Assignment: Exercise Account, PersonalAcct, BusinessAcct
Credit:CSE 3130 Object-Oriented Programming 2

How has your program changed from planning to coding to now? Please explain?

Since each type of account charges a different amount of money upon creation, initially I had planned to subtract the charge in the constructor when the account was made.

```
    public BusinessAcct(double bal, String fName, String lName, String st, String c, String p, String pC)
    {
        super(bal, fName, lName, st, c, p, pC);

        if(super.getBalance() < 500)
        {
            super.withdrawal(10.00);
            System.out.println("$10.00 charged.");
        }
    }

    public String toString()
    {
        String busAcctString;

        busAcctString = "Business Account" + '\n' + super.toString();

        return(busAcctString);
    }

}
```

History Synchronize Git Staging Git Reflog Properties Console

```
ank [Java Application] C:\Program Files\Eclipse\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.3.v20240426-1530\jre\bin\javaw.ex

astname:

treet:

ity:

rovince:

ostal Code:

10.00 charged.
2.00 charged.
pen a personal account: 1
pen a business account: 2

usiness Account
m
```

This would've worked if I had stuck to my original plan to make the account then run the while loop for banking actions underneath the corresponding conditional statement. That means I would've had two of the exact same while loops in my code just with different objects.

```
//While true loop
while(true)
{

    //Switch case statements to change or keep the account object
    switch(accType)
    {
        case 1: account = pAcct; break;
        case 2: account = bAcct; break;
    }

    //User menu
    System.out.println("Current balance is " + money.format(account.getBalance()));
    System.out.println("(D)eposit" + '\n' +
                       "(W)ithdrawal" + '\n' +
                       "(A)ddress change" + '\n' +
                       "(V)iew account details" + '\n' +
                       "(Q)uit");
    //Record user input
    action = input.next();

    //Deposit
    if(action.equalsIgnoreCase("d"))
    {
        System.out.println("Enter deposit amount: ");
        amt = input.nextDouble();
        account.deposit(amt);
    }
    //Withdrawal
    else if(action.equalsIgnoreCase("w"))
    {
        System.out.println("Enter withdrawal amount: ");
        amt = input.nextDouble();
        account.withdrawal(amt);
    }
    //View account details
    else if(action.equalsIgnoreCase("v"))
    {
        System.out.println(account);
    }
    //Change address, if input = 'same' then original values will be kept
    else if(action.equalsIgnoreCase("a"))
    {
        System.out.println("Enter 'same' to keep original address values.");

        System.out.println("Street: ");
        street = input.next();
        if(street.equalsIgnoreCase("same"))
        {
            street = account.getStreet();
        }

        System.out.println("City: ");
```

The way I set up charging money for account creation would've worked here but since the code was lengthy and inefficient I decided to try and find a way to just have a while loop. I borrowed strategies from the store class demo with manager and employee classes.

```
// Prompt user to choose either a personal or business acc
System.out.println("Open a personal account: 1" + '\n' +
                   "Open a business account: 2");

accType = input.nextInt();

//Initialize account object
//Switch case statements in the while loop will change thi
Account account = pAcct;

//While true loop
while(true)
{

    //Switch case statements to change or keep the account
    switch(accType)
    {
        case 1: account = pAcct; break;
        case 2: account = bAcct; break;
    }
```

But because earlier I had set the constructor to charge money each time the object was made this meant that everytime the loop ran then if applicable the account would be charged. To fix this I made charging money into another method, I didn't use the withdrawal method since I wanted to add a message telling the user what they'll be charged or if they won't be charged for each account scenario.

Display.

```
$2.00 will be charged if a personal acount is made.
$10.00 will be charged if a business acount is made.
Open a personal account: 1
Open a business account: 2
1
Current balance is $8.00
```

Method.

```
//Check if money has to be charged to account if balance is too low
public void balanceMinimum()
{
    if(super.getBalance() < 100)
    {
        super.withdrawal(2.00);
        System.out.println("$2.00 will be charged if a personal acount is made.");
    }
    else
    {
        System.out.println("A personal account can be made charge free.");
    }

}
```

Main.

```
    //Create objects for a personal and business account
    PersonalAcct pAcct = new PersonalAcct(bal, fN, lN, street, city, prov, postalCode);
    BusinessAcct bAcct = new BusinessAcct(bal, fN, lN, street, city, prov, postalCode);

    //This is where money will be charged if balance is too low for each account
    pAcct.balanceMinimum();
    bAcct.balanceMinimum();

    // Prompt user to choose either a personal or business account
    System.out.println("Open a personal account: 1" + '\n' +
                       "Open a business account: 2");

    accType = input.nextInt();

    //Initialize account object
    //Switch case statements in the while loop will change this value
    Account account = pAcct;

    //While true loop
    while(true)
    {

        //Switch case statements to change or keep the account object
        switch(accType)
        {
            case 1: account = pAcct; break;
            case 2: account = bAcct; break;
        }
```

Besides this, everything else stayed close to what I had originally planned to do minus a few logic or syntax errors.