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

Describe the errors you've encountered while working on this assignment. What caused the error and how do you overcome the error?

Error - Difficulty in making the account object change into a PersonalAcct or BusinessAcct object. This is because I didn't initialize it.

```
Object account;

{

if(action == "1")
{
    PersonalAcct bAcct = new PersonalAcct(bal, fN, lN, street, city, prov, postal
    account = bAcct;

}
else
{
    BusinessAcct pAcct = new BusinessAcct(bal, fN, lN, street, city, prov, postal
    account = pAcct;
}

    while(true)
    {
        System.out.println(account);
        System.out.println("(D)eposit" + '\n' +
                           "(W)ithdrawal" + '\n' +
                           "(A)ddress change" + '\n' +
                           "(Q)uit");

        action = input.next();

        if(action.equalsIgnoreCase("d"))
        {
            System.out.println("Enter deposit amount: ");
            amt = input.nextDouble();
            account.deposit(amt);
        }
        else if(action.equalsIgnoreCase("w"))
        {
            System.out.println("Enter withdrawal amount: ");
            amt = input.nextDouble();
            account.withdrawal(amt);
        }
        else if(action.equalsIgnoreCase("a"))
```

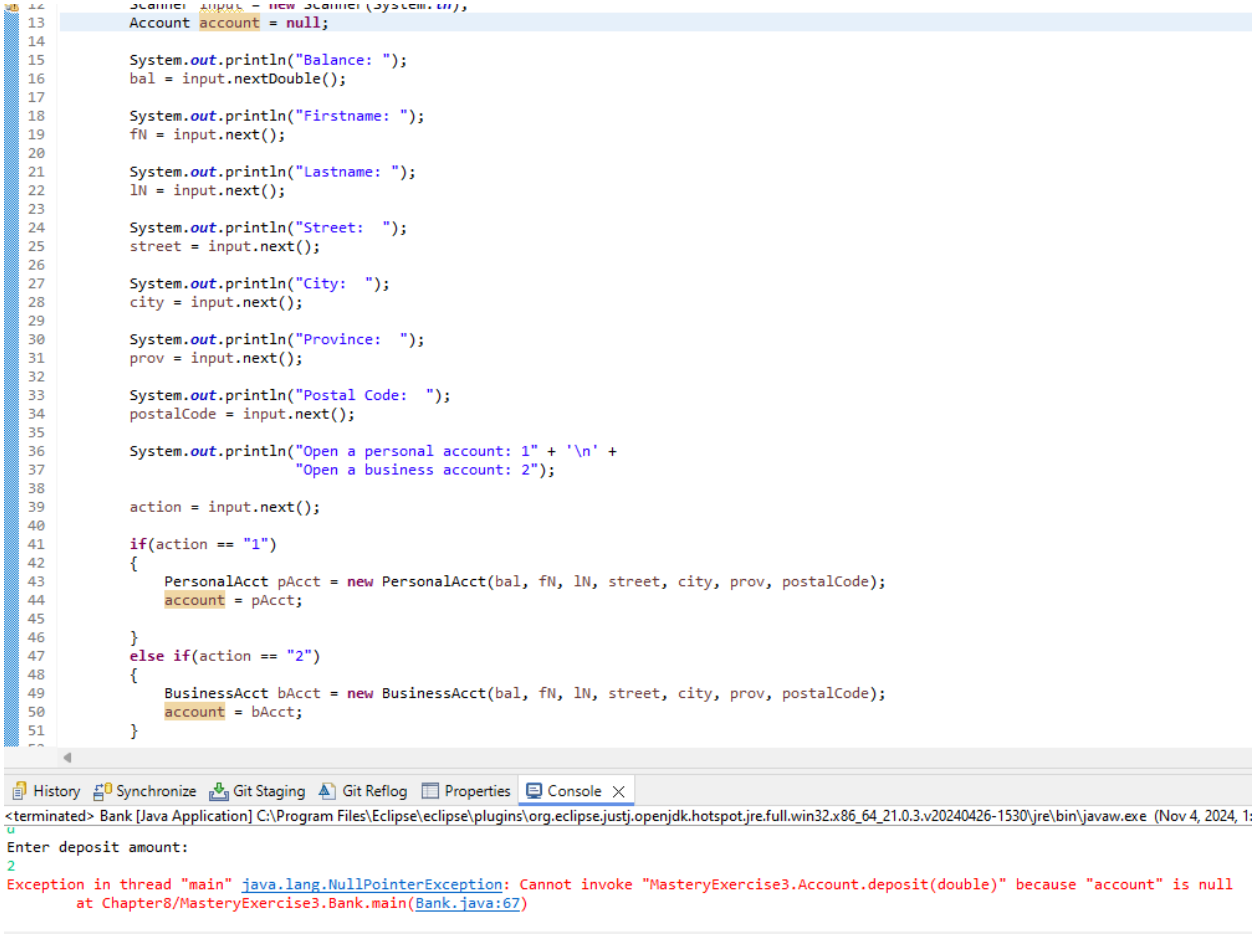
Solution - Made the object into an Account object. This led to more errors.

```
Account account;

{

if(action == "1")
{
    PersonalAcct bAcct = new PersonalAcct(bal, fN, lN,
    account = bAcct;

}
else
{
    BusinessAcct pAcct = new BusinessAcct(bal, fN, lN,
    account = pAcct;
}

    while(true)
    {
        System.out.println(account);
        System.out.println("(D)eposit" + '\n' +
                           "(W)ithdrawal" + '\n' +
                           "(A)ddress change" + '\n' +
                           "(Q)uit");

        action = input.next();

        if(action.equalsIgnoreCase("d"))
        {
            System.out.println("Enter deposit amount:
            amt = input.nextDouble();
            account.deposit(amt);
        }
        else if(action.equalsIgnoreCase("w"))
        {
            System.out.println("Enter withdrawal amoun
            amt = input.nextDouble();
            account.withdrawal(amt);
        }
        else if(action.equalsIgnoreCase("a"))
```

Error - Object Account account not initialized. Tried to initialize as null but that didn't work since it couldn't change its value and would remain as null.

```
if(action == "1")
{
    PersonalAcct pAcct = new PersonalAcct(bal, fN, lN, street, city,
    account = pAcct;

}
else if(action == "2")
{
    BusinessAcct bAcct = new BusinessAcct(bal, fN, lN, street, city,
    account = bAcct;
}

    while(true)
    {
        System.out.println(account);
        System.out.println("(D)eposit" + '\n' +
                           "(W)ithdrawal" + '\n' +
                           "(A)ddress change" + '\n' +
                           "(Q)uit");

        action = input.next();

        if(action.equalsIgnoreCase("d"))
        {
            System.out.println("Enter deposit amount: ");
            amt = input.nextDouble();
            account.deposit(amt);
        }
        else if(action.equalsIgnoreCase("w"))
        {
            System.out.println("Enter withdrawal amount: ");
            amt = input.nextDouble();
            account.withdrawal(amt);
        }
        else if(action.equalsIgnoreCase("a"))
        {
            System.out.println("Enter 'same' to keep original address

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

            System.out.println("City: ");
            city = input.next();
            if(city.equalsIgnoreCase("same"))
```

ynchronize | Git Staging | Git Reflog | Properties | Console



```
        Account account = null;

        System.out.println("Balance: ");
        bal = input.nextDouble();

        System.out.println("Firstname: ");
        fN = input.next();

        System.out.println("Lastname: ");
        lN = input.next();

        System.out.println("Street:  ");
        street = input.next();

        System.out.println("City:  ");
        city = input.next();

        System.out.println("Province:  ");
        prov = input.next();

        System.out.println("Postal Code:  ");
        postalCode = input.next();

        System.out.println("Open a personal account: 1" + '\n' +
                           "Open a business account: 2");

        action = input.next();

        if(action == "1")
        {
            PersonalAcct pAcct = new PersonalAcct(bal, fN, lN, street, city, prov, postalCode);
            account = pAcct;

        }
        else if(action == "2")
        {
            BusinessAcct bAcct = new BusinessAcct(bal, fN, lN, street, city, prov, postalCode);
            account = bAcct;
        }
```

```
History  Synchronize  Git Staging  Git Reflog  Properties  Console
<terminated> Bank [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.exe  (Nov 4, 2024, 1
0
Enter deposit amount:
2
Exception in thread "main" java.lang.NullPointerException: Cannot invoke "MasteryExercise3.Account.deposit(double)" because "account" is null
        at Chapter8/MasteryExercise3.Bank.main(Bank.java:67)
```

Solution - Initialized the account object as a personal account and used switch case statements within the while loop to change it to either a personal or business account object. It wouldn't matter if I initialized it to a business account.

```
//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;
    }
```

Error - Charged money before choosing an account type.

```
    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
```

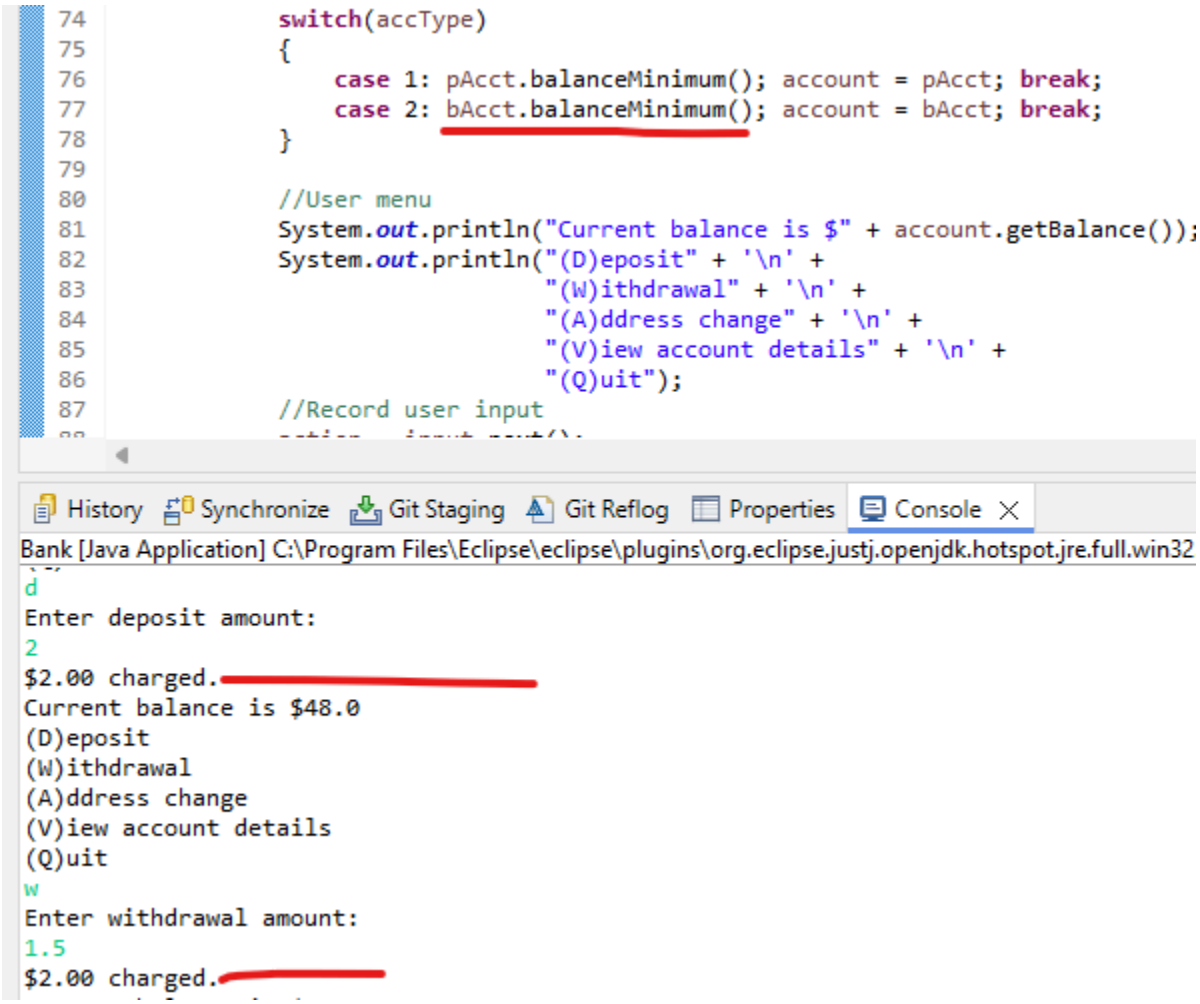
Made methods for the business or personal classes not account class

```
System.out.println("Open a personal account: 1" + '\n' +
                   "Open a business account: 2");

accType = input.nextInt();
Account account = pAcct;
while(true)
    {

        switch(accType)
        {
            case 1: account = pAcct; account.balanceMinimum(); break;
            case 2: account = bAcct; break;
```

Solution, lead to another error - Moved balanceMinimum to the switch case statements.

```
accType = input.nextInt();
Account account = pAcct;
while(true)
    {

        switch(accType)
        {
            case 1: pAcct.balanceMinimum(); account = pAcct; break;
            case 2: bAcct.balanceMinimum(); account = bAcct; break;
        }
```

Error - If balance was too low then each time the loop ran it would charge the account. This was because the balanceMinimum was run each time the loop ran which would charge the account.



Solution - Moved balanceMinimum to beneath creating the peronalAcct and BuinessAcct objects and changed the balanceMinimum method to display a warning of the amount charged if they choose to make that account. If the user chose to make a business or personal account then the amount would already be charged if applicable.

```java
//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;
    }


//Check if money has to be charged to account if balance is too low
public void balanceMinimum()
{
    if(super.getBalance() < 500)
    {
        super.withdrawal(10.00);
        System.out.println("$10.00 will be charged if a business acount is made.");
    }
    else
    {
        System.out.println("A business account can be made charge free.");
    }

}
```