* **DDL (Data Definition Language)** is a language used for creating, removing, and modifying the structure of database objects in a database
* **Constraints** are rules defined optionally at column level or table level
  + Commonly used Constraints:

|  |  |
| --- | --- |
| NOT NULL | Column cannot have NULL Value |
| UNIQUE | All values in column must be unique |
| PRIMARY KEY | Uniquely identifies each row (NOT NULL + UNIQUE) |
| FOREIGN KEY | Links 2 tables and prevents actions that would destroy links between tables.   * Eg, PersonID int FOREIGN KEY REFERENCES Persons(PersonID) |
| CHECK | Column values must satisfy specific condition |
| DEFAULT | Sets default column value if none is specified |

* DDL has a pre-defined syntax for describing data
  + CREATE builds table or database
    - Params include table name and column definitions:
    - CREATE DATABASE *db\_name*
    - CREATE TABLE *table\_name*  
      (*column1\_name datatype [constraints]*,  
       *column2\_name datatype [constraints]*,  
      …);
  + ALTER TABLE adds, drops, rename, and modifies a column in a table
    - ALTER TABLE *table\_name*  
      ADD COLUMN *column\_name datatype;*
    - ALTER TABLE *table\_name*  
      DROP COLUMN *column\_name*
    - ALTER TABLE *table\_name*  
      ALTER COLUMN *column\_name datatype*
      * This alters column datatype
    - ALTER TABLE *table\_name*  
      DROP CONSTRAINT [*constraint\_name*]
  + DROP removes database or table (no longer selectable). Can be recovered using FLASHBACK utility
    - DROP TABLE *table\_name*
  + TRUNCATE deletes *only* data within table
    - TRUNCATE TABLE *table\_name*;
* **Indexes** are special lookup tables that are designed to speed up data retrieval. They are conceptually similar to book indexes
  + CREATE INDEX *index\_name*   
    ON *table\_name* (*col1, col2, …*);
  + DROP INDEX *table\_name.index\_name*;
  + **Clustered indexes** sort and store data rows in table or view based on key values
    - Defines order in which data is stored (1 clustered index per table)
    - Automatically created when a primary key is defined within a table
    - Clustered indexes offer faster data accessing than nonclustered
  + **Nonclustered indexes** store data and indices at separate locations. The index contains pointers to location of that data.
    - Can have multiple non-clustered indexes per table
    - Uses additional disk space, whereas clustered indexes do not
    - Should be created on columns which are used in joins
  + <https://www.guru99.com/clustered-vs-non-clustered-index.html>

**Create Table:**

CREATE TABLE PERSON(

ID int PRIMARY KEY,

FirstName varchar(50),

LastName varchar(50) NOT NULL,

ZipCode char(5),

Salary money CHECK (Salary >= 0)

);

**Alter Table by Adding Constraint to ZipCode Column:**  
ALTER TABLE PERSON

ADD CONSTRAINT CK\_ZIP CHECK (ZipCode LIKE REPLICATE('[0-9]', 5)),

**Populate Table:**  
INSERT INTO PERSON

VALUES(1001, 'Bob', 'Smith', 92805, 80000);

INSERT INTO PERSON

VALUES(1002, 'Mary', 'Smith', 08618, 65000);

INSERT INTO PERSON

VALUES(1003, 'Maria', 'Garcia', 29579, 100000);