

Lab Assignment 11 6422771251 Matas Thanamee

Assignment 1:

Stored Procedure to Deposit Money:

```
DELIMITER // CREATE PROCEDURE DepositMoney(accountID INT, de-
positAmount FLOAT) BEGIN DECLARE currentBalance FLOAT;

-- Get the current balance of the account SELECT bal INTO currentBalance
FROM account WHERE ID = accountID;

-- Update the balance by adding the deposit amount SET currentBalance =
currentBalance + depositAmount;

-- Update the 'bal' column with the new balance UPDATE account SET bal =
currentBalance WHERE ID = accountID; END // DELIMITER ;
```

Stored Procedure to Withdraw Money:

```
DELIMITER // CREATE PROCEDURE WithdrawMoney(accountID INT,
withdrawAmount FLOAT) BEGIN DECLARE currentBalance FLOAT;

-- Get the current balance of the account SELECT bal INTO currentBalance
FROM account WHERE ID = accountID;

-- Check if there is enough balance to withdraw IF currentBalance >= with-
drawAmount THEN -- Update the balance by subtracting the withdraw amount
SET currentBalance = currentBalance - withdrawAmount;

-- Update the 'bal' column with the new balance UPDATE account SET bal =
currentBalance WHERE ID = accountID; ELSE SIGNAL SQLSTATE '45000'
SET MESSAGE_TEXT = 'Not enough money'; END IF; END // DELIMITER
;
```

Assignment 2:

Trigger to Prevent Transaction Edit:

```
DELIMITER // CREATE TRIGGER PreventTransactionEdit BEFORE
UPDATE ON 'transactinos' FOR EACH ROW BEGIN SIGNAL SQLSTATE
'45000' SET MESSAGE_TEXT = 'Transaction edit is not allowed'; END //
DELIMITER ;
```

Trigger to Check Balance before Insert:

```
DELIMITER // CREATE TRIGGER CheckBalanceBeforeInsert BEFORE IN-
SERT ON 'transactinos' FOR EACH ROW BEGIN DECLARE currentBalance
FLOAT;

SELECT bal INTO currentBalance FROM account WHERE ID =
NEW.account_id;
```

```
IF currentBalance < NEW.amount THEN SIGNAL SQLSTATE '45000' SET  
MESSAGE_TEXT = 'Not enough money'; END IF; END // DELIMITER ;
```