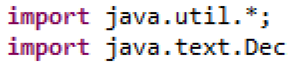
Assignment name: MySavings Mastery Project - Chapter 7
Student name: Misha Stanev

Reflection Log

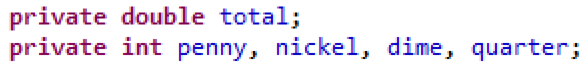
Planning: During the planning stage I planned on coding this application without using methods
Coding: During code I realized that not using methods would make this application much more confusing and longer than it needs to be Now: I stuck with my plan during the coding phase and used methods

Import scanner and decimal format object to format numbers (Shown below)



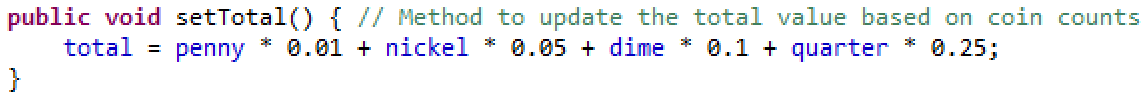
```
import java.util.*;
import java.text.DecimalFormat;
```

Create instance variables (Shown below)



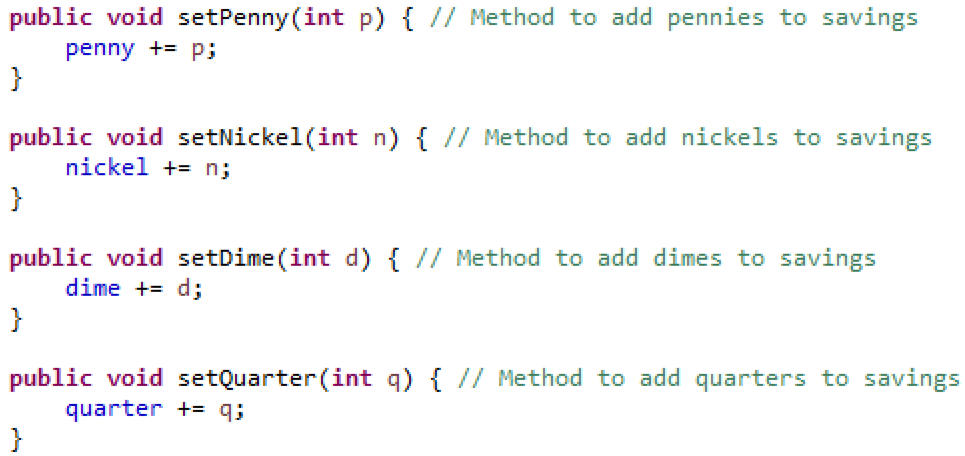
```
private double total;
private int penny, nickel, dime, quarter;
```

Method to calculate total value of coins (Shown below)



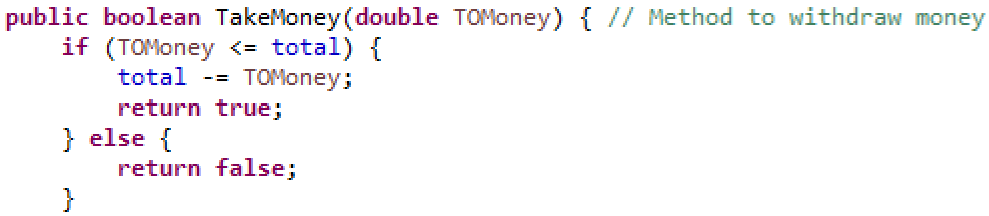
```
public void setTotal() { // Method to update the total value based on coin counts
    total = penny * 0.01 + nickel * 0.05 + dime * 0.1 + quarter * 0.25;
}
```

Create methods to add each coin type (Shown below)



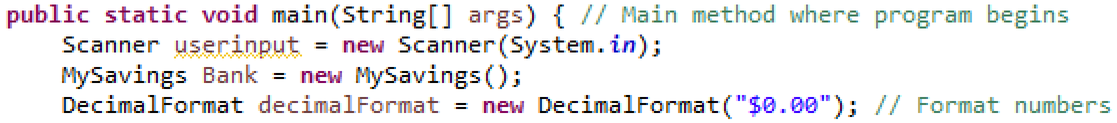
```
public void setPenny(int p) { // Method to add pennies to savings
    penny += p;
}

public void setNickel(int n) { // Method to add nickels to savings
    nickel += n;
}

public void setDime(int d) { // Method to add dimes to savings
    dime += d;
}

public void setQuarter(int q) { // Method to add quarters to savings
    quarter += q;
}
```

Method to withdraw money (Shown below)



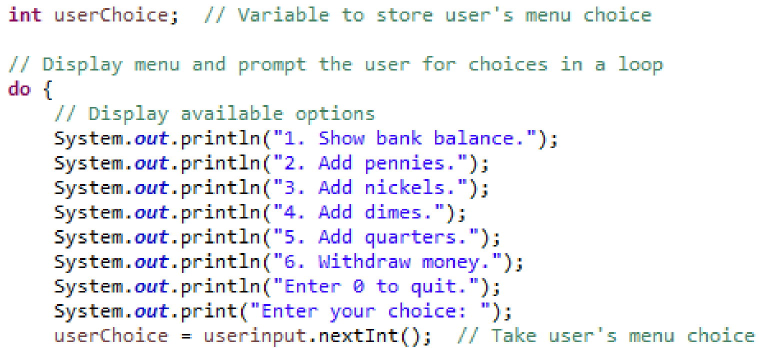
```
public boolean TakeMoney(double TOMoney) { // Method to withdraw money
    if (TOMoney <= total) {
        total -= TOMoney;
        return true;
    } else {
        return false;
    }
```

Create main method where program begins and create a new savings object named Bank, then use decimal format to format money (Shown below)



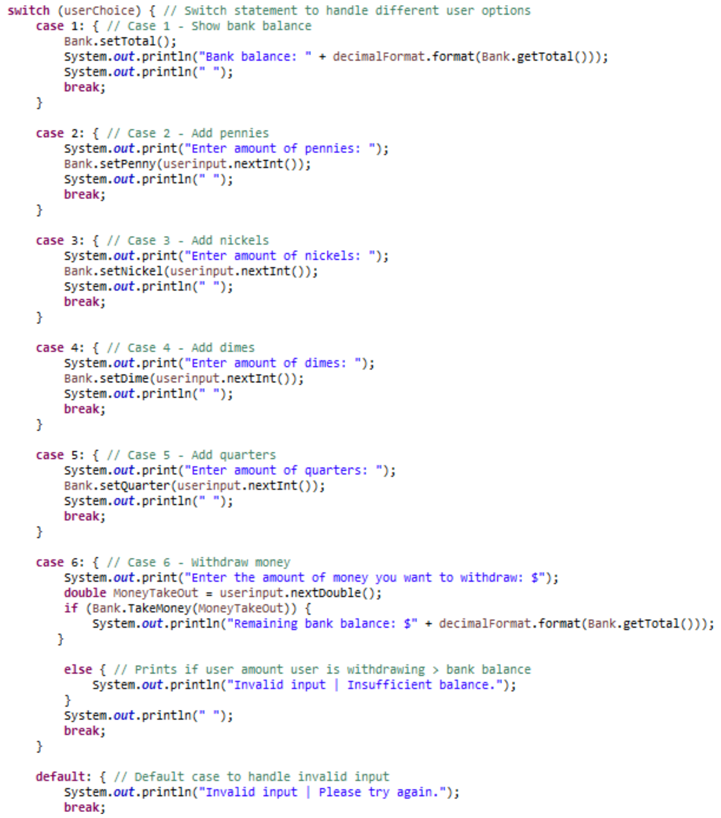
```
public static void main(String[] args) { // Main method where program begins
     Scanner userinput = new Scanner(System.in);
     MySavings Bank = new MySavings();
     DecimalFormat decimalFormat = new DecimalFormat("$0.00"); // Format numbers
```

Display available options for user to choose from, then save int input to userChoice (Shown below)



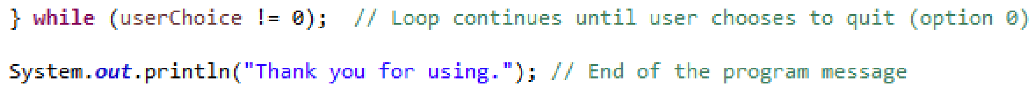
```
int userChoice;  // Variable to store user's menu choice

// Display menu and prompt the user for choices in a loop
do {
    // Display available options
    System.out.println("1. Show bank balance.");
    System.out.println("2. Add pennies.");
    System.out.println("3. Add nickels.");
    System.out.println("4. Add dimes.");
    System.out.println("5. Add quarters.");
    System.out.println("6. Withdraw money.");
    System.out.println("Enter 0 to quit.");
    System.out.print("Enter your choice: ");
    userChoice = userinput.nextInt();  // Take user's menu choice
```

Create switch statement which prints different method depending on int value of userChoice (Shown below)



```
switch (userChoice) { // Switch statement to handle different user options
    case 1: { // Case 1 - Show bank balance
        Bank.setTotal();
        System.out.println("Bank balance: " + decimalFormat.format(Bank.getTotal()));
        System.out.println(" ");
        break;
    }

    case 2: { // Case 2 - Add pennies
        System.out.print("Enter amount of pennies: ");
        Bank.setPenny(userinput.nextInt());
        System.out.println(" ");
        break;
    }

    case 3: { // Case 3 - Add nickels
        System.out.print("Enter amount of nickels: ");
        Bank.setNickel(userinput.nextInt());
        System.out.println(" ");
        break;
    }

    case 4: { // Case 4 - Add dimes
        System.out.print("Enter amount of dimes: ");
        Bank.setDime(userinput.nextInt());
        System.out.println(" ");
        break;
    }

    case 5: { // Case 5 - Add quarters
        System.out.print("Enter amount of quarters: ");
        Bank.setQuarter(userinput.nextInt());
        System.out.println(" ");
        break;
    }

    case 6: { // Case 6 - Withdraw money
        System.out.print("Enter the amount of money you want to withdraw: $");
        double MoneyTakeOut = userinput.nextDouble();
        if (Bank.TakeMoney(MoneyTakeOut)) {
            System.out.println("Remaining bank balance: $" + decimalFormat.format(Bank.getTotal()));
       }

        else { // Prints if user amount user is withdrawing > bank balance
            System.out.println("Invalid input | Insufficient balance.");
        }
        System.out.println(" ");
        break;
    }

    default: { // Default case to handle invalid input
        System.out.println("Invalid input | Please try again.");
        break;
```

Loop continues until user enters 0 (Shown below)



```
} while (userChoice != 0);  // Loop continues until user chooses to quit (option 0)

System.out.println("Thank you for using."); // End of the program message
```