Data Defenition Languages

1. Data integrity is the overall accuracy, consistency, and structured data, which has a strong collection of proccesses to design the phases on databases. To make the databases so it can built up to a data integrity, we need to maintain to make sure data remains intact and unchanged in its entire life cycle. To maintain this integrity we need to know about key or type of data integrity so we can build the integrity. First is entity integrity which is defines each row to be unique within the table, to do this we need to define a primary key. Second, Referential integrity which is concerned with relationships between tables, to achieve this we need a foreign key which matches the value of its parent's primary key. After that, we can maintain data integrity using domain and user defined integrity such as constraint to validate each inputted data on table.

2. Primary key is a candidate key that contain attributes which are required to uniquely identified tuples. However, foreign key is difference with primary key, foreign key is an attribute on a child table which is referenced from the primary key on its parent's table, foreign key can be null and can accept non-unique values which is different with primary key that can only accept unique value. For composite key, this key is a candidate key that consists of more than one attribute. Example of primary key : students are having unique identification number (ID), foreign key : there is a table called Users that have User_id as primary key, on the other hand there is table called admin that has relation with Users table, in the admin table, it can assigned a User_id which is a foreign key that reference the Users table. Composite key : there is a table that have 2 key : UserID, typeCODE, these two keys can differentiate each data more detailed on the table, because each key has its own function to differentiate the data.

3. BEGIN TRAN is a statement to start transaction that is going to made on the table, once this statement declared, we can do things, like delete, update, or adding. This statement is also important to prevent wrong typing on the query, for example, after we make a query but it has a mistake, we can do statement called ROLLBACK. This statement can undo everything that we have done started from the BEGIN TRAN. EXAMPLE:
   BEGIN TRAN
   DELETE FROM Users
   WHERE UserID = 'USR201'
   (in this situation the delete statement already executed and we want to undo it)
   ROLLBACK

   After we executed ROLLBACK, the delete statement will be cancelled, and the data will be restored as right before the BEGIN TRAN statement executed.
   For COMMIT, once we already sure that the transaction is good enough, we can finalize and save it by typing COMMIT statement. EXAMPLE:
   BEGIN TRAN
   DELETE FROM Users
   WHERE UserID = 'USR201'
   (in this situation the delete statement already executed and we want to finalize it)
   COMMIT

   Once commit executed, the data will be saved and can't be undo