

## REGEX

- Powerful tools for pattern matching and string manipulation..
- Provide syntax for searching, matching, and manipulating text.
- '^' = Match beginning of string
- '\$' = Match the end of a string
- [abc] = Matches any character listed between the square brackets
- [^abc] = Matches any character not listed between the square brackets.

```
209      -- REGEXP
```

```
210 •    SELECT name FROM employee WHERE name REGEXP '^ap';
```

```
211
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	name			
▶	stella			
	stella			
	kishore			
	krish			

```
212 •    SELECT name FROM employee WHERE name REGEXP 'ky$';
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	name			
▶	lucky			

```
214 •    SELECT name FROM employee WHERE name REGEXP '[sh]';
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	name			
▶	stella			
	stella			
	kishore			
	krish			

## **STAR SCHEMA**

It's a data model which is used to organize the data in the databases.

- It consists of fact table connected to one or more dimension tables through foreign key relationships.
- After designing the database it looks like a star that's the reason it known as star schema.

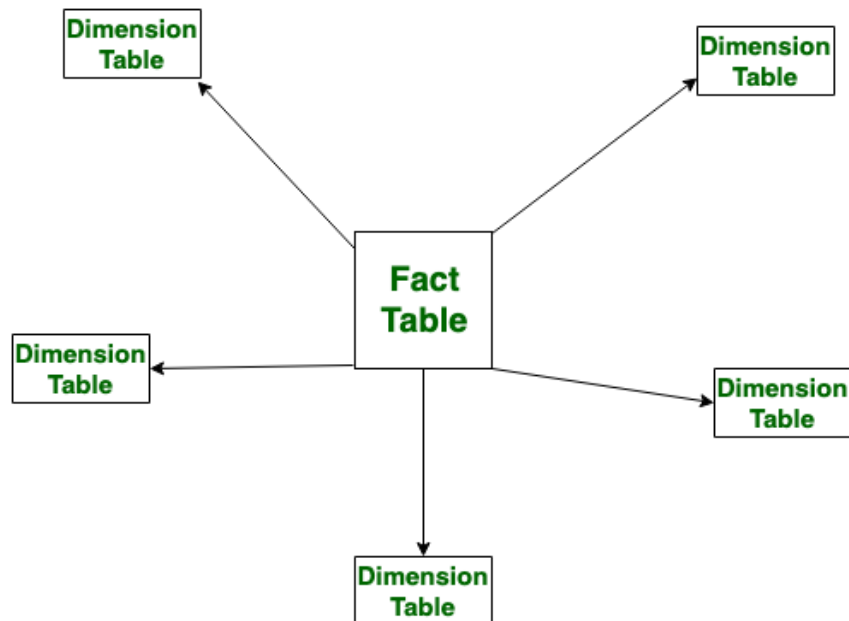
### 1. Fact Table:

- The fact table contains quantitative data.
- The data in fact table can be use for analytical purpose.
- Each row in fact table shows us specific event or transaction.

### 2. Dimension Tables:

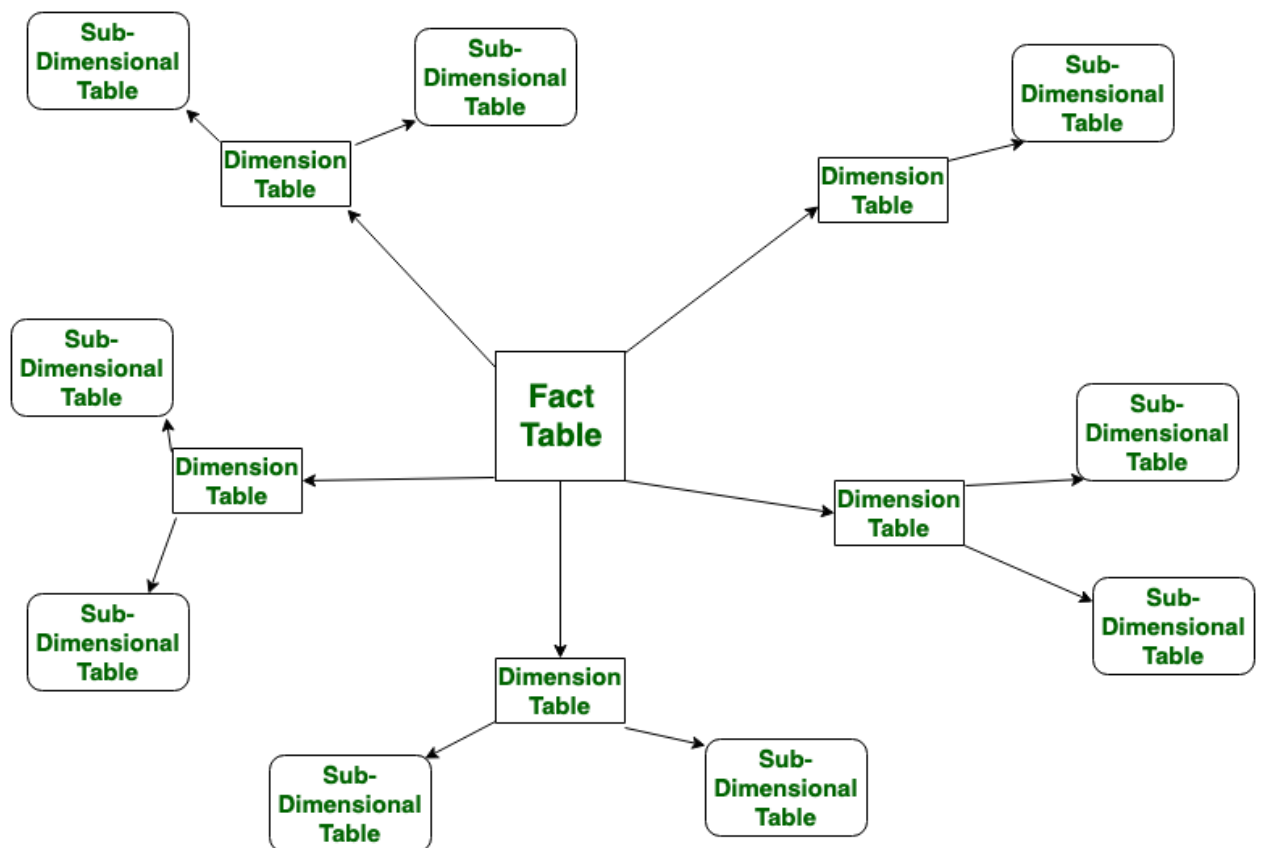
- One or Many dimension table can be present in start Schema
- It contain attributes that provide details about the data in the fact table.

● Fact table and dimension table are connected using foreign key relationships



## **SNOWFLAKING SCHEMA**

- Snowflaking schema is data model that is an extension of a star schema.
- Dimension tables are broken down into subdimensions.
- Dimension tables are broken using concept of normalization
- After designing the database it looks like a Snowflake that's the reason it known as snowflaking schema.



## **MATERIALIZED VIEW**

Also known as a snapshot, It's a database object that contains the results of a precomputed query.

It's different from the normal view as view only stores the query and everytime we invoke the view the query is recomputed every time.

But materialized view actually stores the result set of particular query.

We need to keep the materialized view updated.

```
216      -- MATERIALIZED VIEW
217 •    CREATE VIEW COUNT_ORDER_ID AS
218      SELECT COUNT(OrderID),OrderID FROM OrderDetails GROUP BY OrderID;
219
220 •    SELECT * FROM COUNT_ORDER_ID
```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	COUNT(OrderID)	OrderID			
▶	3	10248			
	2	10249			
	3	10250			
	2	10251			