over time. Analysis Services offers following different ways to create a Time Dimension:

* ***Create Using an Existing Table***: This is one of the common approaches for building a Time Dimension. In this approach, a table is created in the underlying data source and pre-populated with data and it is then used to create the Time Dimension in SSAS.
* ***Generate a Time Table in the Data Source***: In this approach there will be no underlying table and at the time of creation of a Time Dimension in SSAS, Analysis Services creates a table in the underlying data source (relational database) and populates it with the required data using the date range, attributes, and calendars etc. which are specified at the time of creation of Time Dimension in Business Intelligence Development Studio. This option requires permissions to create a table in the underlying data source.
* ***Generate a Time Table on the Server***: In this approach there will be no underlying table and at the time of creation of a Time Dimension in SSAS, Analysis Services creates a table on the server and this table is used to generate the Time Dimension by the wizard. The Dimension created using this approach is called a Server Time Dimension.

### **What is Type property of a Dimension? What is the purpose of setting this property?**

Type property of a Dimension is used to specify the type of information that the Analysis Services Dimension contains. Like a Time Dimension (Contains Years, Quarters, Months, Dates, and so on), Geography Dimension (Contains Geographical Information), and Accounts Dimension (Contains Accounts related information) etc. This property is set to Regular by default.

There are basically two important uses of setting this property:

* This property is used by the [Business Intelligence](https://www.mssqltips.com/sql_server_business_intelligence_tips.asp) Wizard to assign/generate standard classification/[MDX](https://www.mssqltips.com/sqlservertip/2292/how-to-reduce-mdx-code-redundancy-in-sql-server-analysis-services-ssas/) expressions.
* This property is used by the Analysis Services Client Applications like Excel etc. to adjust the user interface/rendering format appropriately. For example, when a dimension is added to a [Pivot Table in Excel](https://www.mssqltips.com/sqlservertutorial/2016/using-excel-and-creating-a-pivot-table-report/), certain types of dimensions are automatically added across the columns axis (like Time Dimension) and a few other types of dimensions are added to the rows axis (Like Product Dimension) automatically.

### **What is a Storage Mode? What are the different storage modes applicable to Dimensions?**

A Storage Mode defines the location in which the Dimension data will be stored and the format (Relational or Multidimensional) in which the data will be stored.

Following are the two Storage Modes supported by Dimensions:

* ***ROLAP (Relational On-Line Analytical Processing):***
  + When the Storage Mode is set to ROLAP for a Dimension, then the Dimension Data will be stored in the relational database tables.
  + This storage Mode offers effective memory usage by avoiding duplication of data compared to MOLAP Storage Mode.
  + Using this Storage Mode will result in a slowdown in the query performance compared to MOLAP Storage Mode.
* ***MOLAP (Multidimensional On-Line Analytical Processing):***
  + When the Storage Mode is set to MOLAP for a Dimension, then the Dimension Data will be stored in a multidimensional format in the Analysis Services/OLAP Server.
  + This Storage Mode offers poor memory usage as it involves duplication of data (first copy is the data in the underlying dimensional tables and the second copy is the data in the Dimension in the OLAP Server).
  + Using this Storage Mode will result in best query performance compared to any other Storage Modes available in SSAS.

### **What is the difference between Attribute Hierarchy and User Defined Hierarchy?**

An Attribute Hierarchy is a Hierarchy created by SQL Server Analysis Services for every Attribute in a Dimension by default. An Attribute by default contains only two levels - An "All" level and a "Detail" level which is nothing but the Dimension Members.

A User Defined Hierarchy is a Hierarchy defined explicitly by the user/developer and often contains multiple levels. For example, a Calendar Hierarchy contains Year, Quarter, Month, and Date as its levels.

Here are some of the highlights/differences of Attribute and User Defined Hierarchies:

* Attribute Hierarchies are always Two-Level (Unless All Level is suppressed) whereas User Defined Hierarchies are often Multi-Level.
* By default, Every Attribute in a Dimension has an Attribute Hierarchy whereas User Defined Hierarchies have to be explicitly defined by the user/developer.
* Every Dimension has at least one Attribute Hierarchy by default whereas every Dimension does not necessarily contain a User Defined Hierarchy. In essence, a Dimension can contain zero, one, or more User Defined Hierarchies.
* Attribute Hierarchies can be enabled or disabled. Disable the Attribute Hierarchy for those attributes which are commonly not used to slice and dice the data during analysis, like Address, Phone Number, and Unit Price etc. Doing this will improve the cube processing performance and also reduces the size of the cube as those attributes are not considered for performing aggregations.
* Attribute Hierarchies can be made visible or hidden. When an Attribute Hierarchy is hidden, it will not be visible to the client application while browsing the Dimension/Cube. Attribute Hierarchies for those attributes which are part of the User Defined Hierarchies, like Day, Month, Quarter, and Year, which are part of the Calendar Hierarchy, can be hidden, since the attribute is available to the end users through the User Defined Hierarchy and helps eliminate the confusion/redundancy for end users.

### **What is an Attribute Relationship? What are the different types of Attribute Relationships?**

An [Attribute Relationship](https://www.mssqltips.com/sqlservertutorial/2007/creating-a-hierarchy/) is a relationship between various attributes within a Dimension. By default, every Attribute in a Dimension is related to the Key Attribute. Quite often these default Attribute Relationships need to be modified to suit the User Defined Hierarchies and other end user requirements.

There are basically two types of Attribute Relationships:

* ***Rigid***: Attribute Relationship should be set to Rigid when the relationship between those attributes is not going to change over time. For example, relationship between a Month and a Date is Rigid since a particular Date always belongs to a particular Month like 1st Feb 2012 always belongs to Feb Month of 2012. Try to set the relationship to Rigid wherever possible.
* ***Flexible***: Attribute Relationship should be set to Flexible when the relationship between those attributes is going to change over time. For example, relationship between an Employee and a Manager is Flexible since a particular Employee might work under one manager during this year (time period) and under a different manager during next year (another time period).

### **What are KeyColumns and NameColumn properties of an Attribute? What is the different between them?**

KeyColumns is a property of an SSAS Dimension Attribute and it forms the Key (Unique) for the attribute. It can be bound to one or more columns in the underlying database table. When User Defined Hierarchies are created in the dimension (Attribute Relationships defined), setting this property becomes very critical and often requires setting this to a combination of more than one column from the Data Source View. For Example, say you have a Date Dimension and a hierarchy called Calendar Hierarchy (Year -> Quarter -> Month). Now what happens is that, Month gets repeated across different quarters and quarters get repeated across different years making the attribute as non-unique (like January can belong to Q1 of any year and similar Q1 can belong to any year). So to make the attribute unique, KeyColumns for Month should be set to something like Year and Month and similarly for Quarter should be set to Year and Quarter.

A NameColumn is a property of an SSAS Dimension Attribute and it is used to identify the column from the underlying Data Source View which provides the name of the attribute which is displayed to the end user by making it more user friendly instead of displaying the Key Column value. For Example, you might have ProductCategoryKey as 1, 2, 3, & 4, and ProductCategoryName as Bikes, Components, Clothing, & Accessories respectively. Now, NameColumn will be set to ProductCategoryName so that user sees them as Bikes, Components etc. even though the data in the background is processed/retrieved using the Key Column values as 1, 2 etc.

Here are some of the highlights/differences of KeyColumns and NameColumn properties:

* KeyColumns property is defaulted to the Attribute itself, and the NameColumn property is defaulted to Key Column (when the KeyColumns is set to only one column).
* Column(s) provided in the KeyColumns should be able to uniquely identify all the values of the respective attribute, whereas NameColumn need not be unique.
* KeyColumns can contain one or more columns whereas NameColumn can contain only one column.

### **What is an Unknown Member? What is its significance?**

An Unknown Member is a built-in member provided by SQL Server Analysis Services. It represents a Missing or Null value. Basically when a Dimension is Processed, Analysis Services populates each of the attributes with distinct values from the underlying data source and in this process, if it encounters Null value then it converts them appropriately (to 0 in case of numeric columns and to empty string in case of string columns) and marks them as Unknown Member for easy interpretation by the end user.

One of the important uses of an Unknown Member is to handle [Early Arriving Facts](https://www.mssqltips.com/sqlservertip/1446/handle-early-arriving-facts-in-sql-server-integration-services-ssis/). This is a scenario, in which the transactional record (Fact) would have come into the system but the corresponding Dimension/Contextual data is yet to come in which could be due to various reasons like Improperly Designed Data Load Process, Failure in the ETL Process, and a Technical Glitch in the transactional system causing delay in pushing the Dimension Data.

Unknown Member can be enabled or disabled (set to None) or set to Visible or Hidden based on the end user requirements.

### **What are Dimension Translations? In what scenarios do you use them?**

Translation in SSAS is a mechanism to support Localization, in which the labels, names, and captions associated with any SSAS Object (pretty much every SSAS Object supports Translations) are translated from one language to another language based on the country/language of the user accessing the data. A Dimension Translation is same as Translation in SSAS, but in this case the labels, names, and captions associated with Dimensions, Attributes, Hierarchies, and/or Dimension Members are translated from one language to another language.

Translations are very useful in achieving higher level of adoption of the BI/Analytics system (SSAS). This will eliminate the language barriers among users from different locations/languages and presents the same information in different languages making single version of truth available to users across different geographical locations.

Here are some of the highlights of Dimension Translations:

* There can be multiple Translations associated with a single Dimension, Attribute, Hierarchy, and Dimension Member etc.
* The collation and language settings/information from the client application/computer is used to determine and provide the analysis services metadata/data to the client application.

**Q1) Explain about Data Mining Archtecture & Techniques?**

|  |  |  |
| --- | --- | --- |
| **Data Mining Architecture & Techniques** | | |
|  | **Resources** | **Explanation** |
| Architecture | Data sources | WWW or Internet is big source of Data |
|  | Database | Database contains data which is ready to be processed |
|  | Data mining engine | It contains modules used to perform data mining tasks |
|  | Pattern evaluation module | It mentions the measure of the differnitaions of patterns |
|  | GUI | It is the communication between user and data mining system |
|  | Knowledge Base | It is a guide for result patterns, based on models it interacts |
| Types Of Mining Architecture | No-coupling Data Mining | It retrives data from a particular data sources |
|  | Loose Coupling Data Mining | It retrives data from a particular database |
|  | Semi-Tight Coupling Data Mining | In this it uses few features of data warehouse sys |
|  | Tight Coupling Data mining | Data layer, Data mining application layer, Front-end layer |
| Data Mining Techniques | Decision Trees | Common technique used for mining, The root acts as condition |
|  | Sequential patterns | It is used to identifiy events, similar patterns of transaction data |
|  | Clustering | By automatic method similar characterstics clusters have to form |
|  | Prediction | It defines the relationship of dependent & independent instances |
|  | Association | It is a relation technique, used to recognize the pattern |
|  | Classification | It depends on ML, Used to classifiy item of particular set to predinned groups |
| Technological Drivers | Database size | For mainatining & Processing data we need powerful sys |
|  | Query complexity | It is used to analyze complex queris in large number. |

**Q. What is SQL Server Analysis Services (SSAS)? List out the features?**  
Microsoft SQL Server 2014 Analysis Services (SSAS) delivers online analytical processing (OLAP) and data mining functionality for [business intelligence applications](https://mindmajix.com/sas-bi/introduction-sas-business-intelligence-clients). Analysis Services supports OLAP by letting us design, create, and manage multidimensional structures that contain data aggregated from other data sources, such as relational databases. For [data mining](https://mindmajix.com/business-analytics-with-r-training) applications, **Analysis Services** lets we design, create, and visualize data mining models that are constructed from other data sources by using a wide variety of industry-standard data mining algorithms.  
Analysis Services is a middle tier server for analytical processing, OLAP, and Data mining. It manages multidimensional cubes of data and provides access to heaps of information including aggregation of data. One can create data mining models from data sources and use it for Business Intelligence also including reporting features.  
Analysis service provides a combined view of the data used in OLAP or Data mining. Services here refer to [OLAP](https://mindmajix.com/sap-bo/how-to-establish-olap-connection-in-sap-bo-analysis), Data mining. Analysis services assists in creating, designing and managing multidimensional structures containing data from varied sources. It provides a wide array of data mining algorithms for specific trends and needs.  
**Some of the key features are:**  
1.Ease of use with a lot of wizards and designers.  
2.Flexible data model creation and management  
3.Scalable architecture to handle OLAP  
4.Provides integration of administration tools, data sources, security, caching, and reporting etc.  
5.Provides extensive support for custom applications

**Q. What is the difference between SSAS 2005 and SSAS2008?**  
1.In 2005 its not possible to create an empty cube but in 2008 we can create an empty cube.  
2.A new feature in Analysis Services 2008 is the Attribute Relationships tab in the Dimension Designer.to implement attribute relationship is complex in ssas 2005  
3.we can create ONLY 2000 partitions per Measure Group in ssas 2005 and the same limit of partitions is removed in ssas 2008.

[Wish to Learn SSAS? Click Here](https://mindmajix.com/ssas-training)

**Q. What is OLAP? How is it different from OLTP?**  
1.OLAP stands for On-Line Analytical Processing. It is a capability or a set of tools which enables the end users to easily and effectively access the data warehouse data using a wide range of tools like **MICROSOFT EXCEL**, **REPORTING SERVICES**, and many other 3rd party **BUSINESS INTELLIGENCE TOOLS.**  
2.OLAP is used for analysis purposes to support day-to-day business decisions and is characterized by less frequent data updates and contains historical data. Whereas, OLTP (On-Line Transactional Processing) is used to support day-to-day business operations and is characterized by frequent data updates and contains the most recent data along with limited historical data based on the retention policy driven by business needs.

**Q. What is a Data Source? What are the different data sources supported by SSAS?**  
**A DATA SOURCE** contains the connection information used by SSAS to connect to the underlying database to load the data into SSAS during processing. A Data Source primarily contains the following information (apart from various other properties like Query timeout, Isolation etc.):  
1.Provider  
2.Server Name  
3.Database Name  
4.Impersonation Information  
SSAS Supports both .Net and OLE DB Providers. Following are some of the major sources supported by SSAS: [SQL Server](https://mindmajix.com/msbi/sql-server-analysis-services-in-ssas-cube), MS Access, Oracle, Teradata, IBM DB2, and other relational databases with the appropriate OLE DB provider.

**Q. What is Impersonation? What are the different impersonation options available in SSAS?**  
Impersonation allows SSAS to assume the identity/security context of the client application which is used by SSAS to perform the server side data operations like data access, **PROCESSING** etc. As part of impersonation, the following options are available in SSAS:  
1.Use a specific Windows user name and password: This option lets you to specify Windows account credentials which will be used by SSAS to perform operations like source data access, processing etc.  
2.Use the service account: When this option is selected, SSAS uses the credentials of the service account under which the [Analysis Services](https://mindmajix.com/msbi/ssas-analysis-services-cube-deployment-methods) service is configured/running for source data access, processing etc.  
3.Use the credentials of the current user: When this option is set, SSAS uses the credentials of the current user for performing operations like DMX Open Queries, Local cubes etc. This option cannot be used for performing server side operations like source data access, processing etc.  
4.Inherit: This option let’s the SSAS server decide which impersonation mode is suitable for each type of operation. When this option is set, by default SSAS will use the service account for operations like processing and the credentials of the current user for operations like Local cubes, querying the data mining models, etc…

**Q. What is a Data Source View?**  
**A DATA SOURCE VIEW (DSV)** is a logical view of the underlying database schema and offers a layer of abstraction for the underlying database schema. This layer acts as a source for SSAS and captures the schema related information from the underlying database. The schematic information present in DSV includes the following:  
1.Underlying database Table(s)/View(s) metadata  
2**.PRIMARY KEY & FOREIGN KEY RELATIONSHIPS** between the underlying database Table(s)  
3.Additional columns in the form of Named Calculations  
4.Complex logic on the underlying Table(s)/View(s) in the form of Named Queries  
SSAS can only see the schematic information present in the DSV and it cannot see the schematic information from the underlying database.

**Q. What is a Named Calculation? In what scenarios do you use it?**  
A Named Calculation is a new column added to a Table in DSV and is based on an expression. This capability allows you to add an extra column into your DSV which is based on one or more columns from underlying data source Table(s)/View(s) combined using an expression without requiring the addition of a physical column in the underlying database Table(s)/View(s).  
The expression used in the Named [Calculation](https://mindmajix.com/msbi/introduction-to-ssas-calculations-and-actions) should conform to the underlying data source dialect. For example, if the underlying data source is SQL Server, then it should conform to **T-SQL,** If is it Oracle, then it should conform to**PL/SQL**, etc…  
Named Calculations can be used in many scenarios, following are some of the common scenarios:  
1.For creating Derived Columns. Say you have First Name and Last Name in the underlying data source Table/View and you want to get the Full Name as “First Name + space + Last Name”. Such things can be added as a Named Calculation.  
2.For performing Lookup Operations. Say you have an Employee table which has AddressID and an Address table in which AddressID is the Primary Key. Now, to get the address information (say Country) into the Employee table in DSV, a Named Calculation can be added to the Employee table with the following expression:  
(  
SELECT Country  
FROM Address  
WHERE AddressID = Employee.AddressID  
)

**Q. What is a Named Query? In what scenarios do you use it?**  
1.A Named Query is a SQL query/expression in your DSV which acts as a Table. It is used to combine data from one or more Table(s)/View(s) from the underlying data source without requiring any schematic changes to the underlying data source Table(s)/View(s).  
2.The SQL Query used in the Named Query should conform to the underlying data source dialect. For example, if the underlying data source is SQL Server, then it should conform to T-SQL, If is it Oracle, then it should conform to PL/SQL, etc…  
3.Named Queries are used in various scenarios, following are some of the common scenarios:  
i)Combining data from multiple Tables/Views from the underlying data source by using either simple or complex join operations.  
ii)Adding filter conditions for filtering out unwanted data or selecting the required data (limiting the data).  
4.Pretty much everything that can be done using a Named Calculation can also be done using a Named Query.

**Q. What are the pros and cons of using Tables and Named Queries in DSV?**  
Following are some of the pros and cons of using Tables and Named Queries in DSV. Tables in the below comparison refers to the Table in DSV which references a single Table or a View in the underlying source database.

|  |  |
| --- | --- |
| **Tables** | **Named Queries** |
| Named Calculations can be added to Tables in DSV. | Named Calculations cannot be added to Named Queries in DSV. |
| Named Calculations cannot be added to Named Queries in DSV. | Keys and Relationships have to be set explicitly in the DSV. |
| Only one Table/View from the underlying data source can be referenced in DSV. | More than one Table/View from the underlying data source can be referenced using a SQL Expression in the DSV. |
| Any filter/limiting conditions cannot be applied on a table in DSV. | Filter/limiting conditions can be applied as part of the SQL expression in the Named Query in the DSV. |

Although Named Calculations and Named Queries can be used to extend the functionality of SSAS to address the evolving business needs, it is always a good practice to first build a good **DIMENSIONAL MODEL** in the beginning of a Data Warehousing/SSAS project.

**Q. What is the purpose of setting Logical Keys and Relationships in DSV?**  
1.Many of the user interfaces/designers/wizards in BIDS which are part of a SSAS project depend on the Primary Key and Relationships between Fact and Dimension tables. Hence it is important to define the Primary Key and Relationships in DSV.  
2.By default, the Data Source View Wizard detects the Physical Primary Keys and Relationships between the tables in the underlying source database and applies the same Keys and Relationships in DSV layer. However, Logical Keys and Relationships need to be defined explicitly in the following scenarios:  
i)If the DSV table is referring to an underlying database View.  
ii)If the DSV table is created as a Named Query.  
iii)If any additional relationships need to be defined in the DSV layer apart from the ones that are physically defined in the underlying source database.

**Q. Is it possible to combine data from multiple data sources in SSAS? If yes, how do you accomplish it?**  
1.SSAS allows combining data from multiple underlying data sources into a single DSV. To be able to add Table(s)/View(s) from multiple data sources, first you need to create a DSV using your first source and this source acts as the primary data source. Now after the initial DSV is created, you can add one or more data sources into DSV which will act as secondary data sources and you can choose additional Table(s)/View(s) from the secondary data sources which you want to include in your DSV.  
2.The key thing while combining data from multiple data sources is that the Primary Data Source must support OPENROWSET queries. Hence in most cases, SQL Server is used as the Primary Data Source.

**Q. What is UDM? Its significance in SSAS?**  
The role of a Unified Dimensional Model (UDM) is to provide a bridge between the user and the data sources. A UDM is constructed over one or more physical data sources, and then the end user issues queries against the UDM using one of a variety of client tools, such as Microsoft Excel. At a minimum, when the UDM is constructed merely as a thin layer over the data source, the advantages to the end user are a simpler, more readily understood model of the data, isolation from heterogeneous backend data sources, and improved performance for summary type queries. In some scenarios a simple UDM like this is constructed totally automatically. With greater investment in the construction of the UDM, additional benefits accrue from the richness of metadata that the model can provide.  
**The UDM provides the following benefits:**  
1.Allows the user model to be greatly enriched.  
2.Provides high performance queries supporting interactive analysis, even over huge data volumes.  
3.Allows business rules to be captured in the model to support richer analysis.

**Q. What is the need for SSAS component?**  
1.Analysis Services is the only component in SQL Server using which we can perform Analysis and Forecast operations.  
2.SSAS is very easy to use and interactive.  
3.Faster Analysis and Troubleshooting.  
4.Ability to create and manage Data warehouses.  
5.Apply efficient Security Principles.

**Q. Explain the TWO-Tier Architecture of SSAS?**  
1.SSAS uses both server and client components to supply OLAP and data mining functionality BI Applications.  
2.The server component is implemented as a Microsoft Windows service. Each instance of Analysis Services implemented as a separate instance of the Windows service.  
3.Clients communicate with Analysis Services using the standard XMLA (XML For Analysis) , protocol for issuing commands and receiving responses, exposed as a web service.

**Q. What are the components of SSAS?**  
1.An OLAP Engine is used for enabling fast adhoc queries by end users. A user can interactively explore data by drilling, slicing or pivoting.  
2.Drilling refers to the process of exploring details of the data.  
3.Slicing refers to the process of placing data in rows and columns.  
4.Pivoting refers to switching categories of data between rows and columns.  
5.In OLAP, we will be using what are called as Dimensional Databases.

**Q. What is FASMI?**  
A database is called an OLAP Database if the database satisfies the FASMI rules :  
1.Fast Analysis–is defined in the OLAP scenario in five seconds or less.  
2.Shared –Must support access  to data by many users in  the factors of Sensitivity and Write Backs.  
3.Multidimensional –The data inside the OLAP Database must be multidimensional in structure.  
4.Information –The OLAP database Must support large volumes of data..

**Q. What languages are used in SSAS?**  
1.Structured Query Language (SQL)  
2.Multidimensional Expressions (MDX) – an industry standard query language orientated towards analysis  
3.Data Mining Extensions (DMX) – an industry standard query language oriented toward data mining.  
4.Analysis Services Scripting Language (ASSL) – used to manage Analysis Services database objects.

**Q. How Cubes are implemented in SSAS?**  
1.Cubes are multidimensional models that store data from one or more sources.  
2.Cubes can also store aggregations  
3.SSAS Cubes are created using the Cube Wizard.  
4.We also build Dimensions when creating Cubes.  
5.Cubes can see only the DSV( logical View).

**Q. What is the difference between a derived measure and a calculated measure?**  
The difference between a derived measure and a calculated measure is when the calculation is performed. A derived measure is calculated before aggregations are created, and the values of the derived measure are stored in the cube. A calculated measure is calculated after aggregations are created, and the values of a calculated measure aren’t stored in the cube. The primary criterion for choosing between a derived measure and a calculated measure is not efficiency, but accuracy.

**Q. What is a partition?**  
A partition in Analysis Services is the physical location of stored cube data. Every cube has at least one partition by default. Each time we create a measure group, another partition is created. Queries run faster against a partitioned cube because Analysis Services only needs to read data from the partitions that contain the answers to the queries. Queries run even faster when partition also stores aggregations, the pre calculated totals for additive measures. Partitions are a powerful and flexible means of managing cubes, especially large cubes.

**Q. While creating a new calculated member in a cube what is the use of property called non-empty behavior?**  
Nonempty behavior is important property for ratio calculations. If the denominator Is empty, an MDX expression will return an error just as it would if the denominator Were equal to zero. By selecting one or more measures for the Non-Empty Behavior property, we are establishing a requirement that each selected measure first be evaluated before the calculation expression is evaluated. If each selected measure is empty, then The expression is also treated as empty and no error is returned.

**Q. What is a RAGGED hierarchy?**  
Under normal circumstances, each level in a hierarchy in Microsoft SQL Server Analysis Services (SSAS) has the same number of members above it as any other member at the same level. In a ragged hierarchy, the logical parent member of at least one member is not in the level immediately above the member. When this occurs, the hierarchy descends to different levels for different drilldown paths. Expanding through every level for every drilldown path is then unnecessarily complicated.

**Q. What are the roles of an Analysis Services Information Worker?**  
The role of an Analysis Services information worker is the traditional “domain expert” role in business intelligence (BI) someone who understands the data employed by a solution and is able to translate the data into business information. The role of an Analysis Services information worker often has one of the following job titles: Business Analyst (Report Consumer), Manager (Report Consumer), Technical Trainer, Help Desk/Operation, or Network Administrator.

Explore SSAS Sample Resumes! Download & Edit, Get Noticed by Top Employers![**Download Now!**](https://mindmajix.com/ssas-sample-resumes)

**Q. What are the different ways of creating Aggregations?**  
We can create aggregations for faster MDX statements using Aggregation Wizard or thru UBO – Usage Based Optimizations. Always, prefer UBO method in realtime performance troubleshooting.

**Q. What is WriteBack? What are the pre-conditions?**  
The **Enable/Disable** Writeback dialog box enables or disables writeback for a measure group in a cube. Enabling writeback on a measure group defines a writeback partition and creates a writeback table for that measure group. Disabling writeback on a measure group removes the writeback partition but does not delete the writeback table, to avoid unanticipated data loss.

**Q. What is processing?**  
Processing is a critical and resource intensive operation in the data warehouse lifecycle and needs to be carefully optimized and executed. Analysis Services offers a high performance and scalable processing architecture with a comprehensive set of controls for database administrators.  
We can process an OLAP database, individual cube, Dimension or a specific Partition in a cube.

**Q. Name few Business Analysis Enhancements for SSAS?**  
The following table lists the business intelligence enhancements that are available in Microsoft SQL Server Analysis Services (SSAS). The table also shows the cube or dimension to which each business intelligence enhancement applies, and indicates whether an enhancement can be applied to an object that was created without using a data source and for which no schema has been generated.

|  |  |  |  |
| --- | --- | --- | --- |
| **Enhancement** | **Type** | **Applied to** | **No data source** |
| Time Intelligence | Cube | Cube | No |
| Account Intelligence | Dimension | Dimension or cube | No |
| Dimension Intelligence | |  | | --- | | Dimension | | Dimension or cube | Yes |
| Custom Aggregation | |  | | --- | | Dimension | | Dimension (unary operator) or cube | No |
| Semiadditive Behavior | Cube | Cube | Yes |
| Custom Member Formula | Dimension | Dimension or cube | No |
| Custom Sorting and Uniqueness Settings | Dimension | Dimension or cube | Yes |
| Dimension Writeback | Dimension | Dimension or cube | Yes |

**Q. What MDX functions do you most commonly use?**  
This is a great question because you only know this answer by experience.  If you ask me this question, the answer practically rushes out of me.  “Cross Join, Descendants, and Non Empty, in addition to Sum, Count, and Aggregate.  My personal favorite is Cross Join because it allows me identify non-contiguous slices of the cube and aggregate even though those cube cells don’t roll up to a natural ancestor.”  Indeed, Cross Join has easily been my bread and butter.

**Q. Where do you put calculated members?**  
The reflexive answer is “in the Measures dimension” but this is the obvious answer.  So I always follow up with another question.  “If you want to create a calculated member that intersects all measures, where do you put it?”  A high percentage of candidates can’t answer this question, and the answer is “In a dimension other than Measures.”  If they can answer it, I immediately ask them why.  The answer is “Because a member in a dimension cannot intersect its own relatives in that dimension.”

**Q. How do I find the bottom 10 customers with the lowest sales in 2003 that were not null?**  
Simply using bottom count will return customers with null sales. You will have to combine it with NONEMPTY or FILTER.  
SELECT { [Measures].[Internet Sales Amount] } ON COLUMNS ,  
BOTTOMCOUNT(  
NONEMPTY(DESCENDANTS( [Customer].[Customer Geography].[All Customers]  
, [Customer].[Customer Geography].[Customer] )  
, ( [Measures].[Internet Sales Amount] ) )  
, 10  
, ( [Measures].[Internet Sales Amount] )  
) ON ROWS  
FROM [Adventure Works]  
WHERE ( [Date].[Calendar].[Calendar Year].&[2003] ) ;

**Q. How in MDX query can I get top 3 sales years based on order quantity?**  
By default Analysis Services returns members in an order specified during attribute design. Attribute properties that define ordering are “Order By” and “Order By Attribute”. Lets say we want to see order counts for each year. In Adventure Works MDX query would be:  
SELECT {[Measures].[Reseller Order Quantity]} ON 0  
, [Date].[Calendar].[Calendar Year].Members ON 1  
FROM [Adventure Works];  
Same query using TopCount:  
SELECT  
{[Measures].[Reseller Order Quantity]} ON 0,  
TopCount([Date].[Calendar].[Calendar Year].Members,3, [Measures].[Reseller Order Quantity]) ON 1  
FROM [Adventure Works];

**Q. How do you extract first tuple from the set?**  
Use could usefunctionSet.Item(0)  
Example:  
SELECT {{[Date].[Calendar].[Calendar Year].Members  
}.Item(0)}  
ON 0  
FROM [Adventure Works]

**Q. How can I setup default dimension member in Calculation script?**  
You can use ALTER CUBE statement. Syntax:  
ALTER CUBE CurrentCube | YourCubeName UPDATE DIMENSION , DEFAULT\_MEMBER=”;

**Q. What is data mart?**  
A data mart is a subset of an organizational data store, usually oriented to a specific purpose or major data subject that may be distributed to support business needs. Data marts are analytical data stores designed to focus on specific business functions for a specific community within an organization.  
Data marts are often derived from subsets of data in a**DATA WAREHOUSE**, though in the bottom-up data warehouse design methodology the data warehouse is created from the union of organizational data marts.  
They are 3 types of data mart they are  
1.Dependent  
2.Independent  
3.Logical data mart

**Q. What are the difference between data mart and data warehouse?**  
Datawarehouse is complete data where as Data mart is Subset of the same.  
Ex:  
All the organization data may related to finance department, HR, banking dept are stored in data warehouse where as in data mart only finance data or HR department data will be stored. So data warehouse is a collection of different data marts.

**Q. Have you ever worked on performance tuning, if yes what are the steps involved in it?**  
We need to identify the bottlenecks to tune the performance, to overcome the bottleneck we need to following the following.  
1.Avoid named queries  
2.Unnecessary relationships between tables  
3.Proper attribute relationships to be given  
4.Proper aggregation design  
5.Proper partitioning of data  
6.Proper dimension usage design  
7.Avoid unnecessary many to many relationships  
8.Avoid unnecessary measures  
9.Set Attribute Hierarchy Enabled = FALSE to Attributes that is not required  
10.Won’t take even single measure which is not necessary.

**Q. What are the difficulties faced in cube development?**  
This question is either to test whether you are really experienced or when he does not have any questions to ask ..  
You can tell any area where you feel difficult to work. But always the best answers will be the following.  
1.Giving attribute relationships  
2.Calculations  
3.Giving dimension usage (many to many relationship)  
4.Analyzing the requirements

**Q. Explain the flow of creating a cube?**  
Steps to create a cube in ssas  
1.Create  a data source.  
2.Create a data source view.  
3.Create Dimensions  
4.Create a cube.  
5.Deploy and Process the cube.

**Q. What is a data source or DS?**  
The data source is the Physical Connection information that analysis service uses to connect to the database that host the data. The data source contains the connection string which specifies the server and the database hosting the data as well as any necessary authentication credentials.

**Q. What is data source view or DSV?**  
A data source view is a persistent set of tables from a data source that supply the data for a particular cube. BIDS also includes a wizard for creating data source views, which you can invoke by right-clicking on the Data Source Views folder in Solution Explorer.  
1.Datasource view is the logical view of the data in the data source.  
2.Data source view  is the only thing a cube can see.

**Q. What is named calculation?**  
A named calculation is a SQL expression represented as a calculated column. This expression appears and behaves as a column in the table. A named calculation lets you extend the relational schema of existing tables or views in a data source view without modifying the tables or views in the underlying data source.  
Named calculation is used to create a new column in the DSV using hard coded values or by using existing columns or even with both.

**Q. What is named query?**  
Named query in DSV is similar to View in Database. This is used to create Virtual table in DSV which will not impact the underlying database. Named query is mainly used to merge the two or more table in the datasource view or to filter columns of a table.

**Q. Why we need named queries?**  
A named query is used to join multiple tables, to remove unnecessary columns from a table of a database. You can achieve the same in database using Views but this Named Queries will be the best bet whe you don’t have access to create Views in database.

**Q. How will you add a new column to an existing table in data source view?**  
By using named calculations we can add a new column to an existing table in the data source view. Named Calculation is explained above.

**Q. What is dimension table?**  
A dimension table contains hierarchical data by which you’d like to summarize. A dimension table contains specific business information, a dimension table that contains the specific name of each member of the dimension. The name of the dimension member is called an “attribute”  
The key attribute in the dimension must contain a unique value for each member of the dimension. This key attribute is called “primary key column”  
The primary key column of each dimension table corresponding to the one of the key column  in any related fact table.

**Q. What is fact table?**  
A fact table contains the basic information that you wish to summarize. The table that stores the detailed value for measure is called fact table. In simple and best we can define as “The table which contains METRICS” that are used to analyse the business.  
It consists of 2 sections  
1) Foreign key to the dimension  
2) measures/facts(a numerical value that used to monitor business activity)

**Q. What is Fact less fact table?**  
This is very important interview question. The “Factless Fact Table” is a table which is similar to Fact Table except for having any measure; I mean that this table just has the links to the dimensions. These tables enable you to track events; indeed they are for recording events.  
Factless fact tables are used for tracking a process or collecting stats. They are called so because, the fact table does not have aggregatable numeric values or information. They are mere key values with reference to the dimensions from which the stats can be collected

**Q. What is attribute relationships, why we need it?**  
Attribute relationships are the way of telling the analysis service engine that how the attributes are related with each other. It will help to relate two or more  attributes to each other. Processing time will be decreased if proper relationships are given. This increases the Cube Processing performance and MDX query performance too.  
In Microsoft SQL Server Analysis Services, attributes within a dimension are always related either directly or indirectly to the key attribute. When you define a dimension based on a star schema, which is where all dimension attributes are derived from the same relational table, an attribute relationship is automatically defined between the key attribute and each non-key attribute of the dimension. When you define a dimension based on a snowflake schema, which is where dimension attributes are derived from multiple related tables, an attribute relationship is automatically defined as follows:  
1.Between the key attribute and each non-key attribute bound to columns in the main dimension table.  
2.Between the key attribute and the attribute bound to the foreign key in the secondary table that links the underlying dimension tables.  
3.Between the attribute bound to foreign key in the secondary table and each non-key attribute bound to columns from the secondary table.

**Q. How many types of attribute relationships are there?**  
They are 2 types of attribute relationships they are  
1.Rigid  
2.Flexible  
**Rigid:** In Rigid relationships  where the relationship between the attributes is fixed, attributes will not change levels or their respective attribute relationships.  
Example: The time dimension. We know that month “January 2009″ will ONLY belong to Year “2009″ and it wont be moved to any other year.  
**Flexible:** In Flexible relationship between the attributes is changed.  
Example: An employee and department. An employee can be in accounts department today but it is possible that the employee will be in Marketing department tomorrow.

**Q. How many types of dimensions are there and what are they?**  
They are 3 types of dimensions:  
1.confirm dimension  
2.junk dimension  
3.degenerate attribute

**Q. What are confirmed dimensions, junk dimension and degenerated dimensions?**  
Confirm dimension: It is the dimension which is sharable across the multiple facts or data model. This is also called as Role Playing Dimensions.  
junk dimension: A number of very small dimensions might be lumped (a small irregularly shaped) together to form a single dimension, a junk dimension – the attributes are not closely related. Grouping of Random flags and text Attributes in a dimension and moving them to a separate sub dimension is known as junk dimension.  
Degenerated dimension: In this degenerate dimension contains their values in fact table and the dimension id not available in dimension table. Degenerated Dimension is a dimension key without corresponding dimension.  
Example: In the PointOfSale Transaction Fact table, we have:  
Date Key (FK), Product Key (FK), Store Key (FK), Promotion Key (FP), and POS Transaction Number  
Date Dimension corresponds to Date Key, Production Dimension corresponds to Production Key. In a traditional parent-child database, POS Transactional Number would be the key to the transaction header record that contains all the info valid for the transaction as a whole, such as the transaction date and store identifier. But in this dimensional model, we have already extracted this info into other dimension. Therefore, POS Transaction Number looks like a dimension key in the fact table but does not have the corresponding dimension table.

**Q. What are the types of database schema?**  
They are 3 types of database schema they are  
1.Star  
2.Snowflake  
3.Starflake

**Q. What is star, snowflake and star flake schema?**  
**Star schema:** In star schema fact table will be directly linked with all dimension tables. The star schema’s dimensions are denormalized with each dimension being represented by a single table. In a star schema a central fact table connects a number of individual dimension tables.  
**Snowflake:** The snowflake schema is an extension of the STAR SCHEMA, where each point of the star explodes into more points. In a star schema, each dimension is represented by a single dimensional table, whereas in a snowflake schema, that dimensional table is normalized into multiple lookup tables, each representing a level in the dimensional hierarchy. In snow flake schema fact table will be linked directly as well as there will be some intermediate dimension tables between fact and dimension tables.  
**Star flake:** A hybrid structure that contains a mixture of star(denormalized) and snowflake(normalized) schema’s.

**Q. How will you hide an attribute?**  
We can hide the attribute by selecting “AttributeHierarchyVisible = False” in properties of the attribute.

**Q. How will you make an attribute not process?**  
By selecting  “ AttributeHierarchyEnabled = False”, we can make an  attribute not in process.

**Q. What is use of IsAggregatable property?**  
In Analysis Service we generally see all dimension has All member. This is because of IsAggregatable property of the attribute. You can set its value to false, so that it will not show All member. Its default member for that attribute. If you hide this member than you will have to set other attribute value to default member else it will pick some value as default and this will create confusion in browsing data if someone is not known to change in default member.

**Q. What are key, name and value columns of an attribute?**  
**Key column of any attribute:** Contains the column or columns that represent the key for the attribute, which is the column in the underlying relational table in the data source view to which the attribute is bound. The value of this column for each member is displayed to users unless a value is specified for the NameColumn property.  
**Name  column of an attribute:** Identifies the column that provides the name of the attribute that is displayed to users, instead of the value in the key column for the attribute. This column is used when the key column value for an attribute member is cryptic or not otherwise useful to the user, or when the key column is based on a composite key. The NameColumn property is not used in parent-child hierarchies; instead, the NameColumn property for child members is used as the member names in a parent-child hierarchy.  
**Value columns of an attribute:** Identifies the column that provides the value of the attribute. If the NameColumn element of the attribute is specified, the same DataItem values are used as default values for the ValueColumn element. If the NameColumn element of the attribute is not specified and the KeyColumns collection of the attribute contains a single KeyColumn element representing a key column with a string data type, the same DataItem values are used as default values for the ValueColumn element.

**Q. What is hierarchy, what are its types and difference between them?**  
A hierarchy is a very important part of any OLAP engine and allows users to drill down from  summary levels hierarchies represent the way user expect to explore data at more detailed level  
hierarchies  is made up of multipule levels creating the structure based on end user requirements.  
->years->quarter->month->week ,are all the levels of calender hierarchy  
They are 2 types of hierarchies they are  
1.Natural hierarchy  
2.Unnatural hierarchy  
**Natural hierarchy:** This means that the attributes are intuitively related to one another. There is a clear relationship from the top of the hierarchy to the bottom.  
Example: An example of this would be date: year, quarter and month follow from each other, and in part, define each other.  
**Unnatural hierarchy:**This means that the attributes are not clearly related.  
Example: An example of this might be geography; we may have country -> state -> city, but it is not clear where Province might sit.

**Q. What is Attribute hierarchy?**  
An attribute hierarchy is created for every attribute in a dimension, and each hierarchy is available for dimensioning fact data. This hierarchy consists of an “All” level and a detail level containing all members of the hierarchy.  
you can organize attributes into user-defined hierarchies to provide navigation paths in a cube. Under certain circumstances, you may want to disable or hide some attributes and their hierarchies.

**Q. What is use of AttributeHierarchyDisplayFolder property ?**  
**AttributeHierarchyDisplayFolder:**Identifies the folder in which to display the associated attribute hierarchy to end users. For example if I set the property value as “Test” to all the Attributes of a dimension then a folder with the name “Test” will be created and all the Attributes will be placed into the same.

**Q. What is use of AttributeHierarchyEnabled?**  
**AttributeHierarchyEnabled:** Determines whether an attribute hierarchy is generated by Analysis Services for the attribute. If the attribute hierarchy is not enabled, the attribute cannot be used in a user-defined hierarchy and the attribute hierarchy cannot be referenced in Multidimensional Expressions (MDX) statements.

**Q. What is use of AttributeHierarchyOptimizedState?**  
**AttributeHierarchyOptimizedState:** Determines the level of optimization applied to the attribute hierarchy. By default, an attribute hierarchy is FullyOptimized, which means that Analysis Services builds indexes for the attribute hierarchy to improve query performance. The other option, NotOptimized, means that no indexes are built for the attribute hierarchy. Using NotOptimized is useful if the attribute hierarchy is used for purposes other than querying, because no additional indexes are built for the attribute. Other uses for an attribute hierarchy can be helping to order another attribute.

**Q. What is use of AttributeHierarchyOrdered ?**  
b Determines whether the associated attribute hierarchy is ordered. The default value is True. However, if an attribute hierarchy will not be used for querying, you can save processing time by changing the value of this property to False.

**Q. What is the use of AttributeHierarchyVisible ?**  
**AttributeHierarchyVisible :**Determines whether the attribute hierarchy is visible to client applications. The default value is True. However, if an attribute hierarchy will not be used for querying, you can save processing time by changing the value of this property to False.

**Q. What are types of storage modes?**  
There are three standard storage modes in OLAP applications  
1.MOLAP  
2.ROLAP  
3.HOLAP

**Q. What is MOLAP and its advantage?**  
MOLAP (Multi dimensional Online Analytical Processing) : MOLAP is the most used storage type. Its designed to offer maximum query performance to the users. the data and aggregations are stored in a multidimensional format, compressed and optimized for performance. This is both good and bad. When a cube with MOLAP storage is processed, the data is pulled from the relational database, the aggregations are performed, and the data is stored in the AS database. The data inside the cube will refresh only when the cube is processed, so latency is high.  
Advantages:  
1.Since the data is stored on the OLAP server in optimized format, queries (even complex calculations) are faster than ROLAP.  
2.The data is compressed so it takes up less space.  
3.And because the data is stored on the OLAP server, you don’t need to keep the connection to the relational database.  
4.Cube browsing is fastest using MOLAP.

**Q. What is ROLAP and its advantage?**  
ROLAP (Relational Online Analytical Processing) : ROLAP does not have the high latency disadvantage of MOLAP. With ROLAP, the data and aggregations are stored in relational format. This means that there will be zero latency between the relational source database and the cube.  
Disadvantage of this mode is the performance, this type gives the poorest query performance because no objects benefit from multi dimensional storage.  
Advantages:  
1.Since the data is kept in the relational database instead of on the OLAP server, you can view the data in almost real time.  
2.Also, since the data is kept in the relational database, it allows for much larger amounts of data, which can mean better scalability.  
3.Low latency.

**Q. What is HOLAP and its advantage?**  
Hybrid Online Analytical Processing (HOLAP): HOLAP is a combination of MOLAP and ROLAP. HOLAP stores the detail data in the relational database but stores the aggregations in multidimensional format. Because of this, the aggregations will need to be processed when changes are occur. With HOLAP you kind of have medium query performance: not as slow as ROLAP, but not as fast as MOLAP. If, however, you were only querying aggregated data or using a cached query, query performance would be similar to MOLAP. But when you need to get that detail data, performance is closer to ROLAP.  
Advantages:  
1.HOLAP is best used when large amounts of aggregations are queried often with little detail data, offering high performance and lower storage requirements.  
2.Cubes are smaller than MOLAP since the detail data is kept in the relational database.  
3.Processing time is less than MOLAP since only aggregations are stored in multidimensional format.  
4.Low latency since processing takes place when changes occur and detail data is kept in the relational database.

**Q. What are Translations and its use?**  
**Translation:** The translation feature in analysis service allows you to display caption and attributes names that correspond to a specific language. It helps in providing GLOBALIZATION to the Cube.

**Q. What is Database dimension?**  
All the dimensions that are created using NEW DIMENSION Wizard are database dimensions. In other words, the dimensions which are at Database level are called Database Dimensions.

**Q. What is Cube dimension?**  
A cube dimension is an instance of a database dimension within a cube is called as cube dimension. A database dimension can be used in multiple cubes, and multiple cube dimensions can be based on a single database dimension

**Q. Difference between Database dimension and Cube dimension?**  
1.The Database dimension has only Name and ID properties, whereas a Cube dimension has several more properties.  
2.Database dimension is created one where as Cube dimension is referenced from database dimension.  
3.Database dimension exists only once.where as Cube dimensions can be created more than one using ROLE PLAYING Dimensions concept.

**Q. How will you add a dimension to cube?**  
To add a dimension to a cube follow these steps.  
1.In Solution Explorer, right-click the cube, and then click View Designer.  
2.In the Design tab for the cube, click the Dimension Usage tab.  
3.Either click the Add Cube Dimension button, or right-click anywhere on the work surface  and then click Add Cube Dimension.  
4.In the Add Cube Dimension dialog box, use one of the following steps:  
5.To add an existing dimension, select the dimension, and then click OK.  
6.To create a new dimension to add to the cube, click New dimension, and then follow the steps in the Dimension Wizard.

**Q. What is SCD (slowly changing dimension)?**  
Slowly changing dimensions (SCD) determine how the historical changes in the dimension tables are handled. Implementing the SCD mechanism enables users to know to which category an item belonged to in any given date.

**Q. What are types of SCD?**   
It is a concept of STORING Historical Changes and whenever an IT guy finds a new way to store then a new type will come into picture. Basically there are 3 types of SCD they are given below   
1.SCD type1  
2.SCD type2  
3.SCD type3

**Q. What is role playing dimension with two examples?**  
**Role play dimensions:** We already discussed about this. This is nothing but CONFIRMED Dimensions. A dimension can play different role in a fact table you can recognize a roleplay dimension when there are multiple columns in a fact table that each have foreign keys to the same dimension table.  
Ex1: There are three dimension keys in the fact internet sales, fact reseller sales tables which all refer to the dimtime table, the same time dimension is used  to track sales by that contain either of these fact table, the corresponding  role-playing dimension are automatically added to the cube.  
Ex2 : In retail banking, for checking account cube we could have transaction date dimension and effective date dimension. Both dimensions have date, month, quarter and year attributes. The formats of attributes are the same on both dimensions, for example the date attribute is in ‘dd-mm-yyyy’ format. Both dimensions have members from 1993 to 2010.

**Q. What is measure group, measure?**  
**Measure groups :** These measure groups can contain different dimensions and be at different granularity  but so long as you model your cube correctly, your users will be able to use measures from each of these measure groups in their queries easily and without worrying about the underlying complexity.  
Creating multiple measure groups : To create a new measure group in the Cube Editor, go to the Cube Structure tab and right-click on the cube name in the Measures pane and select ‘New Measure Group’. You’ll then need to select the fact table to create the measure group from and then the new measure group will be created; any columns that aren’t used as foreign key columns in the DSV will automatically be created as measures, and you’ll also get an extra measure of aggregation type Count. It’s a good idea to delete any measures you are not going to use at this stage.  
**Measures :**  Measures are the numeric values that our users want to aggregate, slice, dice and otherwise analyze, and as a result, it’s important to make sure they behave the way we want them to. One of the fundamental reasons for using Analysis Services is that, unlike a relational database it allows us to build into our cube design business rules about measures: how they should be formatted, how they should aggregate up, how they interact with specific dimensions and so on.

**Q. What is attribute?**  
An attribute is a specification that defines a property of an object, element, or file. It may also refer to or set the specific value for a given instance of such.

**Q. What is surrogate key?**  
A surrogate key is the SQL generated key which acts like an alternate primary key for the table in database, Data warehouses commonly use a surrogate key to uniquely identify an entity. A surrogate is not generated by the user but by the system. A primary difference between a primary key and surrogate key in few databases is that primary key uniquely identifies a record while a Surrogate key uniquely identifies an entity.  
Ex: An employee may be recruited before the year 2000 while another employee with the same name may be recruited after the year 2000. Here, the primary key will uniquely identify the record while the surrogate key will be generated by the system (say a serial number) since the SK is NOT derived from the data.

**Q. How many types of relations are there between dimension and measure group?**  
There are six relations between the dimension and measure group, they are  
1.No Relationship  
2.Regular  
3.Refernce  
4.Many to Many  
5.Data Mining  
6.Fact

**Q. What is regular type, no relation type, fact type, referenced type, many-to-many type with example?**  
**No relationship:** The dimension and measure group are not related.  
**Regular:**The dimension table is joined directly to the fact table.  
**Referenced:** The dimension table is joined to an intermediate table, which in turn, is joined to the fact table.  
**Many to many:** The dimension table is to an intermediate fact table, the intermediate fact table is joined, in turn, to an intermediate dimension table to which the fact table is joined.  
**Data mining:** The target dimension is based on a mining model built from the source dimension. The source dimension must also be included in the cube.  
**Fact table:**The dimension table is the fact table.

**Q. What are calculated members and what is its use?**  
Calculations are item in the cube that are evaluated at runtime  
Calculated members: You can create customized measures or dimension members, called calculated members, by combining cube data, arithmetic operators, numbers, and/or functions.  
Example: You can create a calculated member called Marks that converts dollars to marks by multiplying an existing dollar measure by a conversion rate. Marks can then be displayed to end users in a separate row or column. Calculated member definitions are stored, but their values exist only in memory. In the preceding example, values in marks are displayed to end users but are not stored as cube data.

**Q. What are KPIs and what is its use?**  
In Analysis Services, a KPI is a collection of calculations that are associated with a measure group in a cube that are used to evaluate business success. We use KPI to see the business at the particular point, this is represents with some graphical items such as traffic signals, ganze etc

**Q. What are actions, how many types of actions are there, explain with example?**  
Actions are powerful way of extending the value of SSAS cubes for the end user. They can click on a cube or portion of a cube to start an application with the selected item as a parameter, or to retrieve information about the selected item.  
One of the objects supported by a SQL Server Analysis Services cube is the action. An action is an event that a user can initiate when accessing cube data. The event can take a number of forms. For example, a user might be able to view a Reporting Services report, open a Web page, or drill through to detailed information related to the cube data  
Analysis Services supports three types of actions..  
**Report action:**Report action Returns a Reporting Services report that is associated with the cube data on which the action is based.  
**Drill through:**Drill through Returns a result set that provides detailed information related to the cube data on which the action is based.  
**Standard:** Standard has five action subtypes that are based on the specified cube data.  
Dataset: Returns a multidimensional dataset.  
**Proprietary:** Returns a string that can be interpreted by a client application.  
**Row set:** Returns a tabular row set.  
**Statement:** Returns a command string that can be run by a client application.  
**URL:** Returns a URL that can be opened by a client application, usually a browser.

**Q. What is partition, how will you implement it?**  
You can use the Partition Wizard to define partitions for a measure group in a cube. By default, a single partition is defined for each measure group in a cube. Access and processing performance, however, can degrade for large partitions. By creating multiple partitions, each containing a portion of the data for a measure group, you can improve the access and processing performance for that measure group.

**Q. What is the minimum and maximum number of partitions required for a measure group?**  
In 2005 a MAX of 2000 partitions can be created per measure group and that limit is lifted in later versions.  
In any version the MINIMUM is ONE Partition per measure group.

**Q. What are Aggregations and its use?**  
Aggregations provide performance improvements by allowing Microsoft SQL Server Analysis Services (SSAS) to retrieve pre-calculated totals directly from cube storage instead of having to recalculate data from an underlying data source for each query. To design these aggregations, you can use the Aggregation Design Wizard. This wizard guides you through the following steps:  
1.Selecting standard or custom settings for the storage and caching options of a partition, measure group, or cube.  
2.Providing estimated or actual counts for objects referenced by the partition, measure group, or cube.  
3.Specifying aggregation options and limits to optimize the storage and query performance delivered by designed aggregations.  
4.Saving and optionally processing the partition, measure group, or cube to generate the defined aggregations.  
5.After you use the Aggregation Design Wizard, you can use the Usage-Based Optimization Wizard to design aggregations based on the usage patterns of the business users and client applications that query the cube.

**Q. What is perspective, have you ever created perspective?**  
Perspectives are a way to reduce the complexity of cubes by hidden elements like measure groups, measures, dimensions, hierarchies etc. It’s nothing but slicing of a cube, for ex we are having retail and hospital data and end user is subscribed to see only hospital data, then we can create perspective according to it.

**Q. What is deploy, process and build?**  
**Bulid:**Verifies the project files and create several local files.  
**Deploy:** Deploy the structure of the cube(Skeleton) to the server.  
**Process:** Read the data from the source and build the dimesions and cube structures  
Elaborating the same is given below.  
**Build:** Its is a used to process the data of the cube database. Build is a version of a**PROGRAM.**As a rule, a build is a pre-release version and as such is identified by a build number, rather than by a release number. Reiterative (repeated) builds are an important part of the development process. Throughout development, application components are collected and repeatedly **COMPILED** for testing purposes, to ensure a reliable final product. Build tools, such as **MAKE** or**ANT**, enable developers to automate some programming tasks. As a verb, to build can mean either to write **CODE**or to put individual coded components of a program together.  
**Deployment:**During development of an Analysis Services project in Business Intelligence Development Studio, you frequently deploy the project to a development server in order to create the Analysis Services database defined by the project. This is required to test the project.  
for example, to browse cells in the cube, browse dimension members, or verify key performance indicators (KPIs) formulas.

**Q. What is the maximum size of a dimension?**  
The maximum size of the dimension is **4 gb.**

**Q. What is a cube?**  
The basic unit of storage and analysis in Analysis Services is the cube. A cube is a collection of data that’s been aggregated to allow queries to return data quickly.  
For example, a cube of order data might be aggregated by time period and by title, making the cube fast when you ask questions concerning orders by week or orders by title.

**Q. What is AMO?**  
The full form of AMO is Analysis Managament Objects. This is used to create or alter cubes from .NET code.

# **SSRS Interview Questions**

# **Q. What is SSRS?**

1. SSRS or SQL Server Reporting Service is a server-based report generation software systems from Microsoft and is part of Microsoft BI.
2. It is used for preparing and delivering interactive and variety of reports.
3. It is administered through an web based interface.
4. Reporting services utilizes a web service interface for supporting and developing of customized reporting applications.
5. SSRS lets you create very rich reports (Tabular/Graphical/Interactive) from various data sources with rich data visualization (Charts, Maps, sparklines)
6. SSRS allows are reports to be exported in various formats (Excel, PDF, word etc)

**Q. Explain SSRS Architecture?**

Reporting services architecture comprises of integrated components. It is a multi-tiered, included with application, server and data layers. This architecture is scalable and modular. A single installation can be used across multiple computers. It includes the following components: –  
[Report Manager](https://mindmajix.com/msbi/how-to-access-report-manager-in-ssrs-2012), Reporting Designer, Browser Types Supported by Reporting services, Report server, Report server command line utilities, Report Server Database, Reporting Services Extensibility, Data sources that is supported by Reporting Services.

**Q: Explain Reporting Life Cycle?**

Reporting Services has mainly three phases:

1. **Development of Reports (Developer)**– First of all a report needs to be design which is primarily done by report developer
2. **Management of Reports (DBA)** – Once the Report is being developed, DBA need to ensure
3. **Security** – Only authorized user should access the report
4. **Execution** – How the report will be executed to optimize data sources performance
5. **Scheduling of reports** – so that report are executed on scheduled timings
6. **Report Delivery (DBA + Developer)**– Once the report is being developed and executed now the report should be reached to final recipients (business users) who are going to understand / analyze report data. if any changes, we again go back to development stage.

**Q. What are the Reporting Services components?**

Reporting services components assist in development. These processing components include some tools that are used to create, manage and view reports.

1. **Report Designer** is used to create the reports.
2. **Report Sever** is used to execute and distribute reports.
3. **Report Manager** is used to manage the report server.

**Q. SQL Server Reporting Services vs. Crystal Reports.**

Crystal reports are processed by IIS while SSRS have a report server. Caching in Crystal reports is available through cache server. On the other hand, caching in SSRS is available for Report history snapshots. Crystal reports have standards and user defined field labels. SSRS allows only user defined field labels.

**Q. How does the report manager work in SSRS?**

Report manager is a web application. In SSRS it is accessed by a URL. The interface of this Report manager depends on the permissions of the user. This means to access any functionality or perform any task, the user must be assigned a role. A user with a role of full permissions can entire all the features and menus of the report. To configure the report manager, a URL needs to be defined.

**Q. How can I add Reporting Services reports to my application?**

Visual Studio / SSDT / BI Data Tools (Standard and Enterprise editions) contains a set of freely redistributable Report Viewer controls that make it easy to embed Reporting Services functionality into custom applications. Two versions of the Report Viewer exist, one for rich Windows client applications and one for ASP.NET applications.

**Q. Do I need a report server to run reports in my application?**

In addition to publishing reports to a report server, you can build reports using the Report Designer that is directly integrated with Visual Studio language projects. You can embed reports directly in any Windows Forms or [ASP.NET](https://mindmajix.com/asp-net-training) Web application without access to a report server. The data access in embedded reports is a natural extension of the Visual Studio data facilities. Not only can you use traditional databases as a source of data for your reports, you can use object collections as well.

**Q. What are different types of roles provided by SSRS?**

1. Browsers
2. Content Manager
3. My Reports
4. Publishers
5. Report Builder

**Q. Name and Describe few console utilities for SSRS?**

RSConfig.exe: Configuration of connection properties between the Report Server to the repository database.

* RSKeyMgmt.exe: Management of encryption keys via command-line
* RS.exe: Utility used for deploying the report on report server

**Q. What are the drawbacks of reporting in SSRS?**

For many years, Microsoft had no direct solution for reporting with the SQL Server besides Crystal Reports. Now, they have SQL Server Reporting Services, but it does have several drawbacks. It is still complex to understand the complete functionality and structure of this new component, and many users are still relying on the reporting application they are more familiar with, which is Crystal Reports. Also, components in SSRS like Report Builder and Report Designer are meant for different users for different aspects of the report process, yet complete understanding and exposure to both is important to utilize both functions fully and extensively. There are also issues when exporting very large reports to Microsoft Excel, as it can lead to a loss of data.

**Q. What are the three different part of RDL file explain them?**

In visual studio RDL files has three parts.

1. **Data:** It contains the dataset on which we write the query. Data set is connected with data source.
2. **Design:** In design you can design report. Can create tables and matrix reports. Drag columns values from source.
3. **Preview**: to check the preview after the report run.

**Q. Which language rdl files made of?**

RDL files are written in XML.

**Q. Can you edit the .rdl code associated with a linked report?**

No, because a linked report has no .rdl code of its own. It refers to the .rdl code of the base report

**Q. What is report rendering ?**

Exporting a report data with design o different type of file types is knows as Report rending. SQL Server Reporting Services supports multiple rendering extensions like Word, Excel, CSV, PDF, HTML etc.

**Q. What are the different types of data sources in SSRS?**

SSRS use different data source. Some of them are listed below.

1. Microsoft SQL Server
2. OLEDB
3. Oracle
4. ODBC
5. SQL Server Analysis Service
6. Report Server Model
7. SAP Net weaver BI
8. Hyperion
9. Teradata
10. XML

**Q. What is the name of reporting services config file and what’s it’s used for?**

Reporting service config file is used for report configuration details. It contains the report format and also the report import types. Report service config reside at ISS.

## SSRS Scenario Based Interview Questions

**Q. What are Advantages of SSRS or why we should use SSRS?**

The SQL Server Reporting Services or SSRS has some Advantages

1. It is faster and cheaper
2. Efficient reporting access to information residing in both Oracle and MS SQL Server databases
3. No need for expensive specialist skills
4. The default report designer is integrated with Visual Studio .NET so that we can create application and its reports in the same environmen
5. The security is managed in a role-based manner and can be applied to folders as well as reports
6. Once parameters are defined, the UI for these parameters is automatically generated

Subscription-based reports are automatically sent by mail to the users

**Q. What are the limitations/drawbacks or SSRS 2008 R2?**

The SSRS 2008 R2 has some limitations. Some limitations are given below:

1. There is no print button. In order to print need to export excel, PDF or others format
2. It is very hard to debug expression or custom code
3. ts does not use page number or total pages in report body
4. Don’t support rounding rectangle
5. There is no way to pass values from sub-reports to main report
6. It is not possible to insert a sub-report into the page header or page footer
7. Page header creates extra spaces in the next pages

**Q. What are the Export Options of SSRS?**

SSRS allow many ways of rendering the reports:

1. HTML (MHTML)
2. Excel
3. Acrobat
4. Tiff (image)
5. XML
6. CSV

**Q. What is report snapshot?**

Snapshot means a instance of a report for future reference, that means a copy of report (data is freezed) will be saved on a report server for future reference.

**Q. What are Data Driven Subscriptions?**

Reporting Services provides data-driven subscriptions so that you can customize the distribution of a report based on dynamic subscriber data. Data-driven subscriptions are intended for the following kinds of scenarios: Distributing reports to a large recipient pool whose membership may change from one distribution to the next. For example distribute a monthly report to all current customers. Distributing reports to a specific group of recipients based on predefined criteria. For example send a sales performance report to the top ten sales managers in an organization.

**Q. When to Use Null Data driven Subscription?**

Create a data-driven subscription that uses the Null Delivery Provider. When you specify the Null Delivery Provider as the method of delivery in the subscription, the report server targets the report server database as the delivery destination and uses a specialized rendering extension called the null rendering extension. In contrast with other delivery extensions, the Null Delivery Provider does not have delivery settings that you can configure through a subscription definition.

**Q. How to fine-tune Reports?**

To tune-up the Reporting Services, follow the below mentioned ways: – Expand the Server or utilizing the reporting services of another database server. For better embedding of report contents, report application’s logic and characteristics can have a duplicate copy of data. – Replication of data continuously. Using (nolock), the issues of locking can well be resolved and the performance of the query can be improved. This can be done by using dirty read at the time of duplicating the data is unavailable.

**Q. User wants only to display only PDF as export option in report Manager. How to achieve this?**

You need to edit RsReportServer.Config file to limit the exporting extensions to only PDF.  
You can find the file under;  
%Program Files%Microsoft SQL ServerMSRS10.Reporting ServicesReportServer

**Q. Can you import Microsoft Excel data to SSRS?**

Reporting Services does not import data. It only queries data in whatever format it is stored in their native storage system. I will assume that you’re asking whether you can create reports and use Excel spreadsheets as data sources. The answer is Yes, Reporting Services supports a wide variety of data sources, including Excel files. You’ll get the best performance with the built-in native .NET providers but you should be able to connect to any ODBC or OLE-DB data source, whether it comes from Microsoft or a third-party company.

**Q. Difference between Logical Page and Physical Page in SSRS.**

Logical page breaks are page breaks that you insert before or after report items or groups. Page breaks help to determine how the content is fitted to a report page for optimal viewing when rendering or exporting the report. The following rules apply when rendering logical page breaks: Logical page breaks are ignored for report items that are constantly hidden and for report items where the visibility is controlled by clicking another report item. Logical page breaks are applied on conditionally visible items if they are currently visible at the time the report is rendered. Space is preserved between the report item with the logical page break and its peer report items. Logical page breaks that are inserted before a report item push the report item down to the next page. The report item is rendered at the top of the next page. Logical page breaks defined on items in table or matrix cells are not kept. This does not apply to items in lists.

**Q. Name few Endpoints exposed by SSRS 2012?**

1. Management Endpoints
2. Execution Endpoint
3. SharePoint Proxy Endpoints

**Q. What are the new features are introduced in SQL Server 2012 reporting services?**

1. Power View – interactive data exploration
2. SQL Server 2012 is fully integrated with SharePoint
3. Introduction to Data Alerts, data alerts are a data-driven alerting solution that informs you about changes in report data that are of interest to you, and at a relevant time
4. SQL Server Data tool
5. New rendering extensions supports MS Office 2010
6. Project Crescent is being introduced

**Q. What new data source types were added in SSRS 2014?**

In addition to the data source types available in SSRS (SQL Server, Oracle, ODBC, OLE DB), the following have been added in SSRS 2012: SQL Server Analysis Services SQL Server Integration Services SQL Server Report Builder Models XML (through URL and Web services)

**Q. Is SSRS support other database except MS SQL Server?**

Yes. SSRS can be building based on relational or multidimensional data source like Oracle, OLEDB. ODBC etc

**Q. What is Query parameter in SSRS?**

Query parameters is mentioned in the query of the datasources that are to be included into the SQL script’s WHERE clause of the SQL that can accept parameters. Query parameters begin with the symbol @.The name should not contain spaces and can not begin with numeral. For clarity, we use only letters.

**Q. What is a matrix in SSRS?**

A matrix is a data region linked to a report set. Matrix allows us to create crosstab reports with the report variables displaying on rows and columns. It allows us to drag and drop fields into it.

**Q. What are sub reports and how to create them?**

A sub report is like any other reports which can be called in main report and can be generate through main report. Parameters can be passed from main report to sub report and basis of that report can be generated.

**Q. What is the chart in report?**

Chart reports are for graphical representation. You can get pie charts columns harts and various other options. 3d charts are also available in reporting services.

**Q. How to add the custom code in Report?**

To add the custom codes in report go to report tab on top then properties and there you will find the options for custom code.

**Q. In case you have filters in your report, when filters will be applied in Cached Report instance?**

Filters are applied when a report is rendered, Filters will not create a new cached instance on the Report Server.

**Q. What are data regions?**

Data regions are report items that display repeated rows of summarized information from datasets.

**Q. You want to generate a report that is formatted as a chart. Can you use the Report Wizard to create such a report?**

No, the Report Wizard lets you create only tabular and matrix reports. you must create the chart report directly by using the Report Designer.

**Q. Can we use datagrids for our report in SSRS?**

We have an ASP.NET project that populates a datagrid. Using datagrid as my datasource for my report using SQL Server Reporting Services. Is this possible? The simple answer is no. However, nothing’s ever simple. A set of reporting controls was added in Visual Studio 2010 allowing you to report in a dataset, on data that was supplied by you. So, if you retrieved your data into a dataset, bound the datagrid to the dataset so it had data to display, you could then use that dataset as the datasource for the reporting controls. These are then client-side reports, not server reports though.

**Q. Describe different Processing Modes offered by SSRS?**

1. **Local Processing Mode:** Processes reports in the client application.
2. **Remote Processing Mode:** Renders server reports that are processed on a SQL Server Reporting Services report server.

**Q. What is ReportServer and ReportServerTempDB?**

Reporting Services uses two SQL Server databases for storage by default, the databases are named ReportServer and ReportServerTempdb.

ReportServer is a main database, which store all internal configuration and report meta data whereas ReportServerTempdb is used to store temporary data, session information, and cached reports.

**Q. What is encryption key?**

Encryption keys are used by the report server so that items such as connection strings are maintained securely. These keys are required in case you want to perform restoration of report server databases

[**Check Out SSRS Tutorials**](https://mindmajix.com/ssrs-tutorial)

**Q. How to backup encryption key ?**

Encryption Keys backup, use SQL Server [Reporting Services Configuration](https://mindmajix.com/msbi/reporting-services-configuration-files-ssrs) tool to backup symmetric keys.

**Q. What are the key configuration files for SQL Server Reporting Services ?**

Mostly all Configuration files located in Install Directory:  
Microsoft SQL ServerReporting ServicesReportServer and  ReportManager

* **RSReportServer.config**stores configuration settings for feature areas of the Report Server service: ReportManager, the Report Server Web service, and background processing.
* **RSSrvPolicy.config**stores code access security policies for the server extensions.
* **RSMgrPolicy.config** stores code access security policies for Report Manager
* ReportingServicesService.exe.config stores configuration settings that specify the trace levels and logging options for the Report Server service.
* **RSReportDesigner.config** contains settings for Report Designer and this file is located in the Program FilesVisual Studio 9.0Common7IDEPrivateAssemblies
* **RSPreviewPolicy.config** stores server extensions used during report preview and this file is located in Program Files Microsoft SQL Server100Tools ReportDesigner

### **Q. What is SQL Server Report Builder?**

Report Builder is a business-user, ad-hoc report design client that allows users to design reports based on the business terms (Report Builder model) they are familiar with, but without needing to understand database schemas or how to write SQL or MDX queries. Report Builder works with both SQL Server and Analysis Services data sources.

**Q. In which SQL Server version report builder introduced?**

Report builder introduced in SQL Server 2005. While creating or deploying report model project on report server you can get error or it might not get created. For this you need to check whether the service pack 22 is installed or not.

**Q. How does Report Builder support Analysis Services cubes?**

Report Builder supports relational SQL and Analysis Services data sources in SQL Server. To create a model for Analysis Services cube, go to Report Manager or Management Studio, create a data source for your Analysis Services database, and then select the Generate Model option to create the model.

**Q. How do users use Report Builder with SQL Server data sources?**

While models that provide access to SQL Server Analysis Services are automatically generated on the report server, the Report Builder Model Designer can be used to generate or modify the models that are built on top of SQL Server relational databases. These model-building projects are a new type of project within a Visual Studio–based development shell.

**Q. How do I get Report Builder to generate a parameter that can be set by users viewing the report?**

In the filter dialog box, click the name of the criteria that you would like to prompt the user for when viewing the report. For example, for the criteria Order Year=2000, click Order Year. Select the Prompt option in the drop-down list.

**Q. Can we run Reporting Services with SQL Server Express edition, which is a free version of SQL Server?**

Yes, we can. SQL Server Express Edition with Advanced Services support Reporting Services. These is the free version.

**Q. What are the limitations in SSRS on SQL Server express edition?**

Microsoft offers reporting services free as part of SQL Server Express with Advance Services edition. But it has the following limitations:

1. Management Studio cannot be used to administer report server
2. Report Models will not be available
3. Report Builder is not available
4. Caching, History and Delivery of Report is not available.
5. SQL Server agent is not available
6. No scheduling is possible
7. Remote server database is not available for Report Data Source (Local SQL Server is a only option,)
8. We cannot store the report server database on a remote server (it has to be local only)
9. Reports can be rendered only in Excel, PDF, Image formats only
10. Reporting Services will not be able to use more than 1 GB of RAM
11. No Subscriptions (Standard and Data Driven) can be made
12. Can not be integrated with Share Point
13. Can not implement Role based security
14. Only named instances is supported
15. Scale-out Report Servers will not be available

**Q. What are the tools available in market as an Alternative to SQL Server Reporting Services?**

Non-Open Source:

1. Actuate
2. Hyperion (BRIO)
3. SIEBEL-CRM
4. BusinessObjects
5. Oracle Express OLAP
6. Qlikview
7. Cognos
8. Informatica Power Analyzer
9. Proclarity
10. IntelliView
11. Dundas Chart for .NET
12. MS-Excel
13. SAS
14. MicroStrategies
15. Pentaho

Open Source:

1. Jasper Reports
2. JFreeReport
3. BIRT (Business Intelligence Reporting Tools)
4. OpenReport
5. DataVision
6. Pentaho

**Q. How to deploy the Report?**

We can deploy SSRS report in three ways.

1. **Using Visual Studio:** In visual studio we can directly deploy the report through Solution explorer by providing the report server URL in project properties at Target Server URL. As our choice this will deploy entire project or single report as.
2. **Using Report Server:** We can directly go to the report server and deploy the report by browsing the report from the disk location of server.
3. **Creating the Utility:** SQL server provides the facilities to Create a customize utility to deploy the report.

**Q. What are the new features of SQL Server 2008 R2 reporting service?**

The SQL Server 2008 R2 has introduced a lot of new features. Some of them are given below:

1. New Report Types – Table, Matrix, List, Chart, and Sub report
2. Some New Tools is added to report designer Toolbox
3. Report Data Panel – built in page numbers
4. Report Builder 3.0

**Q. What are the new features of SQL Server 2012 reporting service?**

The SQL Server 2012 has introduced a lot of new features. Some of them are given below:

1. Power View – interactive data exploration
2. SharePoint integration
3. Introduction to Data Alerts
4. SQL Server Data tool
5. New rendering extensions (supports MS Office 2010)
6. Project Crescent is being introduced

**Q. How to backup SQL Server Reporting Services ?**

There are mainly three things, which should be backed up as part of reporting services backup

1. **Report Server Databases,**which can be backed up by SQL server backup and restore method.
2. **SQL Server Reporting Services Configuration,** SQL Server Reporting Services Configuration is saved in config files, which can be copied as part of backup. look for other to know config files and there location.
3. **Encryption Keys backup**, use SQL Server Reporting Services Configuration tool to backup symmetric keys.

**Q. What is the web service used for reporting services?**

Reporting Service Web Service used in SSRS. By accessing this web service you can access all report server component and also get the report deployed on report server.

**Q. What is a cache in SSRS?**

Report server can lay up a copy of processed report in a memory and return the copy when a user opens the report. This server memory is known as cache and the process is called caching.

**Q. Can you always create a cache of a report?**

No, you can create a cache of a report only when certain requirements, such as having credentials stored in the Report Server, are met.

**Q. What are the Types of SSRS?**

The types of SSRS are given below:

1. Parameterized reports
2. Linked reports
3. Snapshot reports
4. Cached reports
5. Ad hoc reports
6. Clickthrough reports
7. Drilldown reports
8. Drill through reports
9. Sub reports

**Q. What is Parameterized Reports in SSRS?**

A parameterized report uses input values to complete report or data processing. With a parameterized report, you can vary the output of a report based on values that are set when the report runs. Parameterized reports are frequently used for Drill through reports, linked reports, and sub reports, connecting and filtering reports with related data.

**Q. What is Linked Report?**

A linked report is a report server item that provides an access point to an existing report. Conceptually, it is similar to a program shortcut that you use to run a program or open a file.  
A linked report is derived from an existing report and retains the original’s report definition. A linked report always inherits report layout and data source properties of the original report. All other properties and settings can be different from those of the original report, including security, parameters, location, subscriptions, and schedules.

You can create a linked report on the report server when you want to create additional versions of an existing report. For example, you could use a single regional sales report to create region-specific reports for all of your sales territories.

Although linked reports are typically based on parameterized reports, a parameterized report is not required. You can create linked reports whenever you want to deploy an existing report with different settings

**Q. What is Snapshot Report?**

A report snapshot is a report that contains layout information and query results that were retrieved at a specific point in time. Unlike on-demand reports, which get up-to-date query results when you select the report, report snapshots are processed on a schedule and then saved to a report server. When you select a report snapshot for viewing, the report server retrieves the stored report from the report server database and shows the data and layout that were current for the report at the time the snapshot was created.

Report snapshots are not saved in a particular rendering format. Instead, report snapshots are rendered in a final viewing format (such as HTML) only when a user or an application requests it. Deferred rendering makes a snapshot portable. The report can be rendered in the correct format for the requesting device or Web browser.

Report snapshots serve three purposes:

1. **Report history:** By creating a series of report snapshots, you can build a history of a report that shows how data changes over time.
2. **Consistency:** Use report snapshots when you want to provide consistent results for multiple users who must work with identical sets of data. With volatile data, an on-demand report can produce different results from one minute to the next. A report snapshot, by contrast, allows you to make valid comparisons against other reports or analytical tools that contain data from the same point in time.
3. **Performance:**By scheduling large reports to run during off-peak hours, you can reduce processing impact on the report server during core business hours.

**Q. What is Cached Report?**

A cached report is a saved copy of a processed report. Cached reports are used to improve performance by reducing the number of processing requests to the report processor and by reducing the time required to retrieve large reports. They have a mandatory expiration period, usually in minutes.

**Q. What are Click through Reports?**

A click through report is a report that displays related data from a report model when you click the interactive data contained within your model-based report. These reports are generated by the report server based on the information contained within the report model. The person who created the model determines which fields are interactive and which fields are returned when a click through report is opened. These field settings cannot be changed in the report authoring tools. Click through reports are auto-generated. However, you can create an alternative customized report to the model for interactive data items that is displayed instead. The custom report is a standard Reporting Services report.

**Q. What are Drilldown Reports?**

Drilldown reports initially hide complexity and enable the user to toggle conditionally hidden report items to control how much detail data they want to see. Drilldown reports must retrieve all possible data that can be shown in the report. For reports with large amounts of data, consider drill through reports instead.

**Q. What are Drill through Reports?**

Drill through reports are standard reports that are accessed through a hyperlink on a text box in the original report. Drill through reports work with a main report and are the target of a Drill through action for a report item such as placeholder text or a chart. The main report displays summary information, for example in a matrix or chart. Actions defined in the matrix or chart provide Drill through links to reports that display greater details based on the aggregate in the main report. Drill through reports can be filtered by parameters, but they do not have to be. Drill through reports differ from sub reports in that the report does not display within the original report, but opens separately. They differ from clickthrough reports in that they are not autogenerated from the data source, but are instead custom reports that are saved on the report server. They differ from drilldown reports in that they retrieve the report data only for the specified parameters or for the dataset query.

**Q. What is Sub report?**

A sub report is a report that displays another report inside the body of a main report. Conceptually, a sub report is similar to a frame in a Web page. It is used to embed a report within a report. Any report can be used as a sub report. The sub report can use different data sources than the main report. The report that the sub report displays is stored on a report server, usually in the same folder as the parent report. You can set up the parent report to pass parameters to the sub report.  
Although a sub report can be repeated within data regions using a parameter to filter data in each instance of the sub report, sub reports are typically used with a main report as a briefing book or as a container for a collection of related reports. For reports with many instances of sub reports, consider using Drill through reports instead.

**Q. What is Data Set in report?**

Data set is a set of data which we want to show in report. Data source is the source of data from where we are getting this data (database server name, database name, connection string).

**Q. Are there issues when exporting SSRS reports into Microsoft Excel? When my users are trying to export a SSRS report into Microsoft Excel, one or two columns in the report appear to merge together. Why might this be?**

Exporting from SSRS is not always perfect, even if you stay within the Microsoft range of products. If you have extra resources, you could splurge for an add-on that offers much better control over exporting to Excel, such as Office Writer. From my experience, though, it is usually headers or footers that cause exporting issues. If any of these headers or footers overlap with data columns in your report, you will find that the exported version of the report has merged cells. Also, check columns next to each other to make sure that there is no overlap, as well.

**Q. What is report subscription?**

Subscriptions are standing requests to deliver report data to requested recipients. Once the report is being subscribed and subscriber will get updates from report server on scheduled interval.

**Q. Can you use a stored procedure to provide data to an SSRS report?**

Yes, you can use a stored procedure. However, your stored procedure should return only a single result set. If it returns multiple result sets, only the first one is used for the report dataset.

**Q. How to send a SSRS report from SSIS?**

Often there is a requirement to be able to send a SSRS report in Excel, PDF or another format to different users from a SSIS package one it has finished performing a data load. In order to do this, first you need to create a subscription to the report. You can create a SSRS report subscription from Report Manager. At the report subscription you can mention the report format and the email address of the recipient. When you create a schedule for the SSRS report, a SQL Server Agent Job will be created. From the SSIS, by using sp\_start\_job and passing the relevant job name you can execute the SSRS report subscription.

**Q. You want to use BIDS to deploy a report to a different server than the one you chose in the Report Wizard. How can you change the server URL?**

You can right-click the project in Solution Explorer and then change the Target-Server URL property.

**Q. Can we deploy SSRS reports on our personal website?**

Your reports can only be deployed on a reporting services site. Your only option for viewing them from other sites is an HTTP link. Some tools, like SharePoint offer controls allowing you to view reports in the context of the other websites, but the report is still deployed to and hosted from reporting services.

## Introduction To Business Intelligence Interview Questions and Answer

So you have finally found your dream job in Business Intelligence but are wondering how to crack the Business Intelligence Interview and what could be the probable Business Intelligence Interview Questions. Every interview is different and the scope of a job is different too. Keeping this in mind we have designed the most common Business Intelligence Interview Questions and Answers to help you get success in your interview.

Below are the top Business Intelligence Interview Questions and Answers so that you can easily crack these Business Intelligence Interview Questions which are asked in an interview

### 1.What is Business Intelligence?

**Answer:**  
The term ‘[Business Intelligence](https://www.educba.com/course/business-intelligence-tableau/)’ (BI) provided the user with data and tool to answer any [decision making](https://www.educba.com/course/decision-making-using-marginal-costing/) an important question of an organization, it can be related to run the business or part of a business. In short, the business intelligence is used for reporting the specified data of any business which is very important and using which higher [management](https://www.educba.com/employee-performance-management-tools/) of any organization will take the decision for the growth of their business. Normally below decisions can be decided by any organization from Business Intelligence tool:  
•BI is used to determine whether a business is running as per plan.  
•BI is used to identify which things are actually going wrong.  
•BI is used to take and monitor corrective actions.  
•BI is used to identify the current trends of their business.

### 2.What are different stages and benefits of Business Intelligence?

**Answer:**  
There are following five stages of Business Intelligence:  
**Data Source**: It is about extracting data from multiple data source.

•**Data Analysis**: It is about providing proper [analysis](https://www.educba.com/sentiment-analysis-to-build-your-brand/) report based on useful knowledge from a collection of data.  
•**Decision-Making Support**: It is about to using information in the proper way. It always targets to provide proper graph on important events like take over, market changes, and poor staff performance.  
•**Situation Awareness**: It is about filtering out irrelevant information and setting the remaining information in the context of the business and its[environment](https://www.educba.com/inclusive-work-environment/).  
•**Risk Management**: It is about to discover that what corrective actions might be taken, or decisions made, at different times.

Following are different benefits of Business Intelligence:  
•Improving decisions making.  
•Speed up on decision making.  
•Optimizing internal [business process](https://www.educba.com/business-process-re-engineering-vs-continuous-improvement/).  
•Increase operational efficiency.  
•Helping or driving for new revenues.  
•Gaining an advantage in terms of competitive markets with another close competitor.

### 3.What are different Business Intelligence tools available in the market?

**Answer:**  
There are a lot of intelligence tools available in the market, in between them below are most popular:

 Popular Course in this category

**Data Scientist Course**43 Online Courses | 170+ Hours | Verifiable Certificate of Completion | Lifetime Access   
4.8 (1,397 ratings)

Course Price   
$249 ~~$599~~   
[**View Course**](https://www.educba.com/data-science/courses/data-scientist-course/?btnz=edu-blg-inline-banner1)

Related Courses

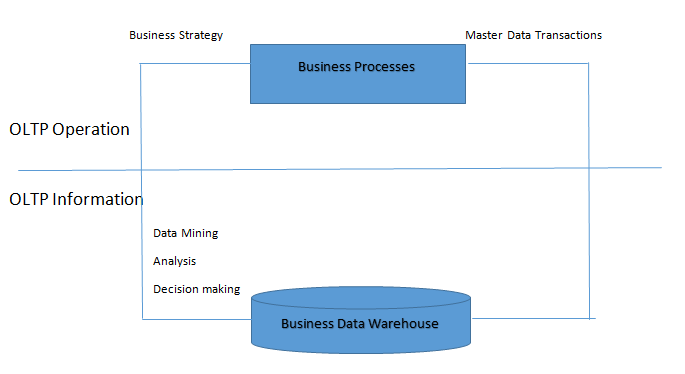
[SAS Training Course](https://www.educba.com/data-science/courses/sas-training-course/?btnz=edu-blg-inline-banner1)[Machine Learning Course](https://www.educba.com/data-science/courses/machine-learning-course/?btnz=edu-blg-inline-banner1)[AWS Training Course](https://www.educba.com/data-science/courses/aws-training-course/?btnz=edu-blg-inline-banner1)

•Oracle Business Intelligence Enterprise Edition (OBIEE)  
•[Cognos](https://www.educba.com/cognos-interview-questions/)  
•Microstrategy  
•SAS Business Intelligence  
•Business Object  
•Tableu  
•Microsoft Business Intelligence Tool  
•[Oracle](https://www.educba.com/course/oracle-application-express-online-training/)Hyperion System

### 4.What is a universe in Business Analytics?

**Answer:**  
The universe is kind of semantic layer in between database and user interface or more correctly it is one of the interfacing layers in between the client (business user) and data warehouse. It actually defines an entire relationship between various tables in a [data warehouse](https://www.educba.com/10-popular-data-warehouse-tools/).

### 5.Define or list the differences between OLAP and OLTP?

**Answer:**  
In general, we can assume OLTP is actually helping to provide source data in a data warehouse and OLAP help to analyze the same.

|  |  |  |
| --- | --- | --- |
|  | **OLTP System** | **OLAP System** |
| **Source of Data** | [Operational data](https://www.educba.com/course/operational-risk-management-w-r-t-us-uk-markets/), OLTP are an original source of data. | [Consolidated data](https://www.educba.com/course/ifrs-10-consolidated-financial-statements/), OLAP data has come from various OLTP databases. |
| **Purpose of Data** | For any kind of current or fundamental business tasks. | To help with future planning, problem-solving or decision making. |
| **Data Updating** | End users initiated data frequently insert or update in the transactional database. | Data updated based on a batch job after one defined time interval. This time can be less or more than one day. |
| **Processing Speed** | As usual typically very fast. | Obviously depends on the amount of data. After refreshing batch data, sometimes complex queries are taken more than hours. Common habit to add an index to improve the speed. |
| **Space requirement** | Again relative small considering historical data in the archived state. | Obviously larger as it has to hold all the historical data, the existence of aggregation[structures](https://www.educba.com/course/data-structures-and-algorithms/), also require more indexes than OLTP. |
| **Database Architecture** | Normalized data, so all the tables and data have a proper relationship. | Typically de-normalize of few tables (like factor dimensions). It normally used a star or snowflake schema. |
| **Backup and Recovery** | Back up is essential requirements on OLTP, as it’s a day to day data, so any loss of data is likely to entail significant monetary loss and legal liability. | Instead of regular backups, some environment may consider simply reloading the OLTP data as a recovery method. |

### 6.What is dashboard in a data warehouse?

**Answer:**  
The dashboard is nothing but the arrangement of all the reports and[graphs](https://www.educba.com/visual-basic-graph-with-dynamic-input/) on one page. It is nothing but the collection of reports in a different format which has same functionality display on the same page.

### 7.Explain the difference between data warehouse and transnational system.

**Answer:**

|  |  |
| --- | --- |
| **Transactional System** | **Data Warehouse System** |
| It normally designed to process with day to day data, so mainly concentrate on high volume transaction processing, rather than backend reporting. | It normally designed to processed high volume analytical reporting and subsequence. It also elaborating report generation. |
| It normally processes driven, means an action of it depends on business-specific task or execution. | It is actually subject-oriented, means it load data from a transactional system, then open to use for any kind of analytical reporting which helps organization for taken proper decision based on that specific subject. |
| It normally handling current transactional data. | It normally handling historical data. |
| Data within a transactional system can insert or update or delete in each task. | Data warehouse data is called as non-volatile, meaning that new data can be added regularly, but once loaded those data are rarely changed. |
| In case of performance or speed, we should always prefer a transactional system for inserting, updating or deleting small volumes of data. | We should always prefer data warehouse to fast retrieval of a relatively large volume of data. |

### 8.Explain Fact and Dimension table with an example.

**Answer:**  
A Fact table is the center table in star schema of a data warehouse. It actually holding quantitative information for analysis, and maximum time it de-normalized.

A dimension table is one of the important tables in star schema of data warehouse, which stores attribute, or dimension, that describe the objects in a fact table.

Fact table mainly holds two types of columns. The foreign key column allows joins with dimension tables, and the major columns contain the data that is being analyzed.

Example: suppose one company sells products to customers. So every sale will be one fact, so fact table holds that information like below:

|  |  |  |  |
| --- | --- | --- | --- |
| **Time ID** | **Product ID** | **Customer ID** | **Unit Sold** |
| 4 | 17 | 2 | 1 |
| 8 | 21 | 3 | 2 |
| 8 | 4 | 1 | 1 |

Now in the fact table, there has customer id, so we need to maintain one dimension table for a customer like below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Customer ID** | **Name** | **Gender** | **Income** | **Education** | **Region** |
| 1 | SS | M | 2 | 3 | 4 |
| 2 | AC | M | 3 | 5 | 1 |
| 3 | MS | F | 1 | 7 | 3 |

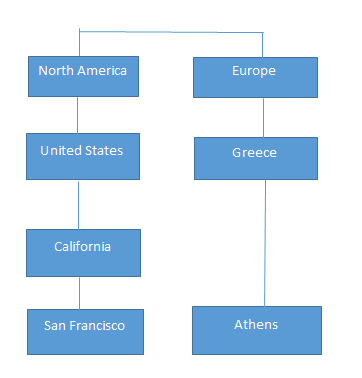
### 9. Define or list the differences between a snowflake schema and star schema.

**Answer:**

|  |  |  |
| --- | --- | --- |
|  | **Snowflake Schemas** | **Star Schema** |
| Maintenance | No redundancy, so easier to maintain. | Holding redundant data, so less easy to maintain. |
| Complexity | More complex query, hence less easy to understand. | Lower complex query, so easy to understand. |
| Query Performance | More foreign key, so longer query execution time. | Less number of foreign key, so query execution is faster compared to snowflake. |
| Utilization | Good to use for data warehouse core to simplify complex relationship (many: many). | Good for Data Mart with simple relationship (1:1 or 1: many). |
| Dimension Table | A snowflake schema may have more than one dimension table for each dimension. | Star schema contains only single dimension table for each dimension. |
| De-normalize | Fact table is in de-normalized form, but dimension table is in normalized form. | Fact and Dimension both the tables are in de-normalized form. |

### 10.Explain or Define a RAGGED hierarchy.

**Answer:**  
Ragged hierarchy actually maintaining a relationship in case of parent member of at least one member of the dimension is not in the level immediately above the member. As an example, if we think about geographical hierarchy and considering North America as continent then it has a country (like the United States), province or state (like California), and city (like San Francisco). But if we consider Europe, Greece, or Athens it doesn’t have this kind of hierarchy. So in this example, Europe, Greece or Athens branches descend to a different depth, creating a ragged hierarchy.



## [What does a DBA do all day?](https://www.techrepublic.com/blog/the-enterprise-cloud/what-does-a-dba-do-all-day/)

## General tasks

* **Installation, configuration, upgrade, and migration**Although system administrators are generally responsible for the hardware and operating system on a given server, installation of the database software is typically up to the DBA. This job role requires knowledge of the hardware prerequisites for an efficient database server, and communicating those requirements to the system administrator. The DBA then installs the database software and selects from various options in the product to configure it for the purpose it is being deployed. As new releases and patches are developed, it's the DBA's job to decide which are appropriate and to install them. If the server is a replacement for an existing one, it's the DBA's job to get the data from the old server to the new one.
* **Backup and recovery**DBAs are responsible for developing, implementing, and periodically testing a backup and recovery plan for the databases they manage. Even in large shops where a separate system administrator performs server backups, the DBA has final responsibility for making sure that the backups are being done as scheduled and that they include all the files needed to make database recovery possible after a failure. When failures do occur, the DBA needs to know how to use the backups to return the database to operational status as quickly as possible, without losing any transactions that were committed. There are several ways the database can fail, and the DBA must have a strategy to recover from each. From a business standpoint, there is a cost to doing backups, and the DBA makes management aware of the cost/risk tradeoffs of various backup methods.
* **Database security**Because databases centralize the storage of data, they are attractive targets for hackers and even curious employees. The DBA must understand the particular security model that the database product uses and how to use it effectively to control access to the data. The three basic security tasks are authentication (setting up user accounts to control logins to the database), authorization (setting permissions on various parts of the database), and auditing (tracking who did what with the database). The auditing task is particularly important currently, as regulatory laws like Sarbanes-Oxley and HIPAA have reporting requirements that must be met.
* **Storage and capacity planning**The primary purpose of a database is to store and retrieve data, so planning how much disk storage will be required and monitoring available disk space are key DBA responsibilities. Watching growth trends are important so that the DBA can advise management on long-term capacity plans.
* **Performance monitoring and tuning**The DBA is responsible for monitoring the database server on a regular basis to identify bottlenecks (parts of the system that are slowing down processing) and remedy them. Tuning a database server is done on multiple levels. The capacity of the server hardware and the way the operating system is configured can become limiting factors, as can the database software configuration. The way the database is physically laid out on the disk drives and the types of indexing chosen also have an effect. The way queries against the database are coded can dramatically change how fast results are returned. A DBA needs to understand which monitoring tools are available at each of these levels and how to use them to tune the system. Proactive tuning is an attitude of designing performance into an application from the start, rather than waiting for problems to occur and fixing them. It requires working closely with developers of applications that run against the database to make sure that best practices are followed so good performance will result.
* **Troubleshooting**When things do go wrong with the database server, the DBA needs to know how to quickly ascertain the problem and to correct it without losing data or making the situation worse.

## Special environments

In addition to these basic responsibilities, some DBAs need special skills because of how the database is being used.

* **High availability**With the advent of the Internet, many databases that could have been available only during the day are now required to be available 24 hours a day, 7 days a week. Web sites have changed from static, pre-defined content to dynamically created content, using a database to create the page layout at the time a page is requested. If the Web site is available 24x7, so must the underlying database. Managing a database in this environment requires an understanding of which types of maintenance operations can be done online (with the database available to users) and which must be scheduled for a maintenance "window" when the database may be shut down. It also requires planning for redundant hardware and/or software components, so that when one fails, others will keep the total system available to its users. Techniques like online backups, clustering, replication, and standby databases are all tools the DBA can use to ensure higher availability.
* **Very Large Databases (VLDBs)**As companies have found more and more uses for database technology, they tend to save more data. Also, the type of data stored in databases has changed, from structured data in neat rows and columns to unstructured data such as documents, images, sound files, and even fingerprints. Both trends have the same result: larger databases. Managing a VLDB requires special skills of the DBA. The time required to do simple operations like copying a table can be prohibitive unless done correctly. The DBA needs to understand techniques like table partitioning (Oracle), federated databases (SQL Server), or replication (MySQL) to enable a database to scale to large sizes while still being manageable.
* **Data Extraction, Transformation, and Loading (ETL)**In data warehouse environments, a key task is efficiently loading the data warehouse or data mart with large volumes of data extracted from multiple existing production systems. Often these production systems have different formats than the standardized definitions in the data warehouse, so data must be transformed (or "cleansed") before loading. Extracting the data may or may not be the DBA's responsibility in a given company, but making sure what is extracted is useful is, and the DBA is a key part of the team.