The objective of this lab is to

- DB handling (insert, update, delete from database)
- Classes

Instructions!

- Keep your student identity cards with you.
- This is an individual lab, you are strictly **NOT** allowed to discuss your solutions with your fellow colleagues, even not allowed to ask how is he/she is doing, it may result in a negative marking.
- You can **ONLY** discuss this with TAs or Ma'am.
- Save your work frequently. Make a habit of pressing CTRL+S after every line of code you write.
- This is a **GRADED** lab, so, at the end of the lab session, you should have your complete work ready for evaluation.
- Follow proper coding conventions and write comments.
- ***Total Time for this Lab is 90 minutes.***

Task 01 [30 Marks - 90 minutes]

ATM System in python

In this task, you'll design similar system as the previous lab with some extended functionality using database. Instead of utilizing file, you will use database to store your user's information. You have to implement classes where needed (Account Class, User Class etc). Create tables in Database like Users, and an Accounts table that contain the number of users registered for an account.

Below mentioned are the details of each major component.

- **Exceptions (same as previous)**
  - Create a custom exception class ***ATMException,*** which handles:
    - ***AccountNotFound*** (while login, if user enters wrong account number).
    - ***InsufficientBalance*** (While withdrawing money more than balance).
    - ***InvalidPin*** *(While setting up pin, it should be exactly 4-digit numerical value, other than that, raise this exception).*

    For all other exceptions like negative numbers, invalid format of pin, database error etc. you can use python's built-in exceptions.

Make an account with a default Account Number and a Password for Charity Management and its account type will be Charity which has no limitations for withdrawal and deposit.

- **Register an account**
  - A User can register his/her account, with *account number and password.*
    - **Account Number** will be automated (assigned by your program), think of like a sequence with some prefix like ATM001, ATM002 etc.
    - **Password** should be exactly 4 digits numerical number.
    - **Account balance** keep the default opening balance 100 (no need to take it from user).
    - **Type of Account** should be a string according to the information given below.
    - **Interest Rate** depend on the type of account given below
    - **Number of Transactions** initialize it with zero and then keep track of it (no need to take it from user)

      If you want to add any other properties like username etc. You can do it.

  - After successful registration, the data should be **saved in a database,**
    table format should be:
    account number, password, balance, type of account, interest rate, no. of transactions

- **Login**
  - Once a user is registered, one can login with his/her account number and password. After login, user can perform following functionalities:
    - **Check his/her balance.**
      - Just print the balance in one's account after applying respective interest rate
    - **Withdraw**
      - Take the amount from the user.
      - Handle exception if the user enters negative amount or amount more than his/her balance.
    - **Deposit**
      - Take the amount user wants to deposit.
      - Handle exception if the user enters negative amount.
    - **Transfer Amount**
      - Take the amount from the user
      - Take the account number where amount will be transferred
      - Check for the type of account.
      - Add the amount to the receiving account.
      - Handle exception if the user enters negative amount.

- **Types of Account**
  - On the base of purpose they serve, accounts can be classified into following types
    - **Basic Bank Account**
      - Maximum four transactions i.e., two deposits and two withdrawals with no fee on monthly basis. Account holders will have to pay a small fee for additional transactions (100 for each additional transactions)
    - **Current Account**
      - No restrictions on the number of deposits and withdrawals
    - **Savings Account**
      - No restrictions on the number of deposits and withdrawals
      - Earn interest at a rate of 4%
    - **Fixed Deposit Account**
      - Cannot withdraw money for a specific period
      - Earn interest at a rate of 8.26%

NOTE: Your Program shouldn't stuck while taking user input no matter what a user enters, file not found error, user already exists etc., all these exceptions and errors must be handled carefully.