Day09

* We started with fetching data operation in python
  + First learnt how to fetch one data using: .fetchone()
  + Then how to fetch multiple data using - .fetchall()
* Then we learnt about how to create table using function but here parameters are predefined so not that future proof so to make it further updatable added the features
  + Take the column name in input as list
  + Then join it depending upon the “,” in the list
  + And boom your code works and now dynamically you can add as many columns as you want with any datatype required
* Then we learnt how to rename table name using function in python
* Then we learnt how to add one more column in our table using func in python
* Then we learnt how to rename the column name
* Then we learnt how to modify the column’s datatype which it will take as input : matlab int ka varchar etc
* Then we learnt how to delete table(drop table) using functions
* Also learnt about how to truncate the table that is delete values in table by keeping structure as it is
* Then learnt how to insert data in table using function but single data at a time(in afternoon session we decided to modify it)
* Then we learnt how to delete data/row using id and name both functions we executed
* How to update the existing value in table we can update single value also in a row and keeping other element as it is
* Now in Afternoon session,we started with filtering data on some condition like start and end for name column in table
* And then we tried executing the insert data for multiple data in our function

The question you will think that why’re we solving the problem that do not exist like why require to create func directly create the table and maybe you are right but it reduces the future work; it is not like that you will only create table once in your life but this code will be written once and we can use it everythime we want or require so future simplicity is more important than current complexity

**NOW sir took amazing problem statement/case study**

**Problem Statement**

Create a **student marks evaluation system** that retrieves a student's subject-wise marks from a MySQL database based on their seat number, calculates their **average score**, and categorizes their performance into predefined levels (**Extraordinary, Excellent, Average, Pass, or Fail**).

Approach:

**Approach**

1. **Establish Database Connection**
   * Use pymysql to connect to the MySQL database (pywit) and initialize a cursor for executing queries.
2. **Take User Input**
   * Accept name (for display) and seat\_no (for fetching records) from the user.
3. **Fetch Data from Database**
   * Query the marks table to retrieve subject-wise marks for the given seat\_no.
4. **Process and Calculate Average**
   * Extract individual subject marks, display them, and compute the **average score**.
5. **Classify Performance**
   * Compare the average score with predefined thresholds to determine and print the student's performance category.