Credit Name: CSE2140 2nd Language Programming
Assignment Name: My Savings Mastery

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

Created a public class with a private double variable called total.

```
public class MySavings {
    private double total;
```

Prepared decimal format to shorten any long decimals to 2 decimal places.

```
//Shortens number to 2 decimal places
DecimalFormat df = new DecimalFormat("#0.00");
```

Created constructor method with default values for total.

```
//Constructor method with default values
public MySavings() {
    total = 0;
}
```

Created method that overloads constructor method with a double as the parameter.

```
//Overloading constructor method
public MySavings(double t) {
    total = t;
}
```

Created an access method that will return total as a double.

```
//Access method
public double getTotal() {
    return total;
}
```

Created a modifier method with a double as the parameter that you can change/alter the total, to be able to add coins. Doesn't return a value, only initializes the variable.

```
//Modifier method
public void setTotal(double t) {
    total += t;
}
```

Created methods for each type of coin to be able to add their respective values to the total.

```
//Add coins
public void addPennies() {
    total = total + 0.01;
}
public void addNickels() {
    total = total + 0.05;
}
public void addDimes() {
    total = total + 0.1;
}
public void addQuarters() {
    total = total + 0.25;
}
```

Created a string method to be able to withdraw money with a double as the parameters. Subtracts double from the total. Returns a string stating the new balance.

```
//Subtract money
public String takeMoney(double newTotal) {
    total -= newTotal;
    return("Your new balance is: $" + df.format(total));
}
```

Created a string method to display the total balance. Returns a string stating balance.

```
//Show total
```

In the client code in the main method, delcare the variables as int and double as we are dealing with numbers.

```
public static void main(String[] args) {
    // TODO Auto-generated method stub
    //Declaration area
    int option;
    double withdraw;
```

Prepared for user to input an answer.

```
//Prepare for user input
Scanner userInput = new Scanner(System.in);
```

Create a new object, setting the total as 0.

```
//Create new object
MySavings amount = new MySavings();
amount.setTotal(0);
```

Display options to user, prompting them to input an options. Initiazlize the input.

```
//Display options and prompt user to input an option and record it
System.out.print("1. Show total money in bank."
    + "\n2. Add a penny."
    + "\n3. Add a nickel."
    + "\n4. Add a dime."
    + "\n5. Add a quarter."
    + "\n6. Take out money from the bank."
    + "\nEnter 0 to quit."
    + "\nPlease enter your choice: ");
option = userInput.nextInt();
```

Used a while loop to continue asking user to input option an option until the user input is equal or less than 0. Used a switch case to compare which method, along with object, to use based on user input. Display that the respective coin has been added. For withdrawl, prompt user to input amount to be withdrawn and initialize it. Using the input as the parameters, display new balance.

```
//Continue loop until user input invalid
while (option >= 0) {

    //Display answer to user based on option picked
    switch(option) {
    case 0:
        break;
    case 1: System.out.println(amount.showTotal()); //Shows total
        break;
    case 2: amount.addPennies(); //Adds 0.01 to total
        System.out.println("You have added 1 penny.");
        break;
    case 3: amount.addNickels(); //Adds 0.05 to total
        System.out.println("You have added 1 nickel.");
        break;
    case 4: amount.addDimes(); //Adds 0.1 to total
        System.out.println("You have added 1 dime.");
        break;
    case 5: amount.addQuarters(); //Adds 0.25 to total
        System.out.println("You have added 1 quarter.");
        break;
    case 6: System.out.print("How much money would you like to withdraw?" //Prompt user to input number to withdraw
            + "\n$");
        withdraw = userInput.nextDouble(); //Initialize input
        System.out.println(amount.takeMoney(withdraw)); //Shows total after withdrawal
        break;
    default: System.out.println("Invalid option.");
    }
```

Prompt user to input another option, if the option is less than or equal to 0, while loop will break.

```
//Prompt user to input new option to continue or end loop, initialize input
System.out.print("Please enter your choice: ");
```

```
public String showTotal() {
    return ("Your total balance is: $" + df.format(getTotal()));
}
```

```
option = userInput.nextInt();
//Ends loop if option is 0
if (option <= 0) {
    break;
}
```