NAME: K.PRAVEEN KUMAR

ASSIGNMENT-6(9-12-23)

**Database Schema:**  
It is a Structure that represents the logical storage of the database.

To understand the database schema first understand what is meant by database?

Database:

A database is a place to store information and it can store the simplest data as well as complex data.

* Database schema is the logical representation of database, which shows how the data is stored logically in the entire database.
* A database schema contains schema objects that may include tables, fields, packages, views, relationships, primary key, foreign key.

In actual the data is physically stored in the fields that may be in unstructured form, but to retrieve it and needs to put it into structured form. So the database schemas are used.

Types of database Schema:

1.**Physical database schema**: Specifies how the data is stored physically into storage disks and designing a database at the physical level is called a physical schema.

2.**Logical database schema**: It specifies all the logical constraints that need to be applied to the data. Various tools are used to create a logical database schema, and these tools demonstrate the relationships between the component of your data; this process is called **ER modelling**.

ER modelling means Entity-relationship modelling and the specifies the relationship between entities.

3.**View database schema:** The view level design of a database is known as view schema. This schema generally describes the end-user interaction with the database systems.

**Difference between Database Instance and Database schema:**

Both the terms are related to each other but there will be a little confusion between these two and the **Database Schema** is a representation of planned database and does not actually contain the data.

Whereas the Database Instance is a type of snapshot of an actual database as it existed at an instance of time. It varies or can be changed as per the time.

**Creating a Schema:**

1.MySQL: In mysql “Create schema” is similar to “Create database”.

2.Oracle database: Each schema is already present with each database user.

3.SQL Server: “Create Schema” creates a new schema with the name provided by user.

**Database Schema designs:**

* A schema design is the first step in building a foundation in data management.
* Ineffective schema designs are difficult to manage and consume more memory and other resources.

Types:

1**.Flat Model:** A flat model schema is a type of 2-D array in which each column contains the same type of data, and elements within a row are related to each other.

2.**Hierarchical Model:** The Hierarchical model design contains a tree-like structure. The tree structure contains the root node of data and its child nodes. It uses one to one relationship.

3.**Network Model:** It is designed similar to hierarchical model and the difference is it uses many to many relationship.

4.**Relational Model:** The relational models are used for the relational database, which stores data as relations of the table.

5.**Star Schema:** It is the different way to organize the data and It is best suitable for storing and analysing a huge amount of data, and it works on "Facts" and "Dimensions".

6.**Snowflake Schema:** It is derived from the star schema and it also facts and dimensions tables and the dimension tables can have their own dimension tables.

**Stored Procedure:**

**Syntax:**

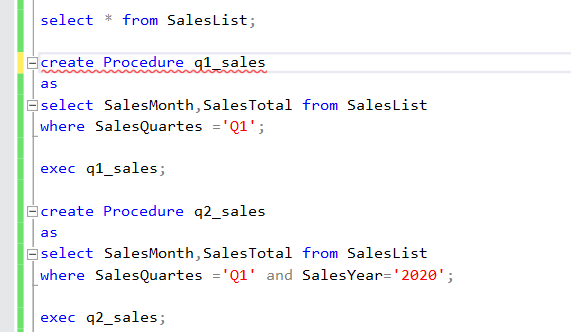
Create Procedure Procedure\_name

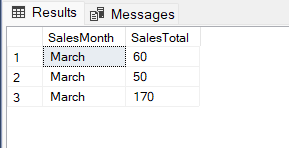
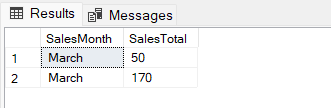
As

Select Column\_name(s) from tablename

Where condition;

**To display** : Exec Procedure\_name;

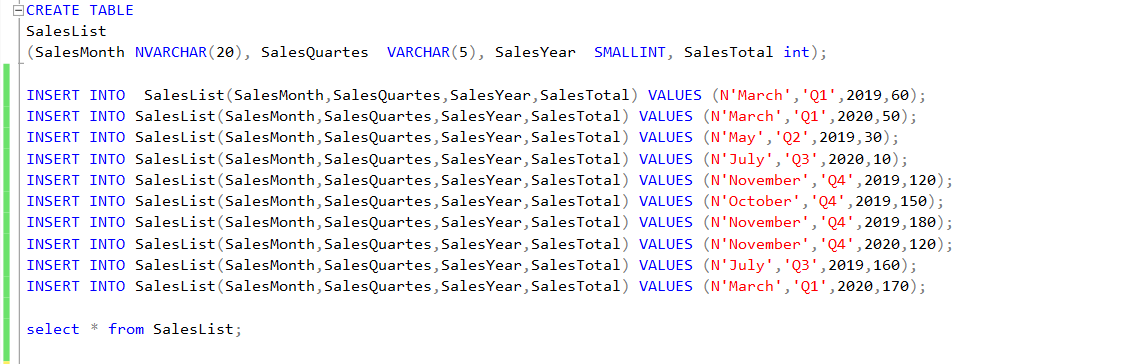


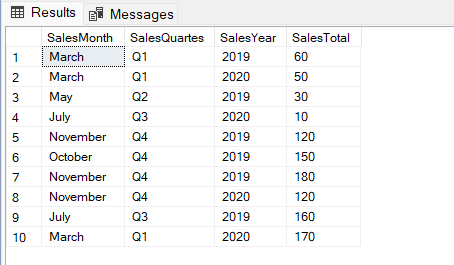
 

**Calculating subtotal in subquery:**

Calculating a subtotal in SQL Query is bit complicated. The SQL offers some GROUPBY extensions that helps us to resolve the issues.

Step 1 : Create table SalesList





Understanding rollup extension:

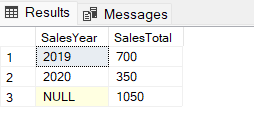
The Groupby statement is used to group the rows which have same values in a new summary row and it is the lead actor of aggregate queries.

ROllUP, cube, Grouping sets are extensions of the groupby statement and add the extra “subtotal” and “Grandtotal” rows to the result.

/\*\*Passing a single column into the rollup\*\*/

SELECT SalesYear, SUM(SalesTotal) AS SalesTotal FROM SalesList

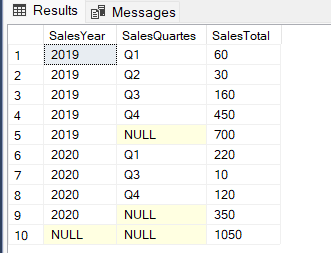
GROUP BY ROLLUP(SalesYear);



/\*\* passing 2 different columns to the rollup so it adds extra 'subtotals' and a 'grand total' row \*\*/

SELECT SalesYear,SalesQuartes, SUM(SalesTotal) AS SalesTotal

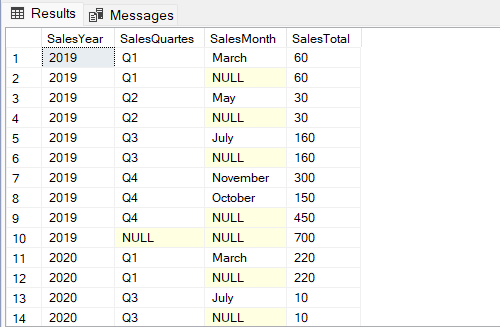
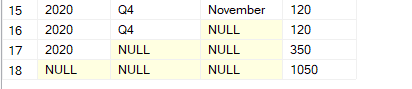
FROM SalesList GROUP BY ROLLUP(SalesYear, SalesQuartes);



/\*\* pass 3 different columns to the roolup so it generates subtotal rows for all heirachies\*\*/

select SalesYear,SalesQuartes,SalesMonth ,SUM(SalesTotal) AS SalesTotal

FROM SalesList GROUP BY ROLLUP(SalesYear, SalesQuartes, SalesMonth);

Grouping Function:

It is used to determine whether the columns in the Groupby list have been aggregated.

Therefore, we use this to identify “NULL” values and replace them.

/\*\* grouping function\*\*/

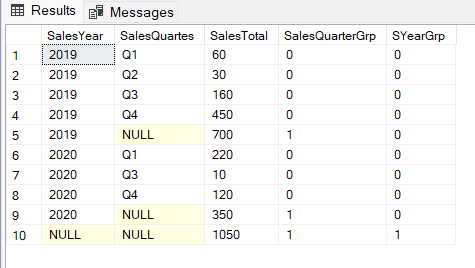
SELECT SalesYear,SalesQuartes,SUM(SalesTotal) AS SalesTotal,

GROUPING(SalesQuartes) AS SalesQuarterGrp,

GROUPING(SalesYear) AS SYearGrp

FROM SalesList

GROUP BY ROLLUP(SalesYear, SalesQuartes);



As we can see, grouped rows are determined by grouping functions. Now use the sql CASE statement and grouping function together so that it will replace NULL values with more meaningful explanations.

/\*\* CASE STATEMENT\*\*/

SELECT CASE

WHEN GROUPING(SalesQuartes)=1 AND GROUPING(SalesYear)=0 THEN 'SubTotal'

WHEN GROUPING(SalesQuartes)=1 AND GROUPING(SalesYear)=1 THEN 'Grand Total'

ELSE

CAST(SalesYear AS varchar(10))

END

AS SalesYear,

SalesQuartes,

SUM(SalesTotal) AS SalesTotal

FROM SalesList

GROUP BY ROLLUP(SalesYear,SalesQuartes);

