**Ineuron Assessment (Screening Task)**

Problem: Make a console with the help of Python and SQL

Functions of the console should be:

1. User can check their bank balance
2. User can withdraw some amount form their account

* Bank amount should be 5000 not less then that.

1. User can check their transaction statemtent.

Have a Three table in a database called as

User Table (user\_id, user\_name, user\_dob, user\_email, user\_created\_date)

BankAccount Table (user\_id, bank\_acc\_id, is\_useractive, amount)

Transaction Table Table (transaction\_date, user\_id, bank\_acc\_id, withdrawn\_amount)

**Solution**:

Step1: Python Program

**Directory name**: Database

**Script name**: CreateDB.py

**Class name**: class DB

**Function:** NA

**Arguments take**: ‘dbname’ as database name and ‘password’ as password of database

**Methods**: def createDatabase()

**Method returns**: name of database ‘self.dbname’

**Functionality**: This class and its method is responsible to create a database and make connection with the respective database.

Step2: Python Program

**Directory name**: Tables

**Script name**: CreateTB.py

**Class name**: class TB

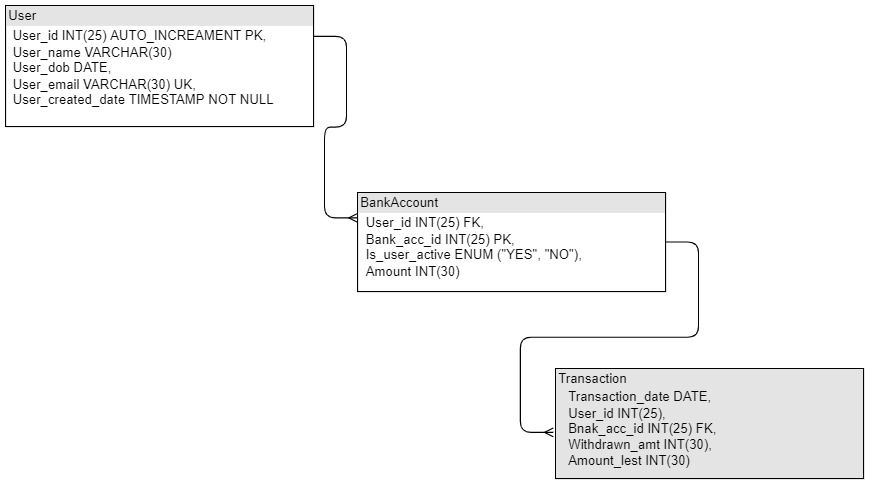
**Function:** NA

**Arguments take**: ‘table1’ as name of first table, ‘table2’ as name of second table and ‘table3’ as name of third table.

**Methods**: def creatingtable1(), def creatingtable2() and creatingtable3()

**Method returns**: name of tables ‘self.table1’, ‘self.table2’ and ‘self.table3’

**Functionality**: This class and its method is responsible to create a table and make connection with the respective database.



Step3: MySQL Stored Procedure

**Store Procedure**: checkamount

**argument**: IN id INT

**Structure**:

*CREATE DEFINER=`root`@`localhost` PROCEDURE `checkamount`(in id int)*

*BEGIN*

*select bankaccount.amount, user.user\_name*

*from user*

*inner join bankaccount on user.user\_id = bankaccount.user\_id*

*where bankaccount.Bank\_acc\_id = id;*

*END*

**Functionality**: This SQL stored procedure is responsible for invoke the User bank amount.

Step4: MySQL Stored Procedure

**Store Procedure**: withdrawn

**argument**: IN amt INT, IN id INT

**Structure**:

*CREATE DEFINER=`root`@`localhost` PROCEDURE `withdrawn`(in amt int, in id int)*

*BEGIN*

*update bankaccount*

*set amount = if(amount>5000, amount-amt, "Amount Limit Error")*

*where Bank\_acc\_id = id;*

*select user\_id, bank\_acc\_id, amount*

*from bankaccount*

*where bankaccount.Bank\_acc\_id = id;*

*END*

**Functionality**: This SQL stored procedure is responsible for invoke the withdrawn amount.

Step5: MySQL Stored Procedure

**Store Procedure**: transactioncheck

**argument**: IN dt DATE, IN id INT

**Structure**:

*CREATE DEFINER=`root`@`localhost` PROCEDURE `transactioncheck`(in dt date, in id int)*

*BEGIN*

*select transaction.withdrawn\_amt, transaction.amount\_left, user.user\_name*

*from user*

*inner join transaction on user.user\_id = transaction.user\_id*

*where transaction.transaction\_date = dt and transaction.user\_id = id;*

*END*

**Functionality**: This SQL stored procedure is responsible for invoke the transaction statement.

Step6: Python Program

**Directory name**: Insert

**Script name**: InsertDT.py

**Class name**: class ID

**Function:** NA

**Arguments take**: It won’t be any arguments.

**Methods**: def insertintoUser() and def insetintoBank()

**Method returns**: It won’t return anything

**Functionality**: This class and its method is responsible to insert records into respective tables.

Step7: Python Program

**Directory name**: Ineuron\_Assesment\_job

**Script name**: main.py

**Class name**: NA

**Function:** def dataintoUser(), def dataintoBank(), CheckAmount(), withdrawn(), transactioncheck()

**Arguments take**: def dataintoUser(), def dataintoBank() this function won’t take any arugments.

CheckAmount(), withdrawn(), transactioncheck() These functions will take arguments as per the Stored Procedure.

**Methods**: NA

**Method returns**: NA

**Functionality**: This is a main script which invokes the SQL Stored Procedures