POSTGRESQL

=============

DATA:

A Data is a Raw fact which defines the attribute of an entity.

DATABASE:

A Database is the place where we can store data in an organized and systematic approach.

APPLICATION:

1. It is an Interface between user and operating system.

2. Every Application has its own dedicated functionallity.

3. Every Application has its own File Extension.

4. It is Platform dependant.

DBMS:

1. DBMS stands for Database management System.

2. The DBMS manages incoming data, organizes it, and provides ways for the data to be modified or extracted by users or other programs.

3. Some DBMS examples include MySQL, PostgreSQL, Microsoft Access, SQL Server, FileMaker, Oracle.

TYPES OF DBMS:

1. Hierarchical DBMS

2. Network DBMS

3. Relational DBMS

4. Object DBMS-(Data Stored in the form of Objects)

RELATIONAL DBMS:

1. A DBMS which organises and store the data by following relational model is known as RDBMS.

2. Similarly, The DBMS which follow the E.F.CODD rule is known as RDBMS.

3. In RDBMS, the data sholuld be stored in the form of tables.

TABLES:

It is the logical representation of data which consists of row and column.

COLUMN:

It is used to represent all the properties of the entities.

ROW:

It is also called as Record/Tuples and it is used to represent all the properties of individual entities.

CELL:

It is the smallest unit of table, which is used to store the data.

CONSTRAINTS:

1. Constraints are used to enforce some rules on columns of the table.

2. It is used to limit the type of data that can go into the table.

3. This ensures the accuracy of data in the table.

NOT NULL:

It ensures that it should have some value or in the other word this constraint is used to specify that the data is mandatory.

UNIQUE:

It is used to avoid duplicacy of values into the column.

CHECK:

It is an extra validation given to a column with a condition.

PRIMARY KEY:

1. It is the combination of unique and notnull.

2. It is used to identify a record uniquely.

3. In a table, we can have only one primary key.

4. It is not mandatory to have a primary key in a table. But, it is highly recommended.

FOREIGN KEY:

1. It is used to establish the relationship between two tables.

2. We can have more than one foreign key in a table and it can be duplicate and null.

3. Foreign key is a key in the child table which is the primary key of its parent table.

CHARACTERISTICS:

1. A table in which foreign key is present is known as child table.

2. A table in which it actually belongs is known as parent table.

3. Foreign Key must and should declare as primary key in the parent table.

4. We cannot insert a value in a column defined as a foreign key, which is not present in the table.

5. So, Foreign key is also called as referential integrity.

DATA TYPES:

1. Numeric

2. Date and Time

3. String Data types

E.F.CODD Rules:

1. The data entered into the cell should be single valued or atomic data.

2. We can store the data in multiple tables, if needed we can establish connection between two tables by using key attributes.

3. Data stored in the tables can be validated in two steps

i] By assigning Datatypes

ii] By assigning Constraints

4. Everything in the database should be stored in table fortmat.

5. A database can only be addressed using a language having linear syntax that supports data definition, data manipulation and data transaction management operations.

6. The data stored in the database must be independent of applications that access the database.

7. All viwes that are theorytically updatable are also updatable by the system.

8. The capability of handling a base or derived relation as a single operand applies not only to the retrieval, but also to insertion, update and deletion of data.

9. All information in the database is represented explicitely at logical level and in exactly one way.

10. Each and every data in database should be accessible by combination of table name, primary key value and column name.

11. Null values are supported in for representing Missing data, inapplicablr information and independent of data.

12. The end user must not able to see that the data is distributed over various locations.

SQL:

1. SQL stands for Structured Query Language.

2. SQL is a language used to interact with the database.

3. According to ANSI (American National Standards Institute), it is the standard language for relational database management systems.

SQL HISTORY:

1. IBM was the first company to develop DBMS which follow Relational model and is also called as System R.

2. IBM developed a language to communicate with system R which is known as SEQUEL.

3. SEQUEL stands for Simple English Query Language.

4. Then, ANSI (American National Standards Institute) took SEQUEL and made some modifications and recommend it as SQL.

5. SQL is the standard language for all the RDBMS Applications.

6. SQL commands are case insesnsitive.

ORDBMS:

An ORDBMS is a relational DBMS that allows developer to integrate the database with their own datatypes and method.

a] ORDBMS -> RDBMS + ODBMS

b] In ORDBMS, we can have our own datatypes.

POSTGRES (or) POSTGRESQL:

a] It is an open source object oriented Database.

b] It is the worlds most advanced open source RDBMS.

SQL COMMANDS:

1] Data Definition Language(DDL)

a] It deals with database or table structure.

b] It helps us to define the structure.

DDL Commands are,

a] Create

b] Alter

c] Drop

d] Rename

e] Truncate

2] Data Manipulation Language(DML)

a] It deals with data in the database.

DML Commands are,

a] Insert

b] Update

c] Delete

3] Data Control Language(DCL)

a] Grant

b] Revoke

4] Transaction Control Language(TCL)

a] Commit

b] rollback

c] savepoint

5] Data Query Language (DQL)

a] select

DDL:

Create:

Syntax --> create database database\_name;

Example --> Create database test;

Alter:

Syntax --> Alter table table\_name add column column\_name datatype;

Example --> Alter table person add column age int;

Drop:

Syntax --> Drop table table\_name;

Example --> Drop table person;

Rename:

Syntax --> Alter table table\_name rename to t\_name;

Example --> Alter table person rename to emp;

DML:

Insert:

Syntax --> Insert into table\_name values(value1,......value\_n);

Example --> Insert into person values(1,'Name');