

ReflectionLog MySavings/PiggyBank

My Savings

```
public static <piggyBank> void main(String[] args) {  
    //links piggybank to mysavings  
    PiggyBank pb = new PiggyBank();  
    Scanner in = new Scanner(System.in);  
    DecimalFormat deca = new DecimalFormat("#.##");
```

The code defines the main method and creates a PiggyBank object, pb. It also initializes a Scanner object, in, to read user input and a DecimalFormat object, deca, to format numbers to two decimal places.

```
int choice = 0;  
  
do {  
    //prompt user to enter a number to pick which type of metric conversion they want  
    System.out.println("1. Show total in bank");  
    System.out.println("2. Add a penny");  
    System.out.println("3. Add a nickel");  
    System.out.println("4. Add a dime");  
    System.out.println("5. Add a quarter");  
    System.out.println("6. Take money out of your bank");  
    System.out.println("Enter 0 to quit");  
    System.out.print("Enter your choice: ");  
    choice = in.nextInt();
```

The code displays a menu of options for the user to choose from, such as adding coins, checking the total balance, or withdrawing money. The program repeatedly prompts the user for input until they choose to quit by entering 0.

```
//switch case to handle different user choices.
switch (choice) {

    //Show total in bank when user picks 1.
    case 1:
        System.out.println("Total in bank: $" + deca.format(pb.bankTotal()));
        break;

        //Add a penny to the bank.
    case 2:

        pb.penny(1);
        System.out.println("Added 1 penny");
        break;

        //Add a nickel when user picks 3.
    case 3:

        pb.nickel(1);
        System.out.println("Added 1 nickel");
        break;

        //Add a dime when user picks 4.
    case 4:

        pb.dime(1);
        System.out.println("Added 1 dime");
        break;
}
```

The code uses a switch statement to handle user choices. If the user selects option 1, it displays the total balance in the bank, formatted to two decimal places. Options 2 and 3 add one penny or one nickel to the bank, respectively, with a confirmation message. Option 4 adds one dime to the bank and confirms the action with a printed message.

```

        pb.dime(1);
        System.out.println("Added 1 dime");
        break;

        //Add a quarter when user picks 5.
    case 5:

        pb.quarter(1);
        System.out.println("Added 1 quarter");
        break;

        //Take money out of piggy bank when user picks 6.
    case 6:

        System.out.println("You took $" + deca.format(pb.bankTotal()) + " out of the bank" );
        pb.takeOut();
        break;

        //option to quit/leave the code.
    case 0:
        System.out.println("Quit successful");
        break;
    }
}
while ( choice != 0);

```

The code handles user input for adding a quarter (option 5) to the bank, and removes money from the bank (option 6), displaying the total amount withdrawn. If the user selects option 0, the program prints a message confirming the quit action and exits the loop. The do-while loop continues until the user chooses to quit. Each action is confirmed with a printed message indicating the operation performed.

PiggyBank

```
// Instance variables to store the number of each type of coin
private int pen;    // Number of pennies

private int nic;    // Number of nickels

private int dime;   // Number of dimes

private int quart;  // Number of quarters

// Constructor to initialize the coin counts to 0
public PiggyBank() {

    pen = 0; // Initialize pennies to 0

    nic = 0; // Initialize nickels to 0

    dime = 0; // Initialize dimes to 0

    quart = 0; // Initialize quarters to 0
}
```

This code defines instance variables to store the number of each type of coin in the piggy bank, such as pennies, nickels, dimes, and quarters. The constructor `PiggyBank()` initializes all these coin counts to 0, ensuring that the piggy bank starts empty when an object of this class is created. The variables `pen`, `nic`, `dime`, and `quart` represent the number of pennies, nickels, dimes, and quarters, respectively.

```

//Method to add pennies to the piggy bank
public void penny(int num) {

    pen += num; //Increase the number of pennies by 'num'
}

// Method to add nickels to the piggy bank
public void nickel(int num) {

    nic += num; //Increase the number of nickels by 'num'
}

//Method to add dimes to the piggy bank
public void dime(int num) {

    dime += num; //Increase the number of dimes by 'num'
}

// Method to add quarters to the piggy bank
public void quarter(int num) {

    quart += num; // Increase the number of quarters by 'num'
}

//Method to calculate the total value of the coins in the piggy bank
public double bankTotal() {

    //Returns the total value by multiplying the number of each coin with its value and summing them up
    return (pen * 0.01) + (nic * 0.05) + (dime * 0.1) + (quart * 0.25);
}

```

These methods allow the user to add specific coins (pennies, nickels, dimes, and quarters) to the piggy bank by increasing the respective coin counts. The bankTotal() method calculates the total value of the coins in the piggy bank by multiplying the count of each coin by its value and summing them up. The methods ensure that the coin counts are updated properly, and the total amount can be calculated based on the coins added. This structure allows for easy tracking and updating of the piggy bank's contents.

```

//Method to empty the piggy bank (remove all coins)
public void takeOut() {

    pen = 0;    //Reset the number of pennies to 0

    nic = 0;    //Reset the number of nickels to 0

    dime = 0;   //Reset the number of dimes to 0

    quart = 0;  //Reset the number of quarters to 0
}

//Method to return a string representation of the current coin counts in the piggy bank
public String toString() {

    //Returns a string with the counts of each type of coin
    String total = "Penny: " + pen + " Nickel: " + nic + " Dime: " + dime + " Quarters: " + quart;

    return total;
}

//Additional method bankTotal1() that currently returns 0, potentially unused
public double bankTotal1() {

```

The takeOut() method resets all the coin counts (pennies, nickels, dimes, and quarters) to 0, effectively emptying the piggy bank. The toString() method returns a string representation of the current coin counts in the piggy bank, summarizing the number of pennies, nickels, dimes, and quarters. These methods allow for easy management and display of the piggy bank's contents.