Database Creation- practical

- Planning is key
- Know what information will be kept in the database
- Determine the number of entities on which you want to store data
- The number of entities will dictate the number of table to create within a database

Database designing practical Cont…

- Then determine the attributes that will describe the entity
- Attributes will be the fields
- And among the fields of the entity, one should be primary key that uniquely identifies the entity
- These fields will form the columns of the table

Types of fields data

- Data that is stored in the row of each field or column differs
- Others are numbers while others are characters or text
- There are a number of data types; these include:

character, numerical, date, time, date time, enumeration and objects

Types of fields data Cont…

- Character- data in form of strings
- This type is broken into fixed (char) and variable character (varchar) data.
- Used to store most of the data in the tables
- Character data such as name are stored as character set
- Phone numbers are also better stored as character because are not used in numerical operations

Types of fields data Cont…

- Numerical- data which are whole numbers of integers
- Whole numbers :0,1,2,3,4----
- Integers: -4,-3,-2,-1,0,1,2,3,4---

- It includes also decimal numbers (points)

Decimals: 23.6,3.6,98.3--

Types of fields data Cont…

- Enumeration –used to keep predetermined YES and NO type of data
- Objects- used to keep large amount of data
- Text and blogs are examples of this data category

Symbols used to represent data types when database design

- CHAR
- VARCHAR
- TEXT
- BLOG
- INT
- DECIMAL
- DATE
- DATETIME
- TIME

MySQL Table Storage Engines Types

- Several storage engines exist
- These include MyISAM and InnoDB
- When creating a table, you need to declare the type of storage engine your table will be using

MyISAM

- MYISAM supports Table-level Locking
- It is designed for need of speed
- Does not support foreign keys hence we call MySQL with MYISAM is DBMS
- MyISAM stores its tables, data and indexes in diskspace using separate three different files. (tablename.FRM, tablename.MYD, tablename.MYI)
- It does not supports transaction. You cannot commit and rollback with MYISAM. Once you issue a command it’s done.

InnoDB

- InnoDB supports Row-level Locking
- It is designed for maximum performance when processing high volume of data
- It also supports foreign keys hence we call MySQL with InnoDB is RDBMS
- InnoDB stores its tables and indexes in a tablespace
- It supports transaction. You can commit and rollback with InnoDB

Database Constraints

- Constraints are used to specify rules for the data in a table
- They limit the type of data that can go into a table.
- This ensures the accuracy and reliability of the data in the table

Database Constraints

- Constraints can be column level or table level.
- Column level constraints apply to a column
- And table level constraints apply to the whole table.

Examples of Database Constraints

a) **NOT NULL** - Ensures that a column cannot have a NULL value

b) **UNIQUE** - Ensures that all values in a column are different

c) **PRIMARY KEY** - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table

d) **FOREIGN KEY** - Uniquely identifies a row/record in another table

e) **CHECK** - Ensures that all values in a column satisfies a specific condition

f) **DEFAULT** - Sets a default value for a column when no value is specified

g) **INDEX** - Used to create and retrieve data from the database very quickly

EXAMPLES on how to use the constraints

```
CREATE TABLE Orders (
    OrderID int NOT NULL,
    OrderNumber int NOT NULL,
    PersonID int,
    PRIMARY KEY (OrderID),
    FOREIGN KEY (PersonID) REFERENCES
Persons(PersonID)
) type=InnoDB;
```

EXAMPLES on how to use the constraints

```
CREATE TABLE Persons (
    ID int NOT NULL,
    Country  char  DEFAUT  ‘ Zambia’,
    LastName varchar(255) NOT NULL,
    FirstName varchar(255),
    Age int,
    CHECK (Age>=18)
) type=InnoDB;
```