DigitExtractor Mastery ReflectionLog

Foreword: Screenshots were taken on both my home and school computer, hence the difference between the package in some of the screenshots.

Below is the first rendition of the TestDigitExtractor class:

```java
package skibidi;

import java.util.Scanner;

public class TestDigitExtractor {

    //Object that returns user input
    public static int number() {
        @SuppressWarnings("resource")
        //Prepare for user input
        Scanner userInput = new Scanner(System.in);
        //Record user input
        int value = userInput.nextInt();
        //Returns user input
        return value;
    }

    public static int choice() {
        @SuppressWarnings("resource")
        //Prepare for user input
        Scanner userInput = new Scanner(System.in);
        //Record user input
        int choice = userInput.nextInt();
        //Returns user input
        return choice;
    }

    public static void main(String[] args) {
        //Prompt user for input
        System.out.println("Enter an integer: ");
        int value = number();

        String input = "";

        //Loops while input is not 0
        while (input != "Q") {
        //Prompt user for input
        System.out.println("Show (W)hole number.");
        System.out.println("Show (O)nes place number.");
        System.out.println("Show (T)ens place number");
        System.out.println("Show(H)undreds place number");
        System.out.println("(Q)uit.");
        System.out.println("Enter your choice: ");

        //Calls PiggyBank Object from MySavings Class with the input variable
        DigitExtractor.Num(value, input);

        }
```

Within the class are the number(), choice(), and main object. The number(), and choice(), both record and return user input as an integer. The main object prompts the user, then calls the number() object and assigns it to the value variable. An input variable is declared, and a while loop runs when the input variable isn't “Q” - This is a logic error. Within the while loop is a series of prompts for the user, along with a call to the Num() object in the DigitExtractor class. Within this first rendition the choice() object isn’t utilized.

Below is the first rendition of the DigitExtractor class:

```
package skibidi;

public class DigitExtractor {

    public static void Num(int number, String choice) {

        int hundreds = number / 100;

        int tens = number / 10 % 10;

        int ones = number % 10;

        if (choice == "W") {
            System.out.println("The hundreds place digit is: " + hundreds);
        } else if (choice == "T") {
            System.out.println("The tens place digit is: " + tens);
        } else if (choice == "O") {
            System.out.println("The ones place digit is: " + ones);
        }

    }
}
```

The DigitExtractor class contains a Num() object that takes an int and a String. Then using the int, it declares and initializes a hundreds, tens, and ones variable accordingly. Then depending on the string it prints the value of the appropriate variable - Logic rror here.

Below is the second rendition of the TestDigitExtractor class:

```java
package skibidi;

import java.util.Scanner;

public class TestDigitExtractor {

    //Object that returns user input
    public static int number() {
        @SuppressWarnings("resource")
        //Prepare for user input
        Scanner userInput = new Scanner(System.in);
        //Record user input
        int value = userInput.nextInt();
        //Returns user input
        return value;
    }

    public static String choice() {
        @SuppressWarnings("resource")
        //Prepare for user input
        Scanner userInput = new Scanner(System.in);
        //Record user input
        String choice = userInput.nextLine();
        //Returns user input
        return choice;
    }

    public static void main(String[] args) {
        //Prompt user for input
        System.out.print("Enter an integer: ");
        int value = number();

        //Declares new variable input
        String input = "";

        //Loops while input is not 0
        while (input != "Q") {

            //Prompt user for input
            System.out.println("Show (W)hole number.");
            System.out.println("Show (O)nes place number.");
            System.out.println("Show (T)ens place number");
            System.out.println("Show (H)undreds place number");
            System.out.println("(Q)uit.");
            System.out.print("Enter your choice: ");

            //Declares input variable as the return of the number() object
            input = choice();

            //Calls Num Object from DigitExtractor Class with the value and string input
            DigitExtractor.Num(value, input);

        }
```

First I fixed the choice() object to return a string value, and initialized the input variable as the object within the while loop. Alongside this I added additional comments.

Below is the second rendition of the DigitExtractor class:

```java
package skibidi;

public class DigitExtractor {

    public static void Num(int number, String choice) {

        int hundreds = number / 100;

        int tens = number / 10 % 10;

        int ones = number % 10;

        if (choice == "W") {
            System.out.println("The number is: " + number);
        } else if (choice == "H") {
            System.out.println("The hundreds place digit is: " + hundreds);
        } else if (choice == "T") {
            System.out.println("The tens place digit is: " + tens);
        } else if (choice == "O") {
            System.out.println("The ones place digit is: " + ones);
        }

    }

}
```

I fixed the if else if statement so that the options align with what the prompt requested, as well as adding the ability to see the current total.

Below is the third rendition of the TestDigitExtractor class:

```java
package skibidi;

import java.util.Scanner;

public class TestDigitExtractor {

    //Object that returns user input
    public static int number() {
        @SuppressWarnings("resource")
        //Prepare for user input
        Scanner userInput = new Scanner(System.in);
        //Record user input
        int value = userInput.nextInt();
        //Returns user input
        return value;
    }

    public static String choice() {
        @SuppressWarnings("resource")
        //Prepare for user input
        Scanner userInput = new Scanner(System.in);
        //Record user input
        String choice = userInput.nextLine();
        //Returns user input
        return choice.toLowerCase();
    }

    public static void main(String[] args) {
        //Prompt user for input
        System.out.print("Enter an integer: ");
        int value = number();

        //Declares new variable input
        String input = "";

        //Loops while input is not 0
        while (input != "Q") {

            //Prompt user for input
            System.out.println("Show (W)hole number.");
            System.out.println("Show (O)nes place number.");
            System.out.println("Show (T)ens place number");
            System.out.println("Show (H)undreds place number");
            System.out.println("(Q)uit.");
            System.out.print("Enter your choice: ");

            //Declares input variable as the return of the number() object
            input = choice();

            //Calls Num Object from DigitExtractor Class with the value and string input
            DigitExtractor.Num(value, input);

        }
```

I made the choice() object return a lowercase value using the .toLowerCase(); method. The while loop remains the same though, creating a logic error.

Below is the third rendition of the DigitExtractor class:

```
package skibidi;

public class DigitExtractor {

    public static void Num(int number, String choice) {

        int hundreds = number / 100;

        int tens = number / 10 % 10;

        int ones = number % 10;

        if (choice == "w") {
            System.out.println("The number is: " + number);
        } else if (choice == "h") {
            System.out.println("The hundreds place digit is: " + hundreds);
        } else if (choice == "t") {
            System.out.println("The tens place digit is: " + tens);
        } else if (choice == "o") {
            System.out.println("The ones place digit is: " + ones);
        }

    }

}
```

I made the if else if statement conditions lowercase to work with the aforementioned toLowerCase(); method.

Below is the final rendition of the TestDigitExtractor class:

```java
package skibidi;

import java.util.Scanner;

public class TestDigitExtractor {

    //Object that returns user input
    public static int number() {
        @SuppressWarnings("resource")
        //Prepare for user input
        Scanner userInput = new Scanner(System.in);
        //Record user input
        int value = userInput.nextInt();
        //Returns user input
        return value;
    }

    public static String choice() {
        @SuppressWarnings("resource")
        //Prepare for user input
        Scanner userInput = new Scanner(System.in);
        //Record user input
        String choice = userInput.nextLine();
        //Returns user input
        return choice.toLowerCase();
    }

    public static void main(String[] args) {
        //Prompt user for input
        System.out.print("Enter an integer: ");
        int value = number();

        //Declares new variable input
        String input;
        input = " ";

        //Loops while input is not 0
        while (!input.equals("q")) {

            //Prompt user for input
            System.out.println("Show (W)hole number.");
            System.out.println("Show (O)nes place number.");
            System.out.println("Show (T)ens place number");
            System.out.println("Show (H)undreds place number");
            System.out.println("(Q)uit.");
            System.out.print("Enter your choice: ");

            //Declares input variable as the return of the number() object
            input = choice();

            //Calls Num Object from DigitExtractor Class with the value and string input
            DigitExtractor.Num(value, input);
```

I fixed the while loop in the main method so that it now checks for a lowercase “q”, and uses the.equals() method. This way the condition is able to be checked. Looking back on the code I believe I could have used a do while loop so that I don't need to initialize the input variable before the loop begins.

Below is the final rendition of the DigitExtractor class:

```
package skibidi;

public class DigitExtractor {

    public static void Num(int number, String choice) {

        int hundreds = number / 100;

        int tens = number / 10 % 10;

        int ones = number % 10;

        if (choice.equals("w")) {
            System.out.println("The number is: " + number);
        } else if (choice.equals("h")) {
            System.out.println("The hundreds place digit is: " + hundreds);
        } else if (choice.equals("t")) {
            System.out.println("The tens place digit is: " + tens);
        } else if (choice.equals("o")) {
            System.out.println("The ones place digit is: " + ones);
        }
    }

}
```

I fixed the condition for the if else if statement, utilizing the .equals(); method. Now the condition is able to be met.

Below is the an example of the output of the final rendition without any errors:

```
Enter an integer: 348
Show (W)hole number.
Show (O)nes place number.
Show (T)ens place number
Show (H)undreds place number
(Q)uit.
Enter your choice: w
The number is: 348
Show (W)hole number.
Show (O)nes place number.
Show (T)ens place number
Show (H)undreds place number
(Q)uit.
Enter your choice: o
The ones place digit is: 8
Show (W)hole number.
Show (O)nes place number.
Show (T)ens place number
Show (H)undreds place number
(Q)uit.
Enter your choice: t
The tens place digit is: 4
Show (W)hole number.
Show (O)nes place number.
Show (T)ens place number
Show (H)undreds place number
(Q)uit.
Enter your choice: h
The hundreds place digit is: 3
Show (W)hole number.
Show (O)nes place number.
Show (T)ens place number
Show (H)undreds place number
(Q)uit.
Enter your choice: q
```