AssessedExerciseTemplate-level6-Copy1

November 3, 2020

1 Short Assessed Exercise

2 Level 6

2.1 Ifthy Fairoze

2.2 31 October 2020

2.3 Version 1

2.4 Summary of the Question

Write a program that is to be used to count how many passengers are travelling on buses that pass a particular bus stop in a given hour. It should use a while loop to repeatedly ask the user to give the number of passengers on the bus that just passed. It should stop when the special code X is entered as the number of passengers. It should then give the number of buses and the total number of passengers counted in that hour. For example, one run might be as follows.

```
How many passengers were on the bus? 2
How many passengers were on the bus? 5
How many passengers were on the bus? 10
How many passengers were on the bus? 3
How many passengers were on the bus? 12
How many passengers were on the bus? 1
How many passengers were on the bus? 0
How many passengers were on the bus? X

There were a total of 33 passengers on 7 buses.
```

2.5 The literate program development

Building on the what I have learned in from the ECS401U Jupyter Notebook, to answer the question, I created two methods of inputString and busDetails. For each method, I will explain about the method, what is does and how I tested it and provide the code of the method itself. Next, I show how to run the solution to the question. Each code fragment can be run in Jupyter notebooks, here. Finally, I show the full program that can only be run outside of notebooks, along with instructions on how to do this and a summary of the structure of the full program.

2.5.1 inputString

What it does   This method takes a question (String type) which is printed and waits for the user to give a response to that question (which is also a string type) which is returned.

Implementation (how it works)   In the first line it sets up a local scanner to the method, linked to the keyboard. This makes avaliable the method scanner.nextLine(). The question asking is then printed and then nextLine then allows the user to type a response which is read and stored in the answer variable (which is declared as a string). Then this response in answer is returned as a result of this method.

```
[2]:  /* This method prints out a question to the user and then the user has to give␣
      ↪a response
      typed as the answer which is returned. */
      public static String inputString(String message)
      {
          Scanner scanner = new Scanner(System.in);
          String answer;
          System.out.println(message);
          answer = scanner.nextLine();
          return answer;
      } // END inputString
```

Testing
```
[3]:  System.out.println(inputString("What is the answer to this question?"));
```

```
What is the answer to this question?
I don't know
I don't know
```

2.5.2 busDetails

What it does   This method asks the user repeatedly to give the number of passengers on the bus that just passed. The user should then give a number response. The loop stops when a special character "X" is entered as the number of passengers. It then gives the number of buses and total number of passengers counted.

Implementation (how it works)   In the first line passengers variable (String type) is declared and intialised with letter "A" which will be used later on in the method. Then the variable busCounter1 (integer type) is declared and initialised which will be used as a counter in the while loop after to count how many buses are counted. The variable totalPassengers (also an integer type) is declared and intialised which will keep record of how many passengers accumulate in the loop. Variable busCounter2 is declared to be used further in the code. In the while loop the test is while the passengers value is not equal to the special code (X). So when the question of how many passengers were on the bus is repeated to the user, if their response (which is stored in variable passengers) is X it will stop the loop. The question is repeated in the body of the while loop using inputString to get the user's response and store it in passengers variable. Each time the loop repeats the busCounter1 variable increases by 1 until the passengers value is equal to the special code. This

is done to keep record of the total buses that have passed. Since the passengers variable is a String type, when the user enters a number as a response to the question we have to change the value to an integer. This is done through an if statement when the passengers value (response to the question) is not the special code X. So this becomes the test of the if statement. So if the if statement test is true (response to question is a number and stored in passengers) then totalPassengers gets the copy of the previous totalPassengers value add the number of the passenger variable. The number in the passenger variable is changed from String to integer using Integer.parseInt of passenger variable so we can add to totalPassengers. This is repeated again in the while loop through the if statement which accumlates the number of passengers. This gives a total of the passengers once the while loop stops (when special code is entered). After the loop busCounter2 gets a copy of busCounter1 value subtracted by 1. This is done to not count the last repeat of the loop. Then it prints the total passengers and total buses passed.

```
[4]: /* This method asks the user how many passsengers are on a bus repeatedly where
     ↪they have to give a response
     until they enter a special code (X) then tells the user the total number of
     ↪passengers and total buses. */
     public static void busDetails()
     {
         String passengers = "A";
         int busCounter1 = 0;
         int totalPassengers = 0;
         int busCounter2;
         while (!passengers.equals("X"))
         {
             busCounter1 = busCounter1 + 1;
             passengers = inputString("How many passengers were on the bus?");
             if ((!passengers.equals("X")))
             {
                 totalPassengers = Integer.parseInt(passengers) + totalPassengers;
             }
         }
         busCounter2 = busCounter1 - 1;
         System.out.println("There were a total of " + totalPassengers + "
     ↪passengers on " + busCounter2 + " buses.");
     } // END busDetails
```

Testing

```
[5]: busDetails();
```

```
How many passengers were on the bus?
2
How many passengers were on the bus?
3
How many passengers were on the bus?
6
How many passengers were on the bus?
```

```
7
How many passengers were on the bus?
5
How many passengers were on the bus?
X
There were a total of 23 passengers on 5 buses.
```

2.5.3 Running the program

Run the following call to simulate running the complete program.

```
[6]: busDetails();
```

```
How many passengers were on the bus?
2
How many passengers were on the bus?
12
How many passengers were on the bus?
5
How many passengers were on the bus?
6
How many passengers were on the bus?
7
How many passengers were on the bus?
10
How many passengers were on the bus?
6
How many passengers were on the bus?
3
How many passengers were on the bus?
X
There were a total of 51 passengers on 8 buses.
```

2.6 The complete program

This version will only compile here. To run it copy it into a file called initials.java on your local computer and compile and run it there.

```java
[ ]: // NAME: Ifthy Fairoze
     // DATE: 31/10/2020
     // VERSION: 1
     /* BRIEF OVERVIEW OF PURPOSE: Asks the user repeatedly how many passengers on a␣
      ↪bus that just passed. Which stops
     when the a special code is entered by the user as a response. Then prints out␣
      ↪total passengers and buses counted
     in that hour. */

     import java.util.Scanner; // Needed to make Scanner available
```

```java
class BusPassengers
{
    public static void main (String [] a)
    {
        busDetails();
        System.exit(0);
    }

    /* This method prints out a question to the user and then the user has to
↪give a response
    typed as the answer which is returned. */
    public static String inputString(String message)
    {
        Scanner scanner = new Scanner(System.in);
        String answer;
        System.out.println(message);
        answer = scanner.nextLine();
        return answer;
    } // END inputString

    /* This method asks the user how many passsengers are on a bus repeatedly
↪where they have to give a response
    until they enter a special code (X) then tells the user the total number of
↪passengers and total buses. */
    public static void busDetails()
    {
        String passengers = "A";
        int busCounter1 = 0;
        int totalPassengers = 0;
        int busCounter2;
        while (!passengers.equals("X"))
        {
            busCounter1 = busCounter1 + 1;
            passengers = inputString("How many passengers were on the bus?");
            if ((!passengers.equals("X")))
            {
                totalPassengers = Integer.parseInt(passengers) +
↪totalPassengers;
            }
        }
        busCounter2 = busCounter1 - 1;
        System.out.println("There were a total of " + totalPassengers + "
↪passengers on " + busCounter2 + " buses.");
    } // END busDetails
}
```

END OF LITERATE DOCUMENT