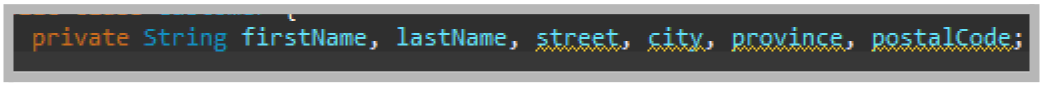
Credit Name: Inheritance and Polymorphism
Assignment Name: Account, PersonalAcct, BusinessAcct

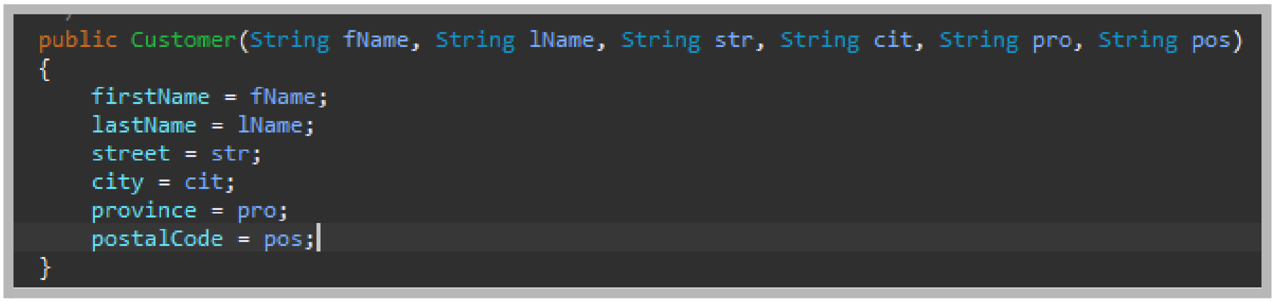
How has your program changed from planning to coding to now? Please explain?

I started working on this mastery by downloading the Account and Customer classes from D2L and scrolled through them to see that they were missing parts. I started working on the missing parts in Customer first.



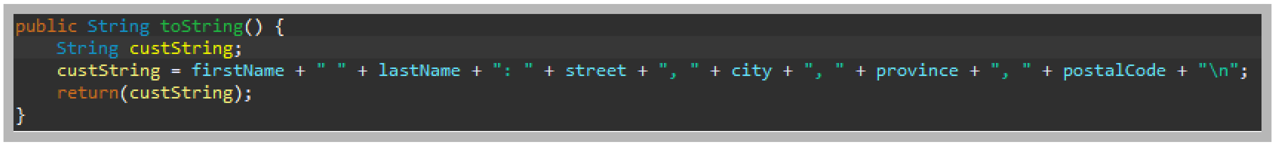
```
private String firstName, lastName, street, city, province, postalCode;
```

First I saw the comment that said I needed to add string variables for the street, city, province, and postal code so I created those.



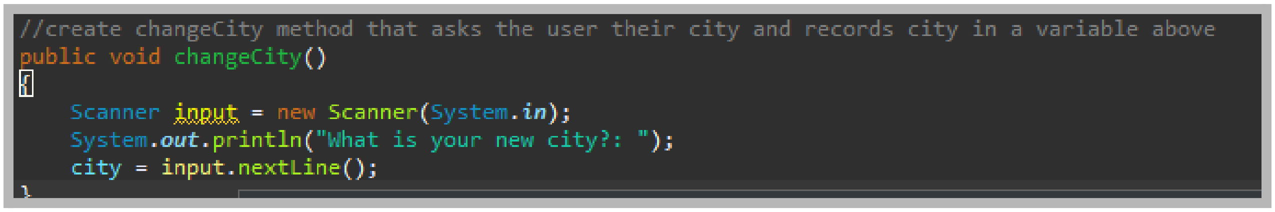
```
public Customer(String fName, String lName, String str, String cit, String pro, String pos)
{
    firstName = fName;
    lastName = lName;
    street = str;
    city = cit;
    province = pro;
    postalCode = pos;
}
```

Next I changed the constructor to incorporate these new variables.



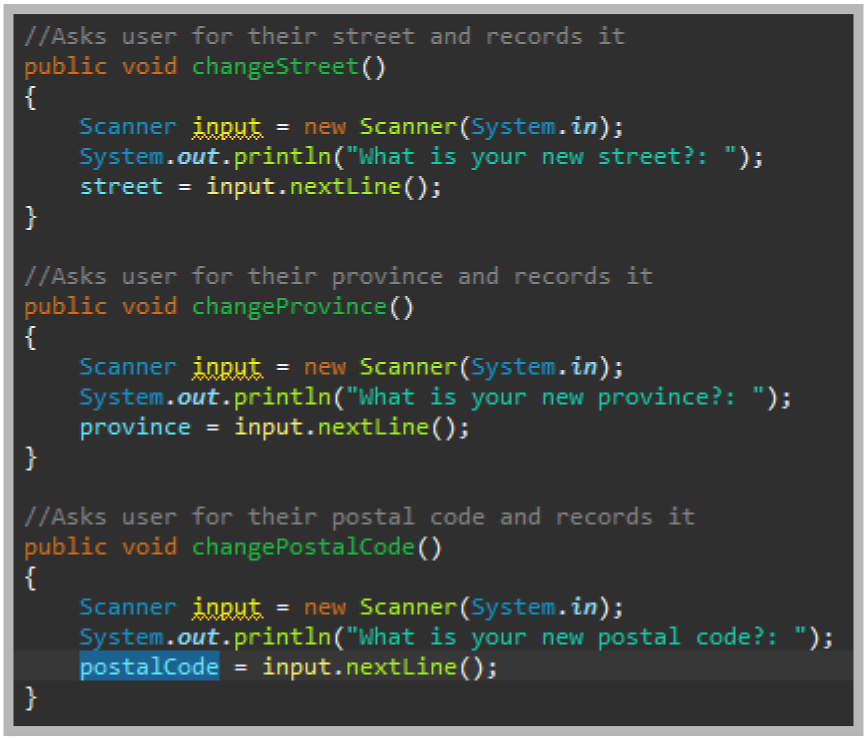
```
public String toString() {
    String custString;
    custString = firstName + " " + lastName + ": " + street + ", " + city + ", " + province + ", " + postalCode + "\n";
    return(custString);
}
```

I moved on to updating the toString method next because it seemed easier than the 4 methods above that I would have to include and wanted to get this out of the way.



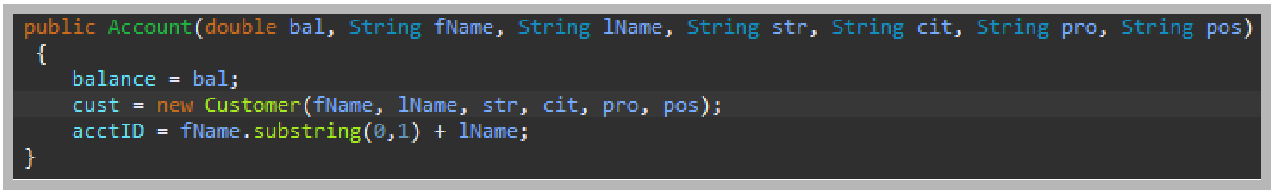
```
//create changeCity method that asks the user their city and records city in a variable above
public void changeCity()
{
    Scanner input = new Scanner(System.in);
    System.out.println("What is your new city?: ");
    city = input.nextLine();
}
```

I then started working on the 4 methods above the toString method starting with changeCity. All I had to do was add code to take the user's input and record it as the new city location.



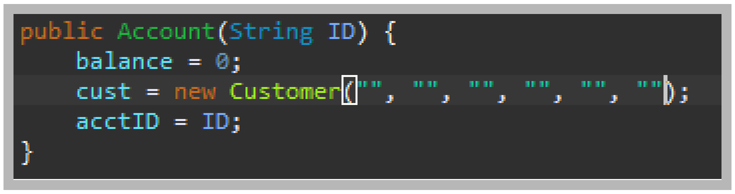
```java
//Asks user for their street and records it
public void changeStreet()
{
    Scanner input = new Scanner(System.in);
    System.out.println("What is your new street?: ");
    street = input.nextLine();
}

//Asks user for their province and records it
public void changeProvince()
{
    Scanner input = new Scanner(System.in);
    System.out.println("What is your new province?: ");
    province = input.nextLine();
}

//Asks user for their postal code and records it
public void changePostalCode()
{
    Scanner input = new Scanner(System.in);
    System.out.println("What is your new postal code?: ");
    postalCode = input.nextLine();
}
```

I used the same formula for the rest of the methods and changing variable names and the question being asked. With Customer completed I moved on to completing Account.



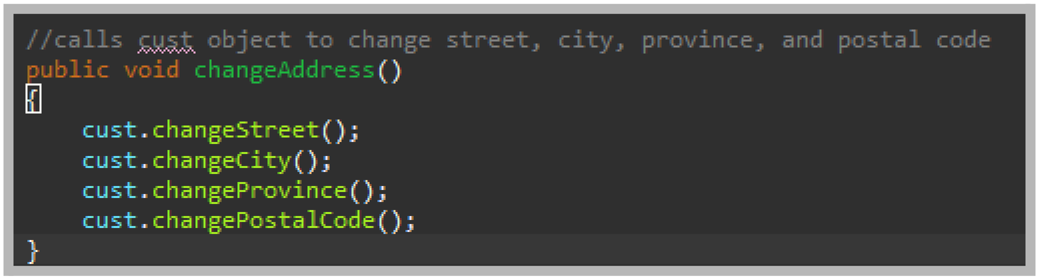
```java
public Account(double bal, String fName, String lName, String str, String cit, String pro, String pos)
 {
    balance = bal;
    cust = new Customer(fName, lName, str, cit, pro, pos);
    acctID = fName.substring(0,1) + lName;
}
```

I started Account by updating the constructor to add the variables in the parameters that the new Customer class needed.



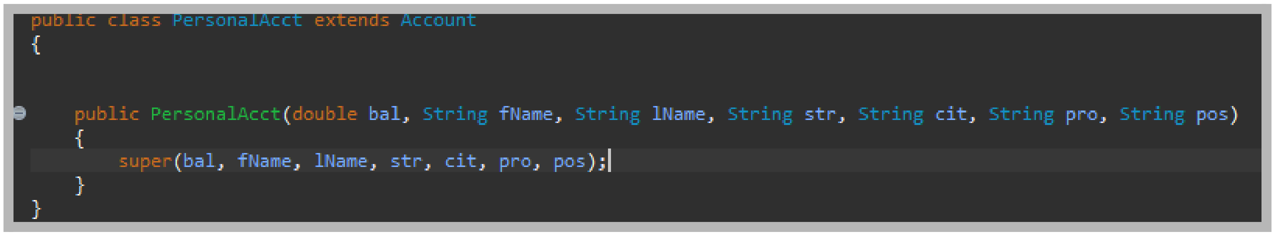
```
public Account(String ID) {
    balance = 0;
    cust = new Customer("", "", "", "", "", "");
    acctID = ID;
}
```

There was also code that needed to be updated in the second constructor to keep up with the changes made in Customer. I needed to add more empty Strings to the parameters of the customer object.



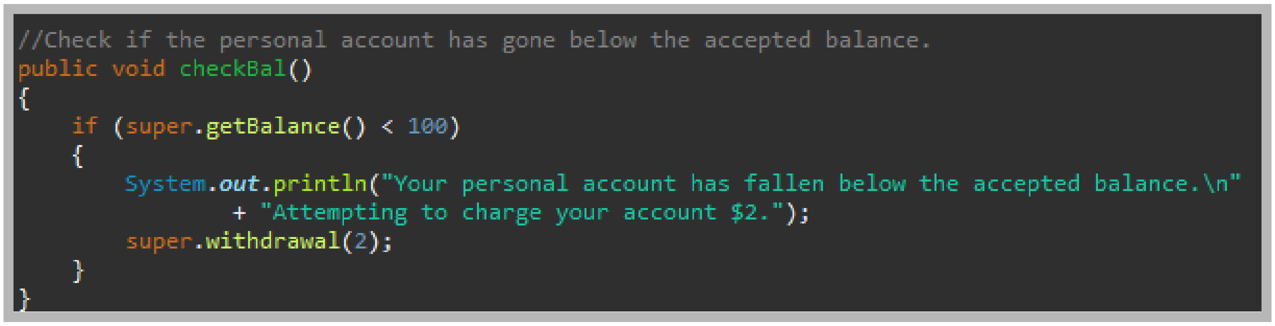
```
//calls cust object to change street, city, province, and postal code
public void changeAddress()
{
    cust.changeStreet();
    cust.changeCity();
    cust.changeProvince();
    cust.changePostalCode();
}
```

By adding the changeAddress method I was finished with the Account class and could move on to working on the PersonalAcct class.



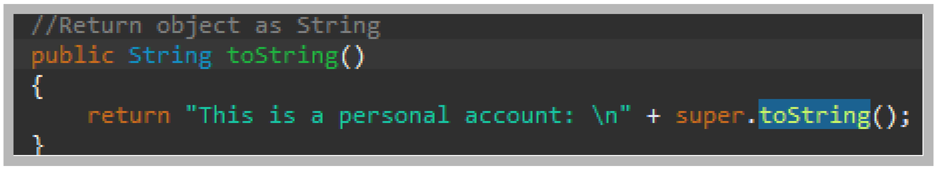
```
public class PersonalAcct extends Account
{

    public PersonalAcct(double bal, String fName, String lName, String str, String cit, String pro, String pos)
    {
        super(bal, fName, lName, str, cit, pro, pos);
    }
}
```

I started PersonalAcct by creating the constructor and using super to take in the parameters and use the account constructor.



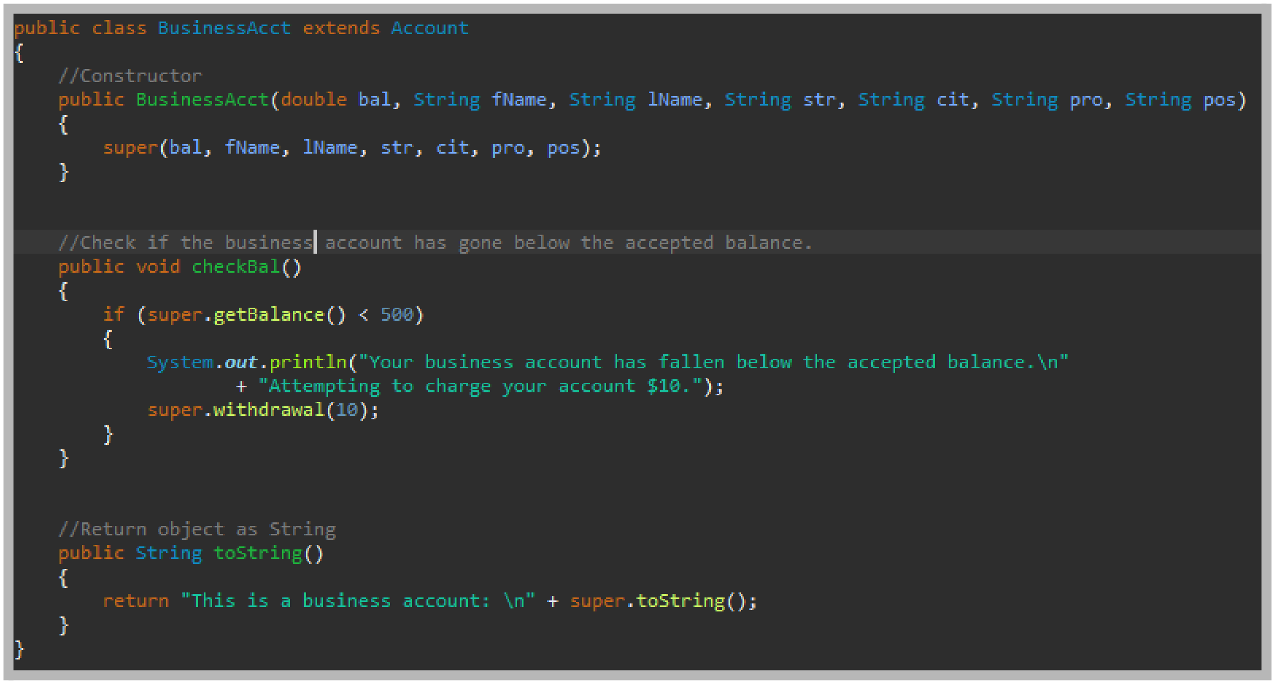
```
//Check if the personal account has gone below the accepted balance.
public void checkBal()
{
    if (super.getBalance() < 100)
    {
        System.out.println("Your personal account has fallen below the accepted balance.\n"
                + "Attempting to charge your account $2.");
        super.withdrawal(2);
    }
}
```

Next I started working on a method that would check if the balance has gone below the accepted value and if it has then it would try and withdraw $2 from the account.



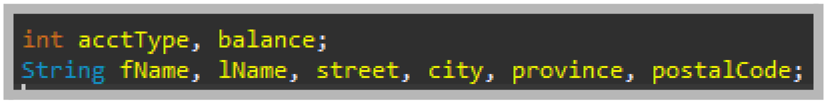
```
//Return object as String
public String toString()
{
    return "This is a personal account: \n" + super.toString();
}
```

Finally I finished the PersonalAcct class by adding the toString method.



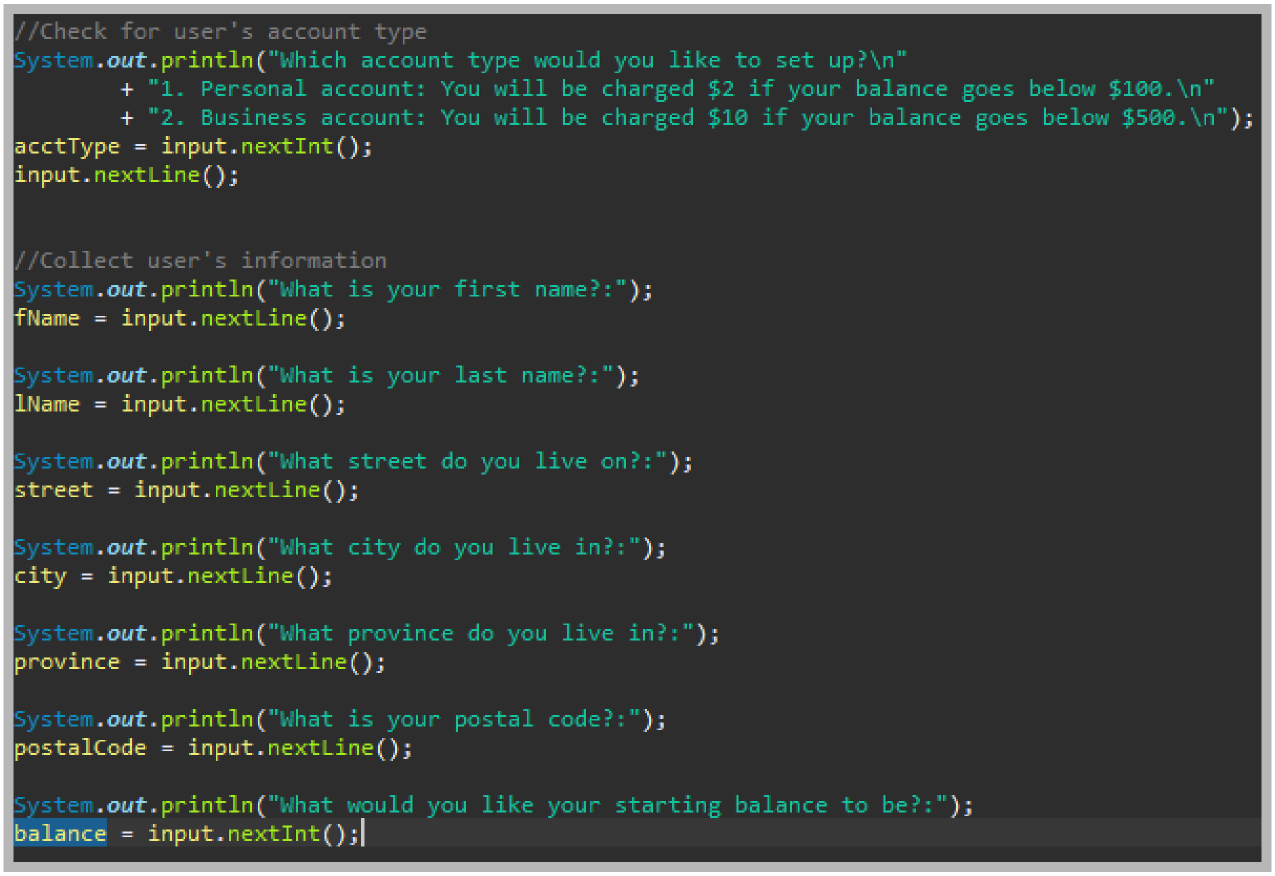
```
public class BusinessAcct extends Account
{
    //Constructor
    public BusinessAcct(double bal, String fName, String lName, String str, String cit, String pro, String pos)
    {
        super(bal, fName, lName, str, cit, pro, pos);
    }


    //Check if the business account has gone below the accepted balance.
    public void checkBal()
    {
        if (super.getBalance() < 500)
        {
            System.out.println("Your business account has fallen below the accepted balance.\n"
                    + "Attempting to charge your account $10.");
            super.withdrawal(10);
        }
    }


    //Return object as String
    public String toString()
    {
        return "This is a business account: \n" + super.toString();
    }
}
```

For the BusinessAcct class I copied the code from PersonalAcct and changed the amount charged, the balance needed to charge, and mentions of personal to business.



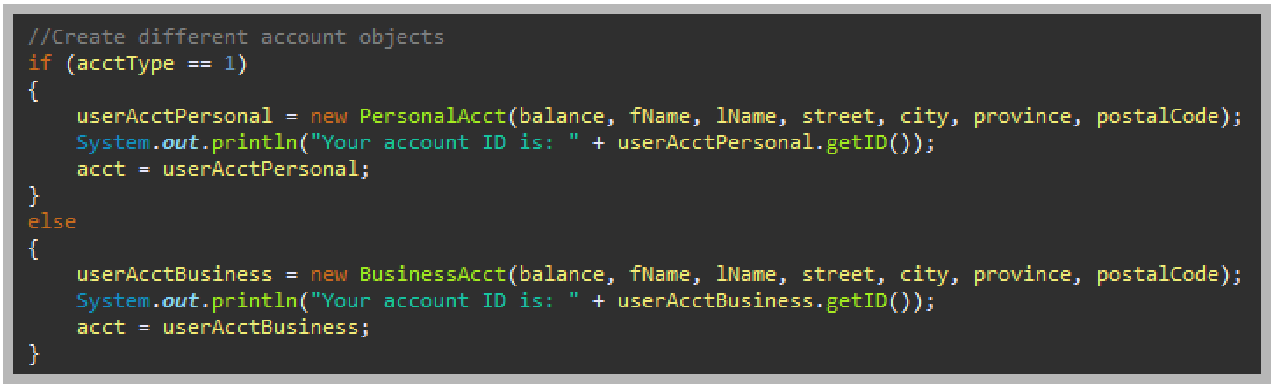
```
int acctType, balance;
String fName, lName, street, city, province, postalCode;
```

I started working on the test code by creating variables that would store the user's info that is needed for the constructor as well as a variable that would store which account the user wants to set up.



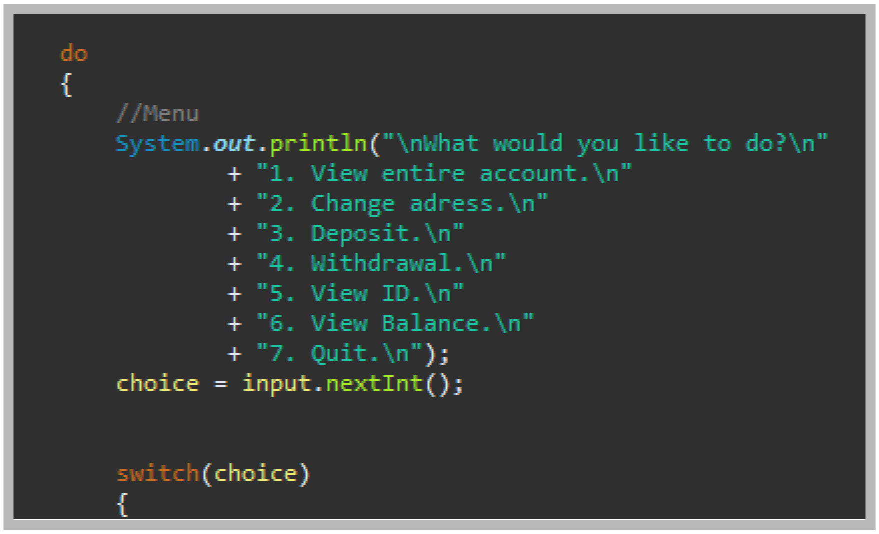
```
//Check for user's account type
System.out.println("Which account type would you like to set up?\n"
        + "1. Personal account: You will be charged $2 if your balance goes below $100.\n"
        + "2. Business account: You will be charged $10 if your balance goes below $500.\n");
acctType = input.nextInt();
input.nextLine();


//Collect user's information
System.out.println("What is your first name?:");
fName = input.nextLine();

System.out.println("What is your last name?:");
lName = input.nextLine();

System.out.println("What street do you live on?:");
street = input.nextLine();

System.out.println("What city do you live in?:");
city = input.nextLine();

System.out.println("What province do you live in?:");
province = input.nextLine();

System.out.println("What is your postal code?:");
postalCode = input.nextLine();

System.out.println("What would you like your starting balance to be?:");
balance = input.nextInt();
```

I then added the code that would ask the user questions that the variables would then store for later use.



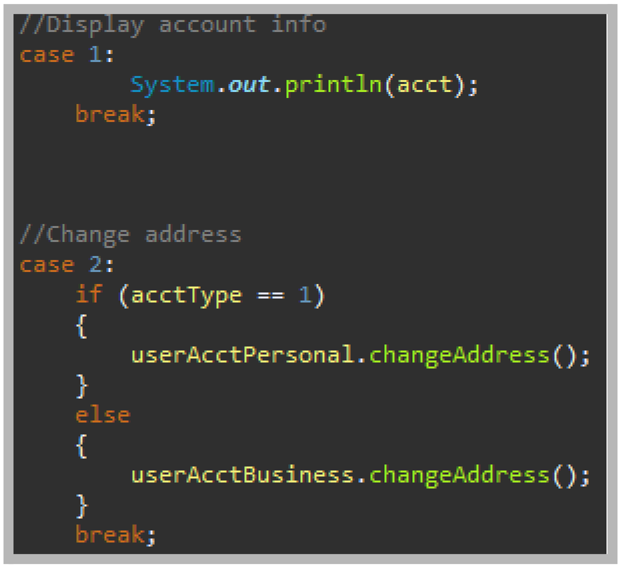
```
//Create different account objects
if (acctType == 1)
{
    userAcctPersonal = new PersonalAcct(balance, fName, lName, street, city, province, postalCode);
    System.out.println("Your account ID is: " + userAcctPersonal.getID());
    acct = userAcctPersonal;
}
else
{
    userAcctBusiness = new BusinessAcct(balance, fName, lName, street, city, province, postalCode);
    System.out.println("Your account ID is: " + userAcctBusiness.getID());
    acct = userAcctBusiness;
}
```

Next I created an if statement to check for the account type and create a PersonalAcct or a BusinessAcct. I also created an Account object that would reduce the number of conditionals I would need later.



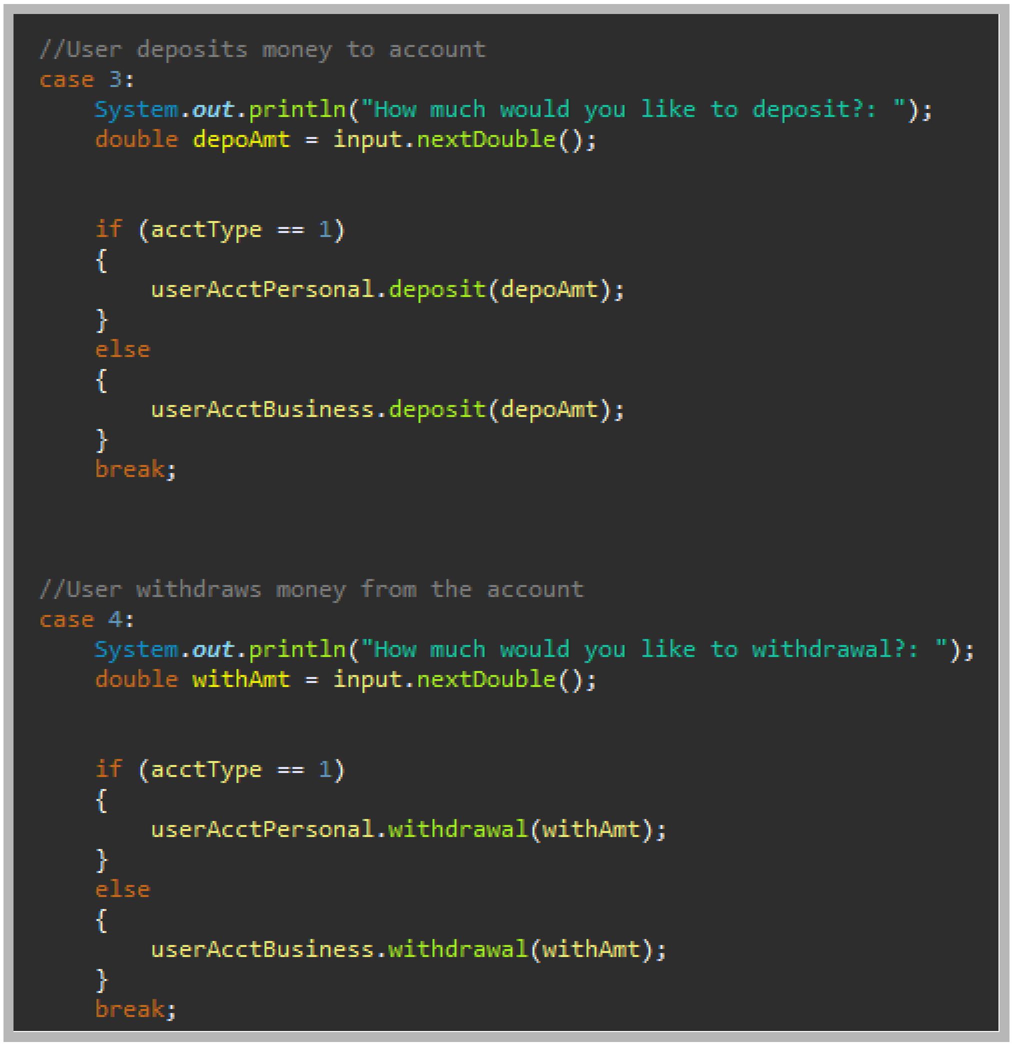
```
do
{
    //Menu
    System.out.println("\nWhat would you like to do?\n"
            + "1. View entire account.\n"
            + "2. Change adress.\n"
            + "3. Deposit.\n"
            + "4. Withdrawal.\n"
            + "5. View ID.\n"
            + "6. View Balance.\n"
            + "7. Quit.\n");
    choice = input.nextInt();


    switch(choice)
    {
```

I added a do while loop after that would contain a menu and perform the actions that the user wanted.



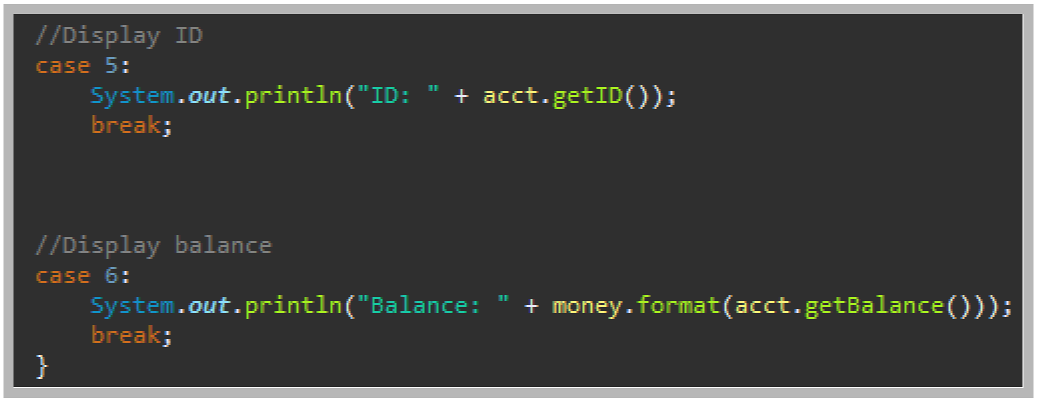
```
//Display account info
case 1:
        System.out.println(acct);
    break;



//Change address
case 2:
    if (acctType == 1)
    {
        userAcctPersonal.changeAddress();
    }
    else
    {
        userAcctBusiness.changeAddress();
    }
    break;
```

I started with the first two options that would just display the entire account, and call the method to change the address.



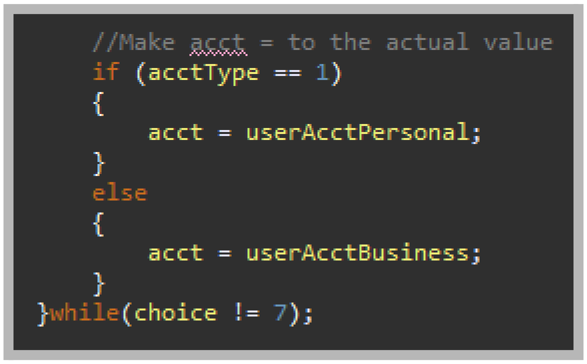
```
//User deposits money to account
case 3:
    System.out.println("How much would you like to deposit?: ");
    double depoAmt = input.nextDouble();


    if (acctType == 1)
    {
        userAcctPersonal.deposit(depoAmt);
    }
    else
    {
        userAcctBusiness.deposit(depoAmt);
    }
    break;



//User withdraws money from the account
case 4:
    System.out.println("How much would you like to withdrawal?: ");
    double withAmt = input.nextDouble();


    if (acctType == 1)
    {
        userAcctPersonal.withdrawal(withAmt);
    }
    else
    {
        userAcctBusiness.withdrawal(withAmt);
    }
    break;
```

Next I moved on to the cases for depositing and withdrawing from the account. Both of them ask how much the user would like to do and then checks which account they have and calls that method.



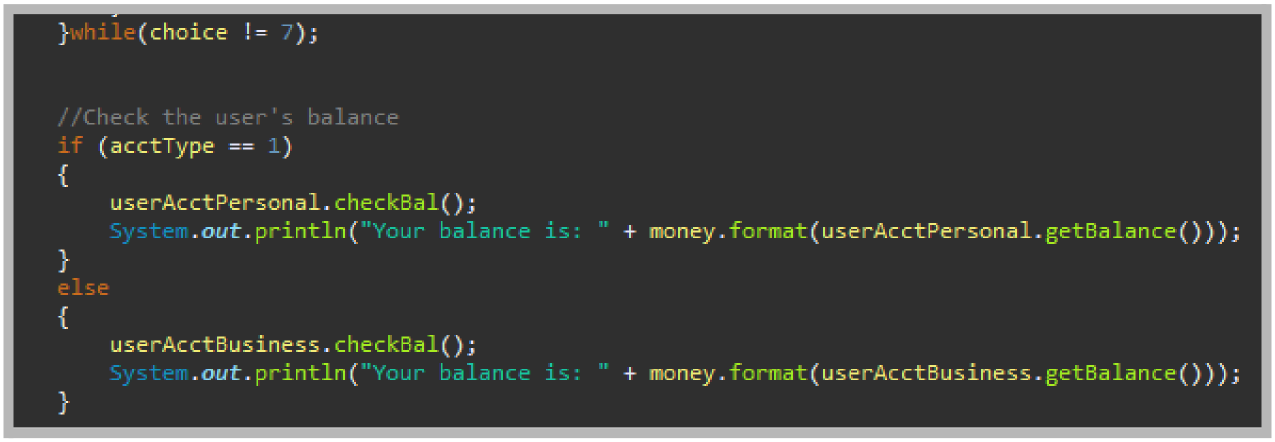
```
//Display ID
case 5:
    System.out.println("ID: " + acct.getID());
    break;



//Display balance
case 6:
    System.out.println("Balance: " + money.format(acct.getBalance()));
    break;
}
```

Finally I added the last two options that would just display the ID and balance that the account has.



```
    //Make acct = to the actual value
    if (acctType == 1)
    {
        acct = userAcctPersonal;
    }
    else
    {
        acct = userAcctBusiness;
    }
}while(choice != 7);
```

At the end of the loop I added code that would check for which account the user picked and then make the general account equal to the possibly modified account.



```
}while(choice != 7);


//Check the user's balance
if (acctType == 1)
{
    userAcctPersonal.checkBal();
    System.out.println("Your balance is: " + money.format(userAcctPersonal.getBalance()));
}
else
{
    userAcctBusiness.checkBal();
    System.out.println("Your balance is: " + money.format(userAcctBusiness.getBalance()));
}
```

Finally, outside of the loop I added code to call the checkBal method and tell the user the final balance just in case they were fined.