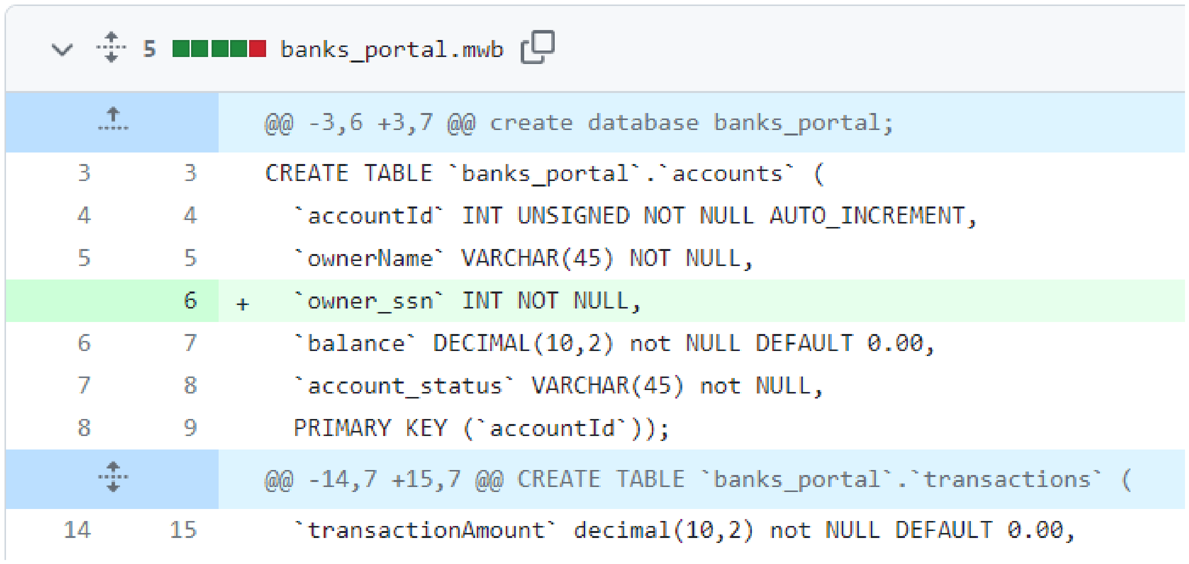
Anthony Guidice

CIS 344

Professor Peralta-Ramos

Final Project

I was tasked with creating a portal for bank tellers to use for customer's accounts. The database and code are meant to store customer's information and be used to withdraw and deposit money into their accounts. The first thing I had to do was create the database and tables for their accounts. Please see below to view how I created the database and tables. These screenshots are being pulled from the code I uploaded to github.

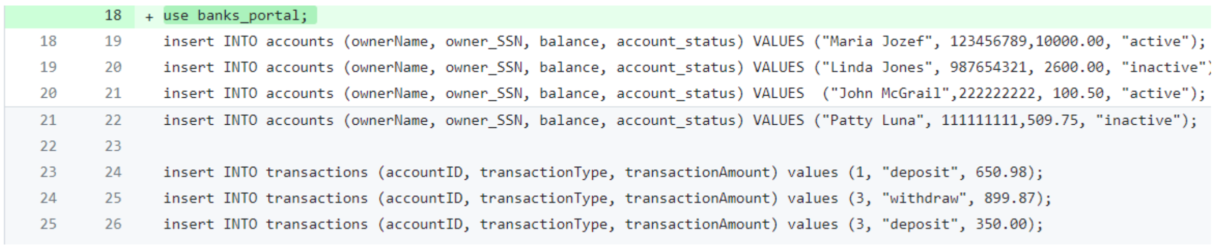


5 banks_portal.mwb

```
@@ -3,6 +3,7 @@ create database banks_portal;
 CREATE TABLE `banks_portal`.`accounts` (
   `accountId` INT UNSIGNED NOT NULL AUTO_INCREMENT,
   `ownerName` VARCHAR(45) NOT NULL,
+  `owner_ssn` INT NOT NULL,
   `balance` DECIMAL(10,2) not NULL DEFAULT 0.00,
   `account_status` VARCHAR(45) not NULL,
   PRIMARY KEY (`accountId`));
@@ -14,7 +15,7 @@ CREATE TABLE `banks_portal`.`transactions` (
   `transactionAmount` decimal(10,2) not NULL DEFAULT 0.00,
```

The first line shows that I created the database, the next few lines are the table within the database and the corresponding columns inside of the table. I added in accountId, which will display the customer's account number, the customer's name the account is under, their social security number to identify them, their available balance, and their account's status. This is all in the 'Accounts' table. The next table I created is the transactions table with the columns transaction id, account id, transaction type, and transaction amount.

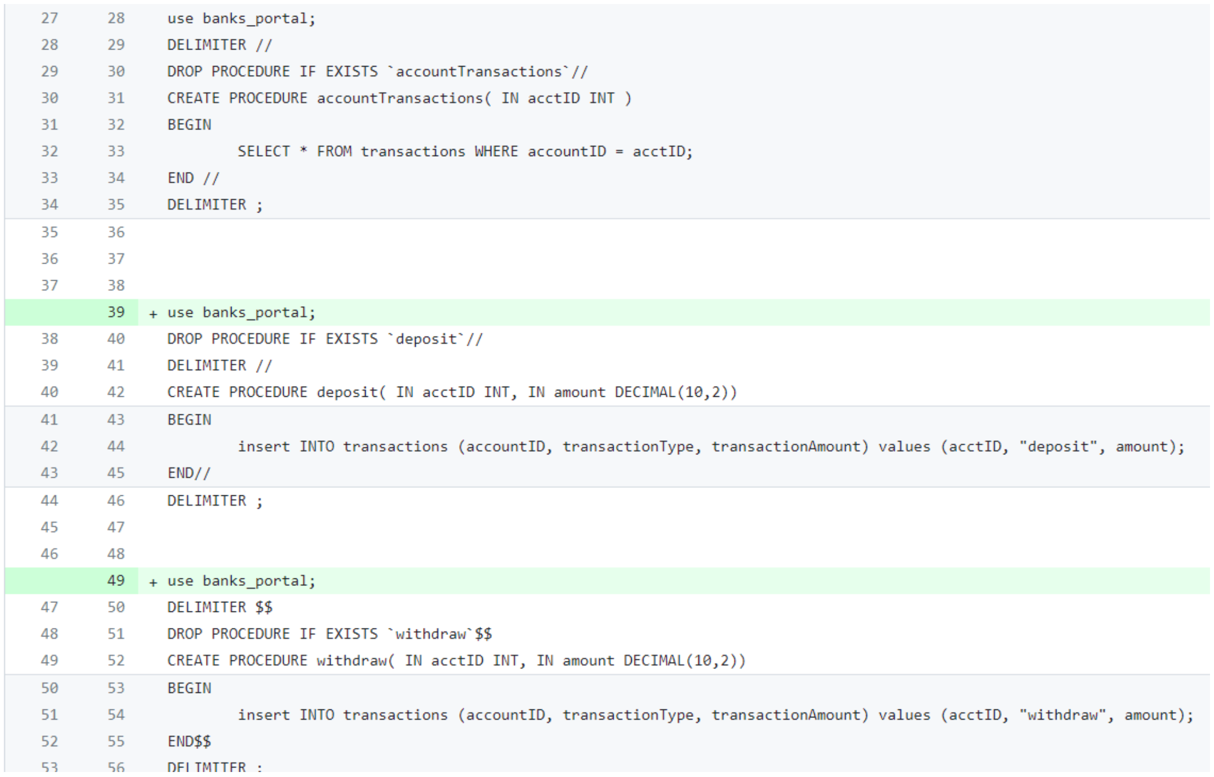
Once the tables were created I had to add values into them, which is what I did with the next few lines of code. Please see the screenshot below to see the values I added.



```
+ use banks_portal;
  insert INTO accounts (ownerName, owner_SSN, balance, account_status) VALUES ("Maria Jozef", 123456789,10000.00, "active");
  insert INTO accounts (ownerName, owner_SSN, balance, account_status) VALUES ("Linda Jones", 987654321, 2600.00, "inactive"
  insert INTO accounts (ownerName, owner_SSN, balance, account_status) VALUES  ("John McGrail",222222222, 100.50, "active");
  insert INTO accounts (ownerName, owner_SSN, balance, account_status) VALUES ("Patty Luna", 111111111,509.75, "inactive");

  insert INTO transactions (accountID, transactionType, transactionAmount) values (1, "deposit", 650.98);
  insert INTO transactions (accountID, transactionType, transactionAmount) values (3, "withdraw", 899.87);
  insert INTO transactions (accountID, transactionType, transactionAmount) values (3, "deposit", 350.00);
```

I added the values that would correspond to real customers and real accounts, so that they would have a function at a bank in the real world. Then I had to add in procedures, so that the tellers can perform withdrawals and deposits into these customers accounts. Please see below for the code I used to add in these procedures.



```
  use banks_portal;
  DELIMITER //
  DROP PROCEDURE IF EXISTS `accountTransactions`//
  CREATE PROCEDURE accountTransactions( IN acctID INT )
  BEGIN
          SELECT * FROM transactions WHERE accountID = acctID;
  END //
  DELIMITER ;



+ use banks_portal;
  DROP PROCEDURE IF EXISTS `deposit`//
  DELIMITER //
  CREATE PROCEDURE deposit( IN acctID INT, IN amount DECIMAL(10,2))
  BEGIN
          insert INTO transactions (accountID, transactionType, transactionAmount) values (acctID, "deposit", amount);
  END//
  DELIMITER ;


+ use banks_portal;
  DELIMITER $$
  DROP PROCEDURE IF EXISTS `withdraw`$$
  CREATE PROCEDURE withdraw( IN acctID INT, IN amount DECIMAL(10,2))
  BEGIN
          insert INTO transactions (accountID, transactionType, transactionAmount) values (acctID, "withdraw", amount);
  END$$
  DELIMITER ;
```

I added in ways for the tellers to see what other transactions the customer has made in case there was a discrepancy. I used delimiters to achieve these goals and also to do simple deposits and withdrawals for the customers. There is still more work to do, so that the tellers can do more complex transactions, but this is a great start to their database.