Java Lab

Bonus Lab

IMPORTANT! Save all your work to a safe location such as oneDrive.

Create a folder for SDPD into which you will save all your work for this module, arranged how you wish. Ideally you should create a folder each week for your lab exercises. Note that you should create **a separate file** for each exercise.

Exercise 1

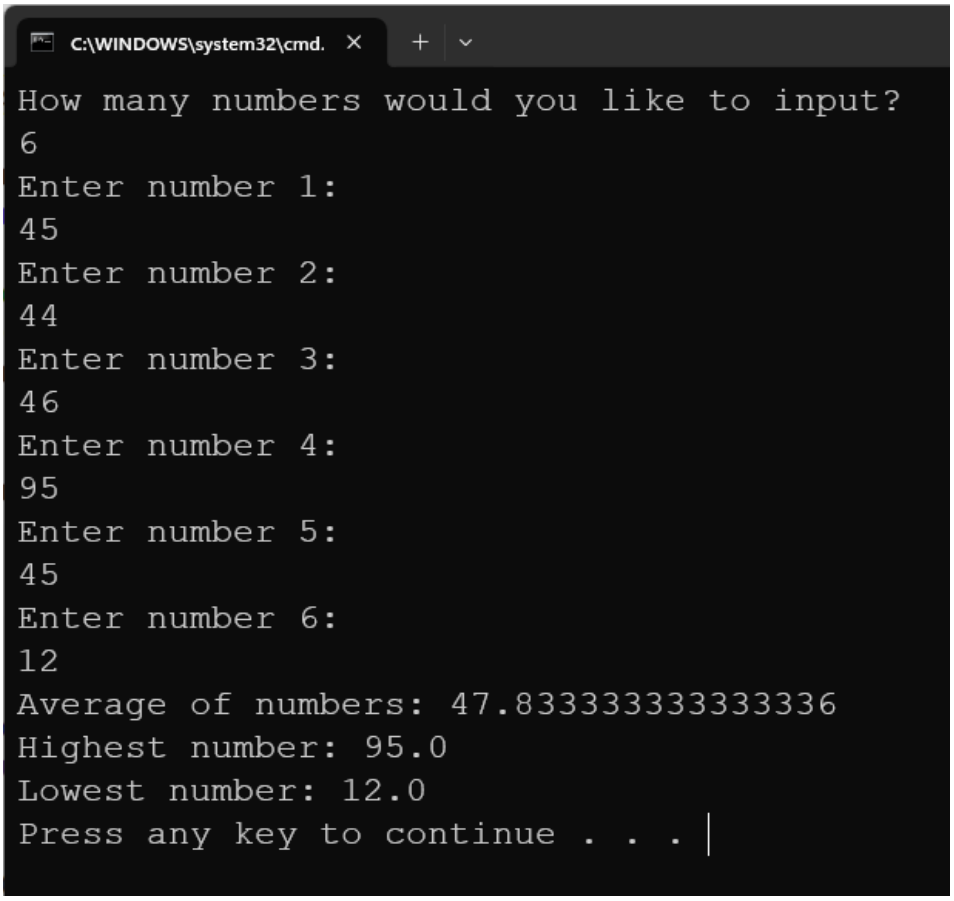
Write a Java program that performs the following tasks:

Prompt the user to specify how many numbers they intend to input. Based on the user's response, allow the user to input that many numbers.

When finished, the program should display the average of all numbers entered, the highest number, and the lowest number to the user.

Constraints:

If the user specifies a non-positive number for the number of inputs, the program should display an appropriate message and terminate without further input.



Exercise 2

Design and implement a text-based ATM simulator in Java that performs the following tasks:

1. PIN Authentication:
    - Before accessing the ATM menu, prompt the user to enter a PIN.
    - The correct PIN is 2310.
    - The user should have only 3 attempts to enter the correct PIN.
    - If the user fails to enter the correct PIN in 3 attempts, display the message "Too many incorrect attempts. Please contact your bank for assistance." and terminate the program.

2. If the correct pin is entered, present the user with an ATM menu that provides the following options:

    - Check Balance
    - Deposit Funds
    - Withdraw Funds
    - Exit

3. Allow the user to check their balance. The initial balance should be set to $100.

4. Allow the user to deposit funds into their account. The deposit amount should be positive. Display an error if the user tries to deposit a negative amount or zero.

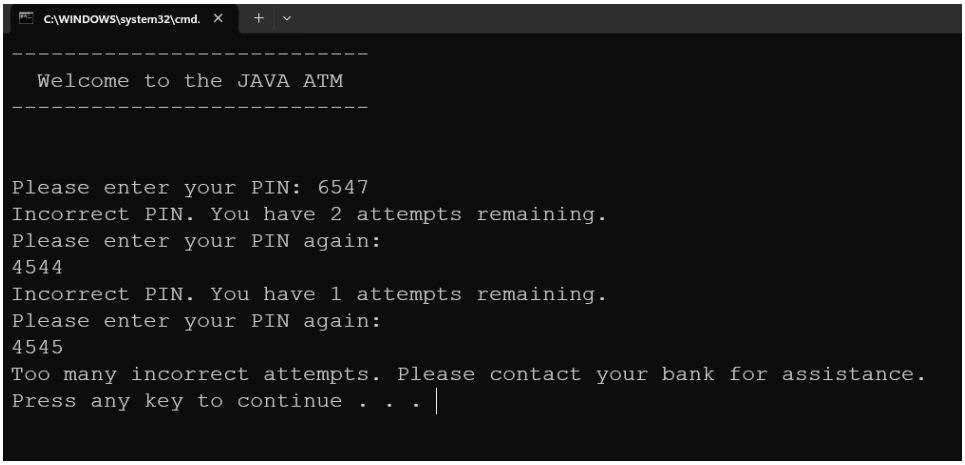
5. Allow the user to withdraw funds from their account with the following conditions:

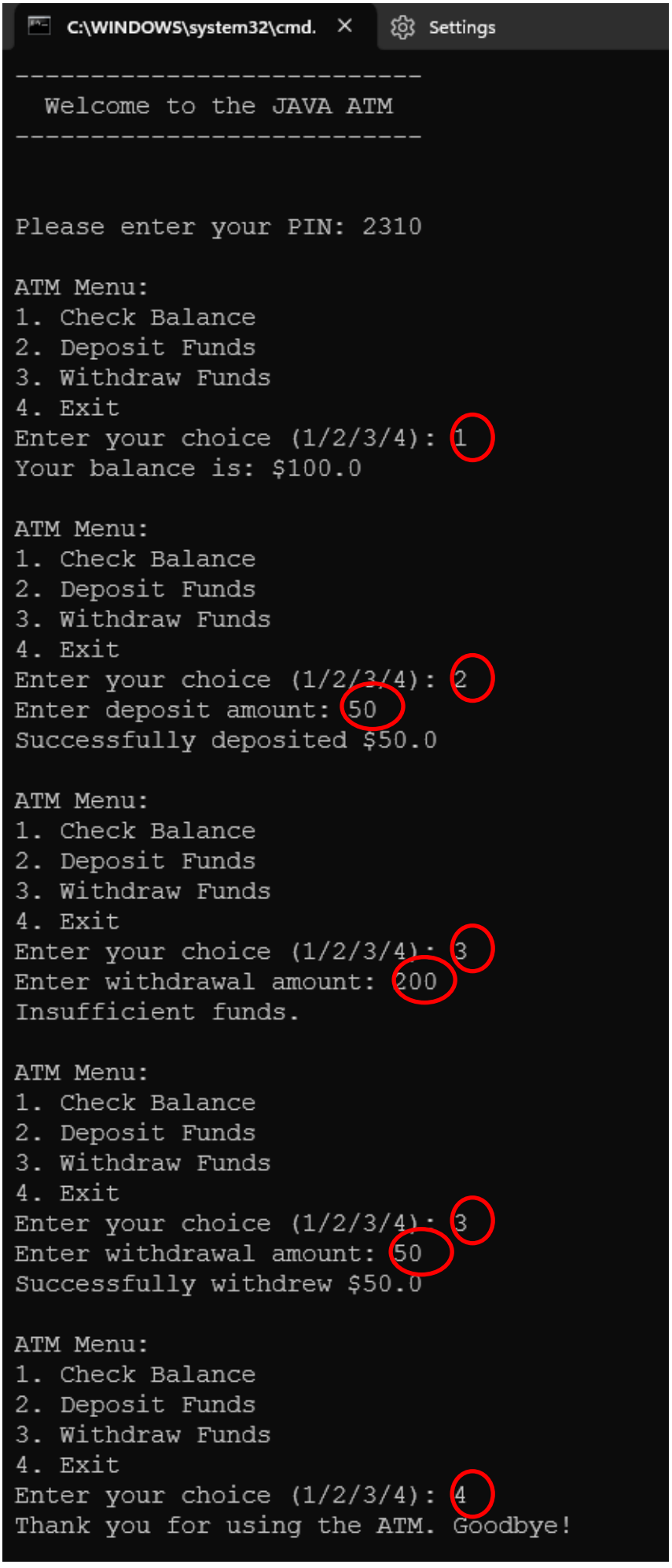
    - The withdrawal amount should be positive. Display an error if the user tries to withdraw a negative amount or zero.
    - The user cannot withdraw more than their current balance. Display an error if the user tries to overdraw.

6. Allow the user to exit the ATM, displaying a goodbye message.

Your output should be similar to as shown below:

Incorrect pins:




C:\WINDOWS\system32\cmd.
Settings

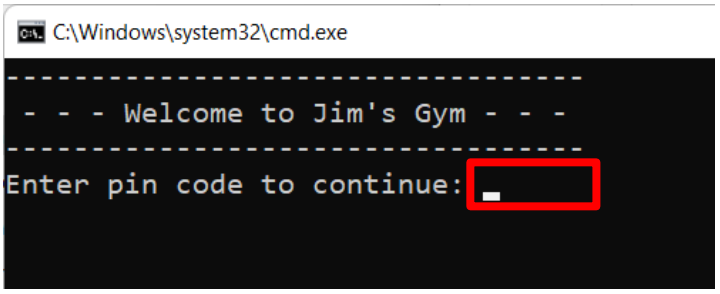
Welcome to the JAVA ATM

Please enter your PIN: 2310
ATM Menu:
1. Check Balance
2. Deposit Funds
3. Withdraw Funds
4. Exit
Enter your choice (1/2/3/4): 1
Your balance is: $100.0
ATM Menu:
1. Check Balance
2. Deposit Funds
3. Withdraw Funds
4. Exit
Enter your choice (1/2/3/4): 2
Enter deposit amount: 50
Successfully deposited $50.0
ATM Menu:
1. Check Balance
2. Deposit Funds
3. Withdraw Funds
4. Exit
Enter your choice (1/2/3/4): 3
Enter withdrawal amount: 200
Insufficient funds.
ATM Menu:
1. Check Balance
2. Deposit Funds
3. Withdraw Funds
4. Exit
Enter your choice (1/2/3/4): 3
Enter withdrawal amount: 50
Successfully withdrew $50.0
ATM Menu:
1. Check Balance
2. Deposit Funds
3. Withdraw Funds
4. Exit
Enter your choice (1/2/3/4): 4
Thank you for using the ATM. Goodbye!

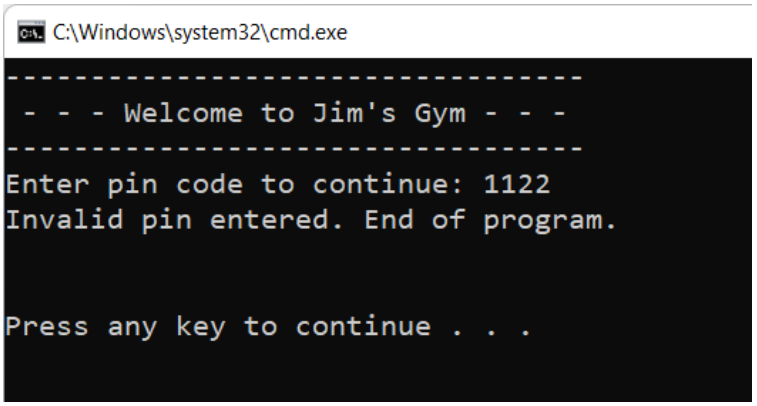
Exercise 3

Write a program in java that allows booking for a set day of the week, for 3 gym machines available for a local gym. The program should initially prompt to enter a pin number. There are 2 users on the system: Alice and Bob. The pin number for Alice is 1234, and the pin number for Bob 5678. These are the only 2 valid pin numbers of the system. If the user enters 1234, then the message “Welcome Alice” is displayed. If the user enters 5678, then the message “Welcome Bob” is displayed. Otherwise, the message “Invalid pin entered. End of program” is displayed and the program terminates.

Step 1: Prompt user for pin number

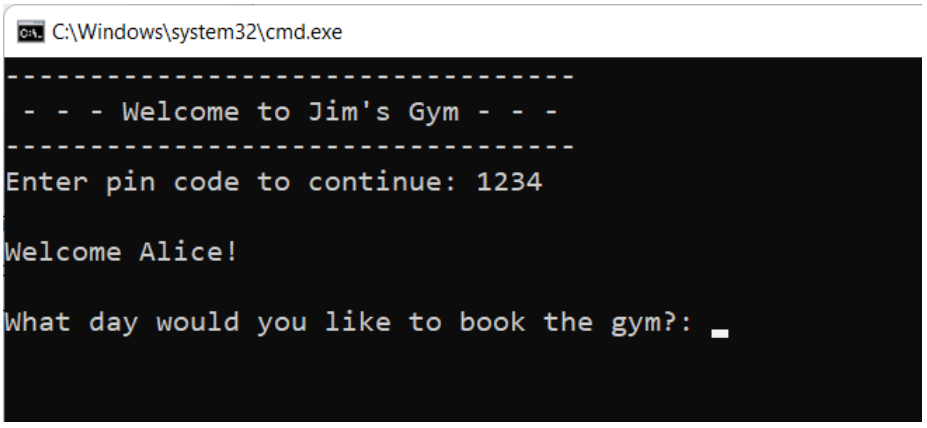


If an invalid pin number is entered, program ends



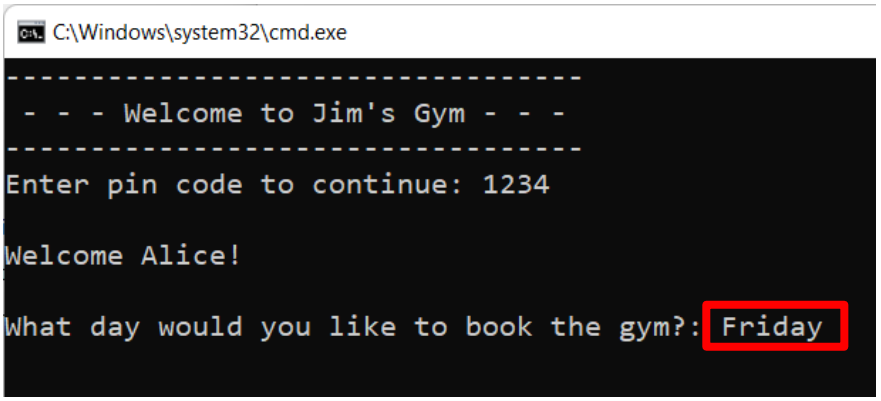
Step 2: The only valid pin numbers are 1234 and 5678

If pin is 1234, message is “Welcome Alice!”, if pin is 5678, message is “Welcome Bob!”



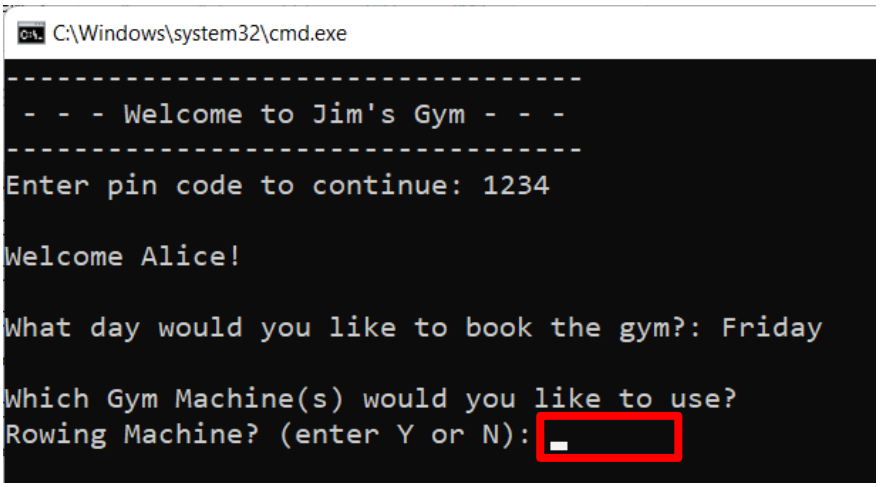
The user is then prompted to type in a day they wish to book, as shown below:

User is prompted to enter day they would like to book:

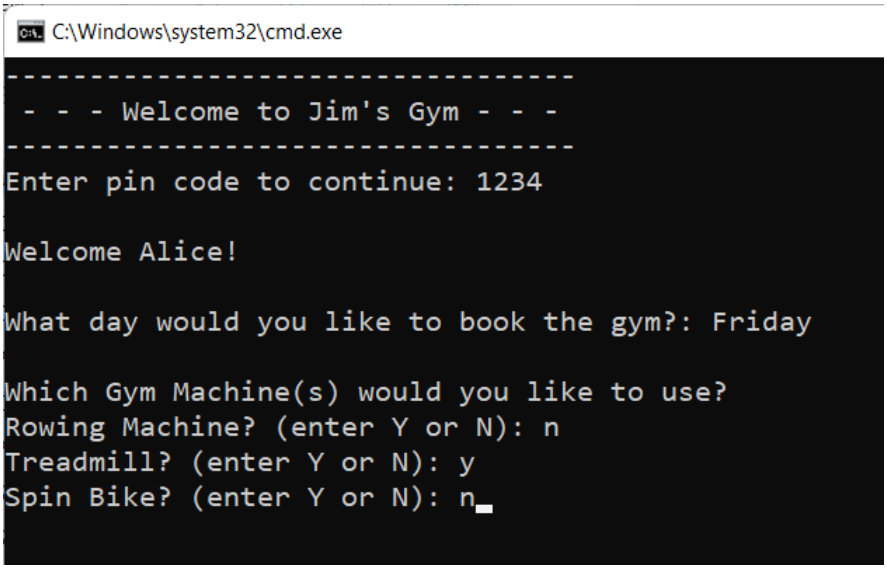


The user is then prompted to type the character Y or N (program should accept uppercase or lowercase) to specify *yes* or *no* to booking the 3 machines available, as shown below:

User is prompted to specify what gym machines to use (enter Y or y for yes or N or n for No).

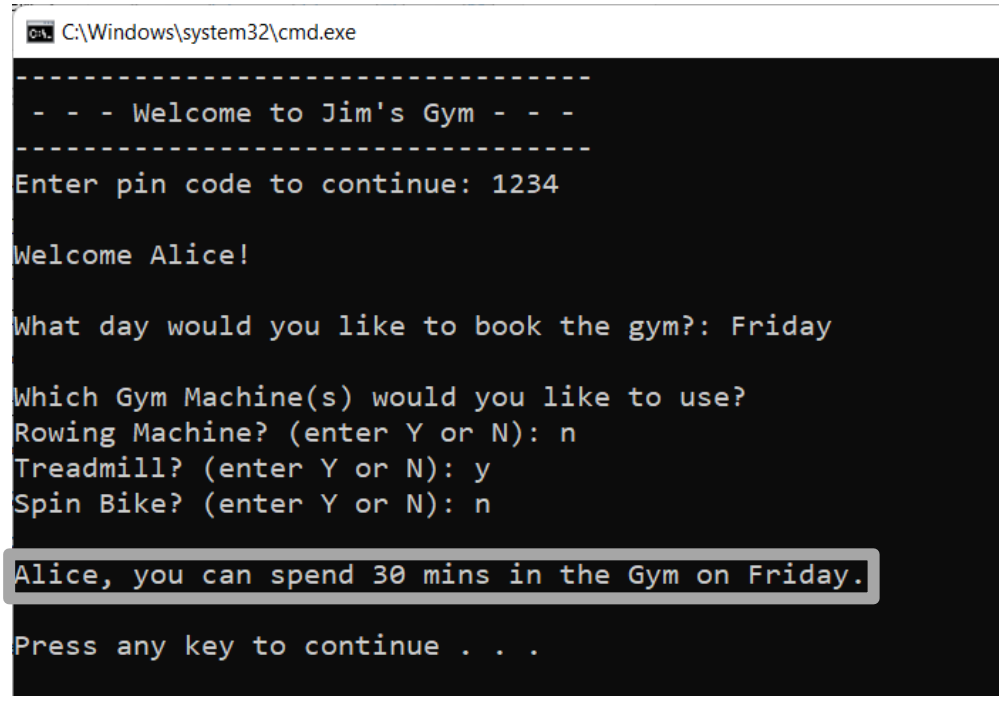


User is prompted to specify Y or N for the three available machines:

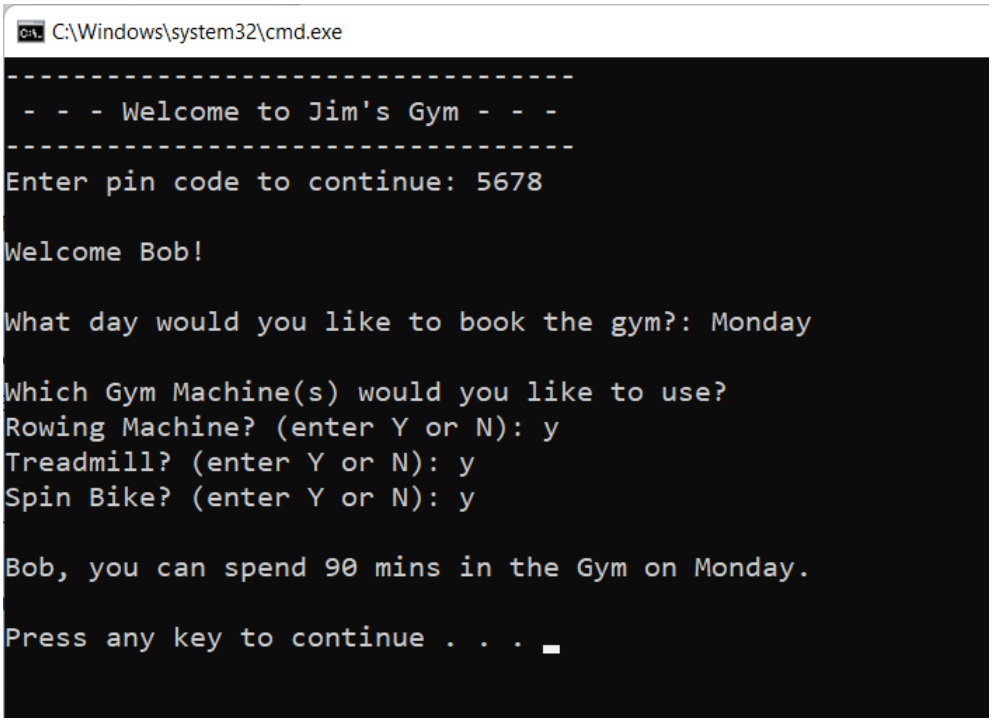


For each machine that is booked, 30 mins is scheduled. Therefore, if one machine is booked, then time allocated will be 30 mins, if 2 machines are booked, time allocated is 60 mins and if 3 machines are booked, 90 mins. This is displayed in a message at the end of the program, along with the name of the user, as shown below:

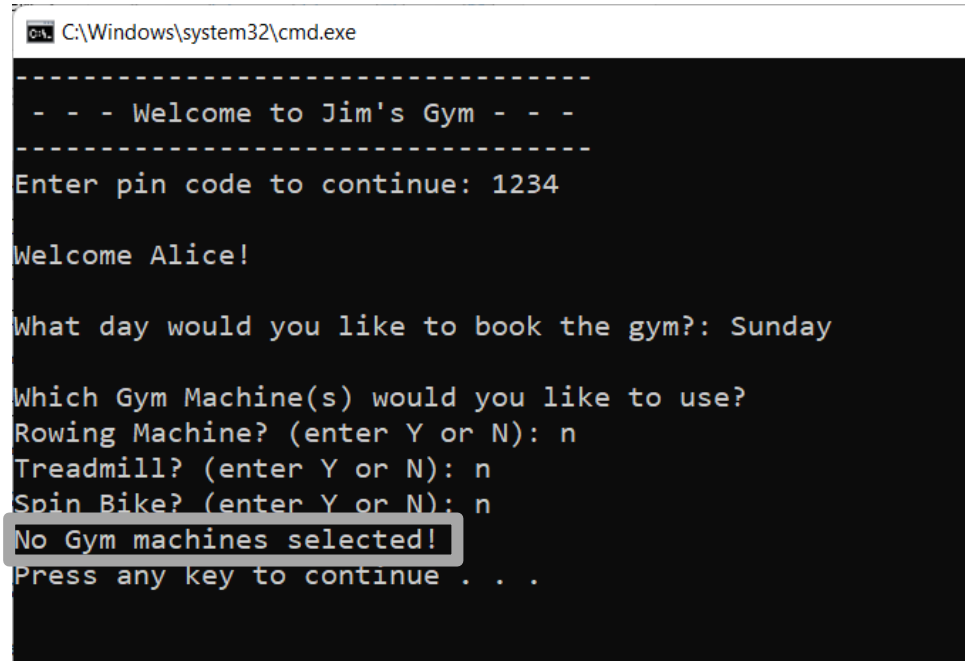
Final message is displayed, as shown below:



Other sample output (with pin 5678), and 3 machines booked:



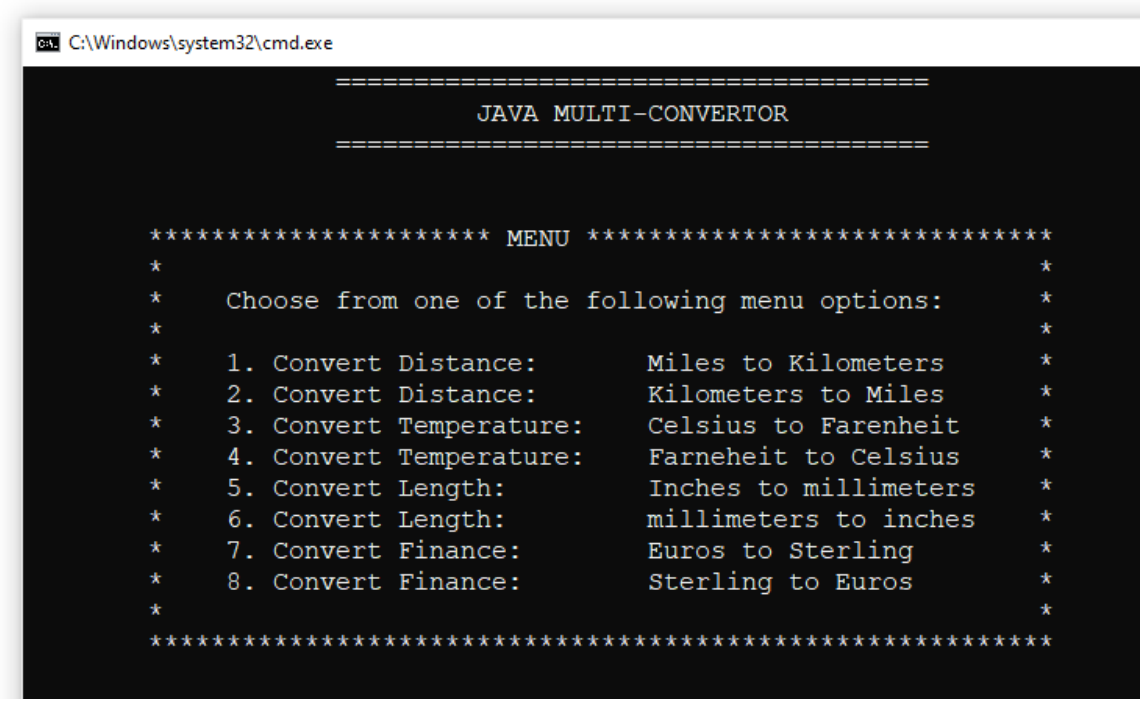
Other sample output (with pin 1234), and no machines booked, message output is "No Gym machines selected":



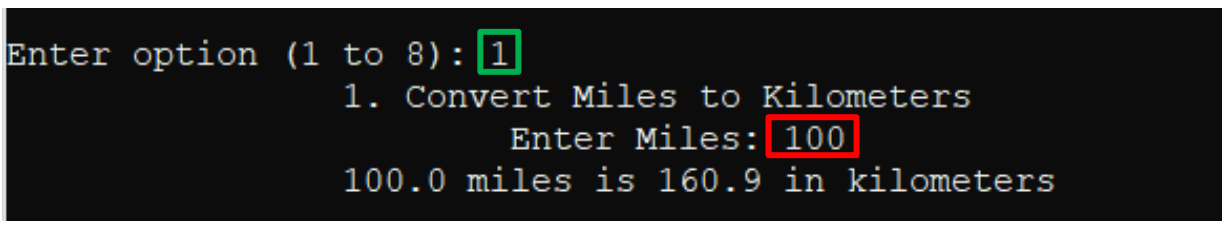
Exercise 4

You are required to create a conversion program that performs a variety of conversions based on user input

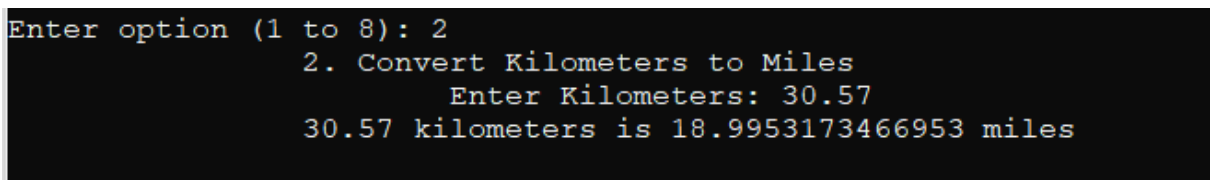
1. When the program starts, it should output to the screen the title “Java multi-Convertor”, followed by a menu box that contains 8 items. Your output should be similar to as shown below:



2. The program should then prompt the user to select from the eight options shown in the menu by entering a number (using scanner) between one and eight (see in green, below). Entering different options will allow the user to perform different calculations. For example, if the user enters option “1”, it will produce the display as shown below (using print or println), allowing the user to enter a value (outlined in red) and the conversion is shown after entering the value:



Sample output on option 2 (using print or println):



Sample output on option 3 (using print or println):

```
Enter option (1 to 8): 3
                3. Convert Celsius to Farenheit
                        Enter temperature in Celsius: 100
                100.0 degrees celsius is 212.0 degrees farenheit
```

Sample output on option 4 (using print or println):

```
Enter option (1 to 8): 4
                4. Convert Farenheit to Celsius
                        Enter temperature in Farenheit: 21.2
                21.2 degrees farenheit is -6.0 degrees celsius
```

Sample output on option 5 (using printf):

```
Enter option (1 to 8): 5
                5. Convert inches to millimeters
                        Enter measurement in inches: 10
                10.0 inches is 254.0 in millimeters
```

Sample output on option 6 (using printf):

```
Enter option (1 to 8): 6
                6. Convert millimeters to inches
                        Enter measurement in millimeters: 100
                100.0 millimeters is 3.9 in inches
```

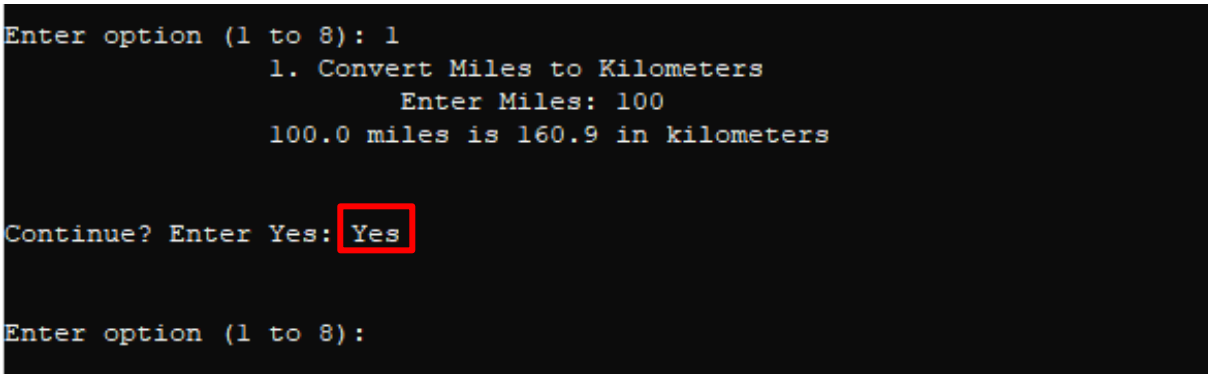
Sample output on option 7 (using printf):

```
Enter option (1 to 8): 7
                7. Convert Euros to Sterling
                        Enter value in Euros: 100
                100.00 Euros is 89.72 in Sterling
```

Sample output on option 8 (using printf):

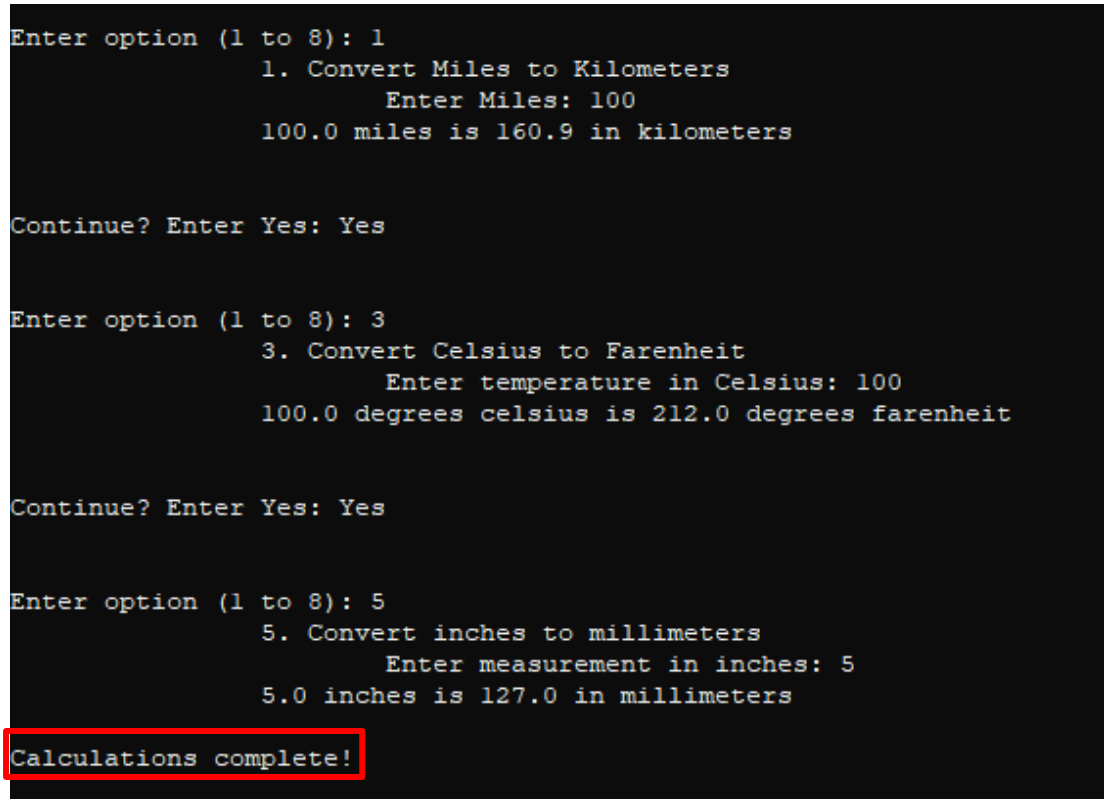
```
Enter option (1 to 8): 8
                8. Convert Sterling to Euros
                        Enter value in Sterling: 55
                55.00 Sterling is 61.30 in Euros
```

3. After a calculation is performed, the user is prompted on whether they would like to perform another calculation by entering the word Yes:

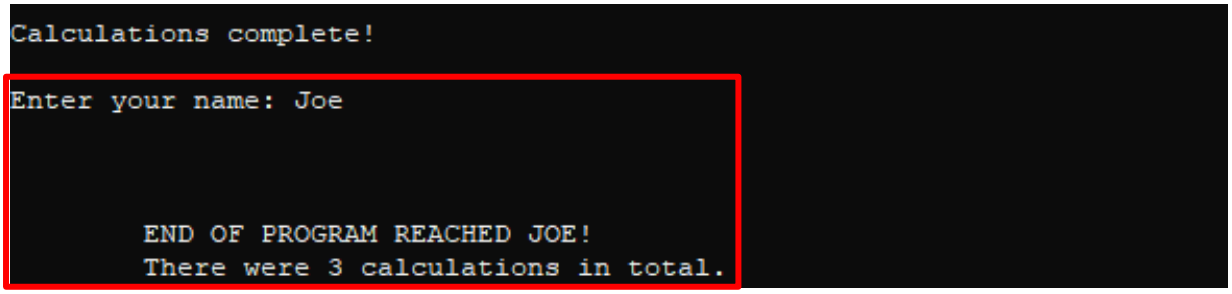


Note that the user should be able to continue regardless of whether the input “yes” is uppercase or lowercase. If any other value is entered the program terminates.

The program should only perform a maximum of three calculations. After the third calculation, the program should terminate.

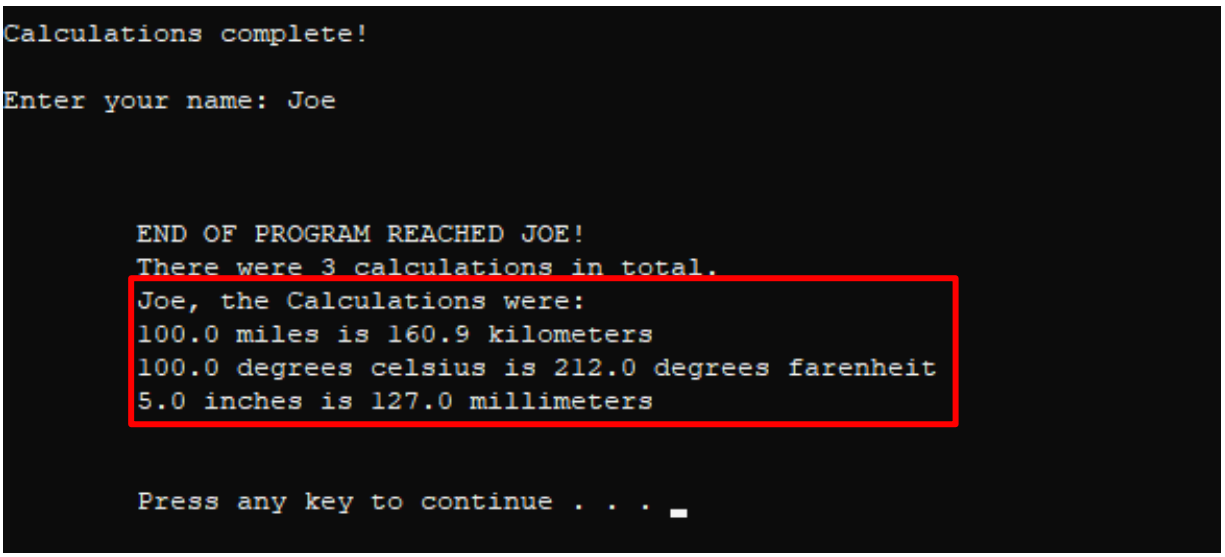


4. The user should then be prompted to enter their name:



The name should appear in a message, for example - “END OF PROGRAM REACHED JOE” that includes the name entered, in ***uppercase*** (even if name is entered in lowercase). There should also be a message stating how many calculations were run (for example, “There were 3 calculations in total”), as shown above.

5. Finally, a message should be displayed stating what calculations were performed, as shown below, eg:



Performing Conversions

The conversions can be performed using the following formulas:

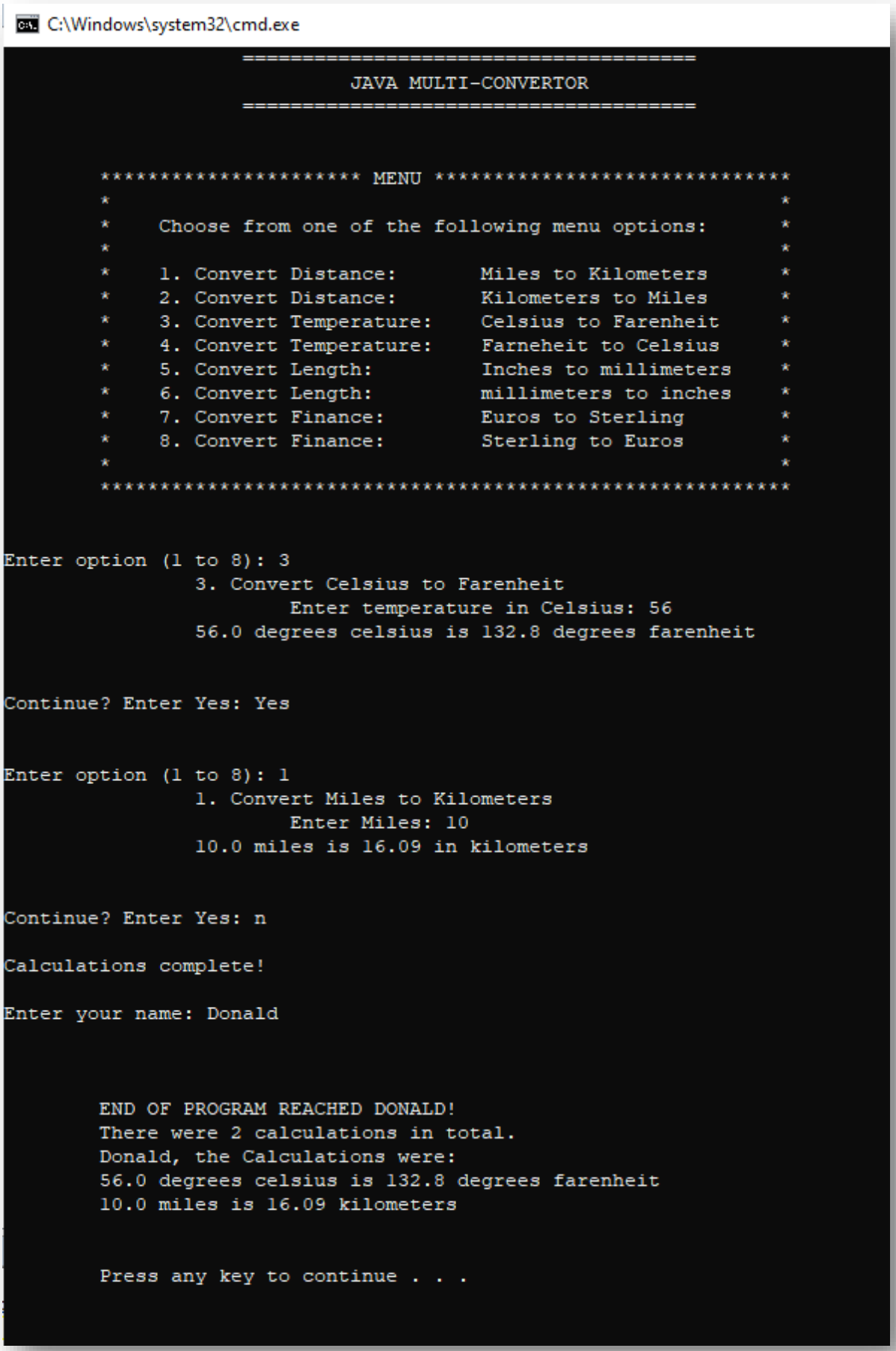
- To convert miles to kilometers: multiply by 1.609
- To convert kilometers to miles: divide by 1.609
- To convert Celsius to Fahrenheit: (Celsius x 1.8)+32
- To convert Fahrenheit to Celsius: (Farenheit-32)/1.8
- To convert inches to millimeters: multiply by 25.4
- To convert millimeters to inches: divide by 25.4
- To convert Euros to Sterling: multiply by 0.897234
- To convert Sterling to euros: divide by 0.897234

SAMPLE OUTPUT

Your program should match the following sample output from the program:

SAMPLE OUTPUT (with only 2 calculations performed)

Your program should match the following sample output from the program:



- Incorrect menu option entered (eg., 9): Note that input of an incorrect option does not count as a calculation.