

CSE 4701, Fall 2019

Project 2 – Bank Accounts and Transaction Processing

Introduction

In this project you will be creating a simplified banking software to practice concepts of **embedded SQL** and **database transactions**. Information on bank accounts must be persistently stored in a MySQL database in your local computer. Your bank account database will have only one table named **account** with the following details.

```
mysql> desc account;
```

Field	Type	Null	Key	Default	Extra
account_no	int(11)	NO	PRI	NULL	auto_increment
name_on_account	int(11)	NO		NULL	
balance	float	NO		NULL	
account_open_date	datetime	NO		CURRENT_TIMESTAMP	

```
4 rows in set (0.00 sec)
```

Part I (Due November 13/2019 (Wed), midnight at HuskyCT) (100 pts)

1. Part I Assignment

Create a local MySQL database named “cse4701f19_project2” and create the **account** table in this database. Use the following SQL script:

```
--
```

```
CREATE DATABASE IF NOT EXISTS cse4701f19_project2;
```

```
USE cse4701f19_project2;
```

```
CREATE TABLE account (
```

```
    account_no INT(11) NOT NULL AUTO_INCREMENT,
```

```
    name_on_account VARCHAR(100) NOT NULL,
```

```
    balance FLOAT NOT NULL DEFAULT '0',
```

```
    account_open_date DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
```

```
    PRIMARY KEY (account_no)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--
```

Now write a program in any programming language of your choice (Python, Java, PHP, etc.) that provides user interface with a menu to do the following operations on this database:

- Create Account** – When a new customer comes to the bank, the bank teller should be able to open a new account using your program. Bank teller will enter **name_on_account** and opening **balance** in order to create a new account. Other attributes **account_no** and **account_open_date** should get auto generated values.

- ii. **Check Balance** – Bank teller should be able to make balance inquiries on existing bank accounts. They will enter **account_no** and your program should show the details (number, name, balance, and account open date). Display error if invalid account number is supplied.

How to create “User Interface Menu”? You can decide on what kind of User Interface you would like to build for this project. Depending on what language you choose, you can either make a Graphical User Interface (GUI) or a Command Line Interface (CLI). Regardless, the various options as outlined in the project description should be present in the menu and user should be able to choose from those options. An example CLI with **menu** using python is given here.

```
Main Menu
1 - Create Account
2 - Check Balance
3 - Deposit
4 - Withdraw
5 - Transfer
0 - Quit
Enter your choice: 1
Name on account: John Doe
Enter Initial Balance: 125
---Account successfully created---
Account number: 10
Name on account: John Doe
Balance: 125.0
Account opened on: 2019-04-17 11:23:35

5 - Transfer
0 - Quit
Enter your choice: 2
Enter account number: 10
---Checking account balance---
Account number: 10
Name on account: John Doe
Balance: 125.0
Account opened on: 2019-04-17 11:23:35
```

2. Deliverables

You are required to submit your source code (zipped if multiple files) and a report (PDF file) with following contents (use same number system in your report):

- i. **Screenshots:** Attach screenshots from your program and corresponding code snippet to demonstrate each subsection of assignment in Section 1 above is working (i.e., **Create Account** and **Check Balance**).
- ii. **Durability:** Once you create few accounts in step (i), close your program and run it again. Check balance for the previously created accounts and show that bank accounts are persistently stored in the database. Make sure you show timestamp of transaction. An example printing time stamp is given.