Joe Del Rocco
jdelrocco [at] stetson [dot] edu
Assistant Professor of Practice, Computer Science
Stetson University
421 N Woodland Blvd, DeLand, FL, 32723
www.stetson.edu

CSCI 141
(all sections)
Fall 2021
Assignment

Assignment 3

Due: Monday 10/11/2021 11:59pm

Contents

Program

This program is meant to give you practice with indexing arrays and using a definite (for) loop to iterate through them. You will ask the user how many sales they want to "load" (randomly generate), then you will create an array of doubles of that size and fill it with random fake sales. Then you will write two functions, one to print out the sales figures, and one to print out metrics. See the Example Output.

Here is the link to the GitHub Classroom assignment:
https://classroom.github.com/a/K0D2Ep8Y

(a) Generate sales

In real world point-of-sales programs used in retail stores, restaurants, and shops, sales figures are collected per transaction and logged to a database or file system (local or on the cloud). But that is all beyond the scope of this assignment. We will simply generate some random sales.

First ask the user how many sales they want to "load." Then create an array of type `double` of that size. In other words, if the user wants 10 sales, create a double array of size 10. Then use a definite (for) loop to iterate through the array of sales and assign each element to a random decimal number between 0 and 100.

Tip

You can generate a random `double` between `0.0` and `100.0` by calling your `Random` object's `.nextDouble()` method and multiplying the returned `double` by `100.0`.

(b) printSales()

Write a function with exactly this prototype:

```
private static void printSales(double[] stuff)
```

Inside the function, iterate through the array `stuff` and print each value out on its own line.

Tip

If you want to print out a double rounded to 2 decimal places, you can use the `String.format()` function. Here is an example of a double rounded to 2 decimal places, right-aligned, and filling at least 7 digits of whitespace:

```
String str = String.format("%7.2f", value); // value is a double
System.out.println(str);
```

(c) computeMetrics()

Write a function with exactly this prototype:

```
private static void computeMetrics(double[] stuff)
```

This function will also iterate through the `stuff` array, but instead of printing each double out, it will compute some metrics. Compute the total amount of all doubles by adding them up. Find the highest and lowest double in the array. Compute the mean (average) of all the doubles. Finally, print all of these metrics out.

(d) Call your functions from main()

After you have finished your functions, make sure to call them from `main()` passing your array of sales to each function. You can call your functions after you create and fill your array with random sales figures. You can also print separators between the function calls to separate the metrics from the sales figures.

Submission

You will commit and push your changes to your specific GitHub Classroom repository for this assignment. Please follow the directions in this assignment to install all software. Commit and push your code changes early and often any time before the due date. Please see the advice below; it is important for grading purposes. **Failure to follow these directions will result in a loss of points.**

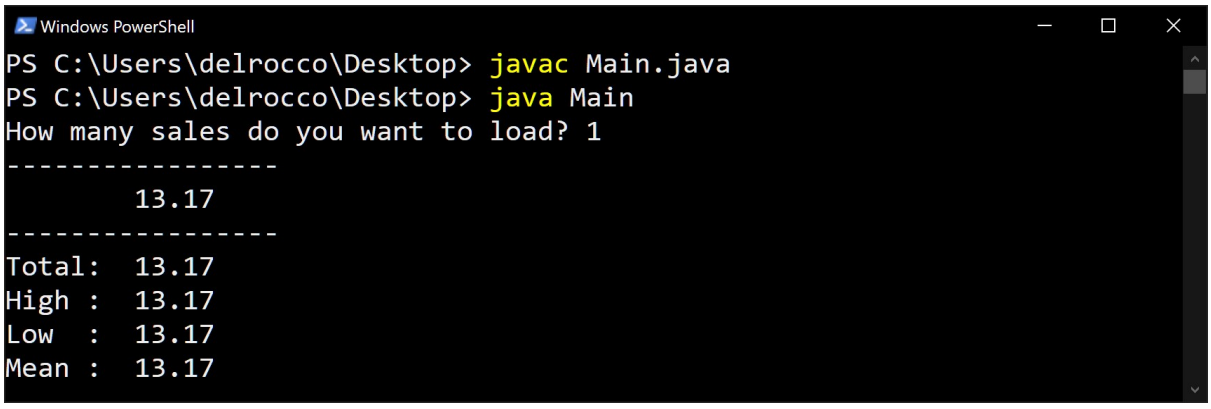
Always make sure to:

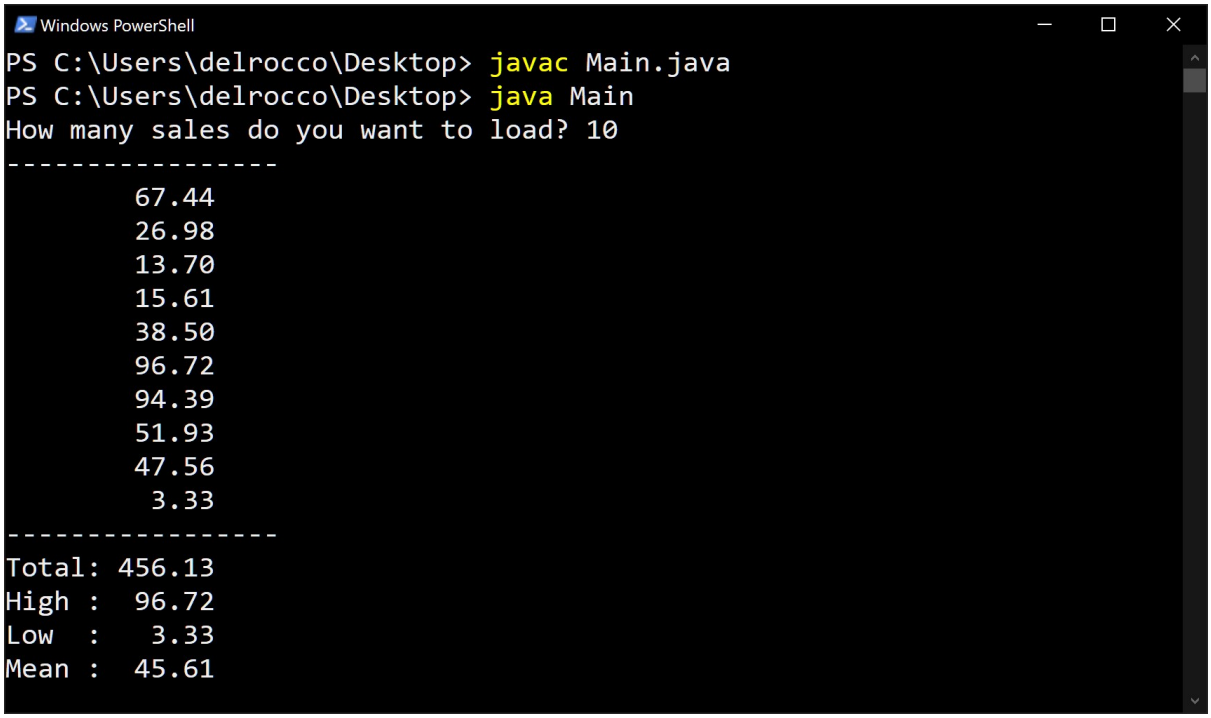
- Keep all source files in the folder called `src`, which is one directory in from the root of your repo
- Do not commit multiple copies of the same named source file; modify the ones provided to you. In other words, do not make an old and new version of the same file
- The main starting source file should always be called `Main`
- When loading resources, do not use absolute paths to files on your drive; use relative paths
- Do not have the keyword `package` at the top of any files. Some IDEs add your files to a custom package by default. Please remove this line, as it complicates grading.

Rubric

Task	Percentage
Assignment files submitted to Canvas instead of GitHub	Grade is 0%
General attempt at the assignment	50%
Properly creating array of correct size	10%
Properly calling functions from `main()`	10%
`printSales()`	10%
`computeMetrics()`	20%
Total	100%

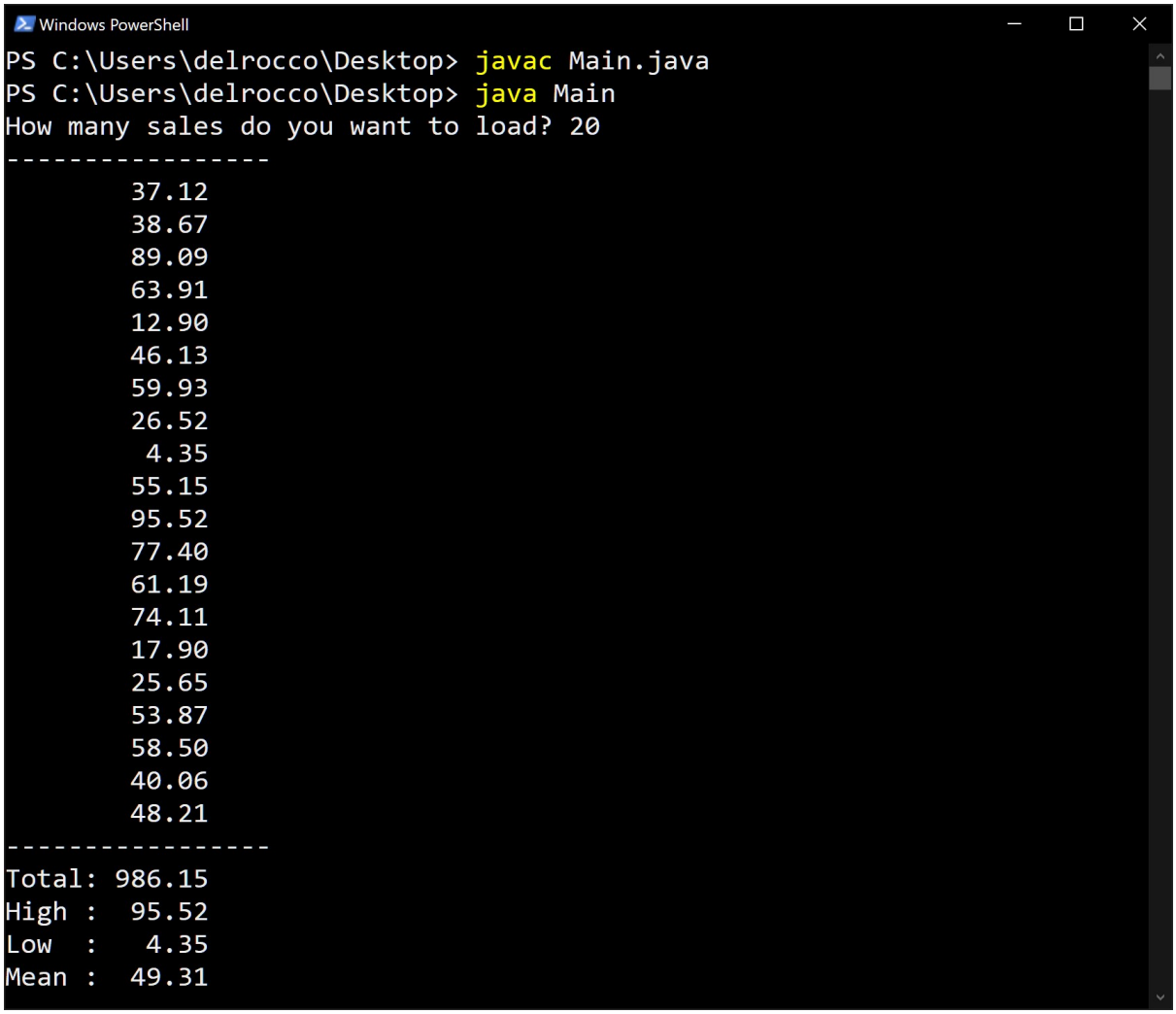
Example Output




Windows PowerShell
PS C:\Users\delrocco\Desktop> javac Main.java
PS C:\Users\delrocco\Desktop> java Main
How many sales do you want to load? 10

67.44
26.98
13.70
15.61
38.50
96.72
94.39
51.93
47.56
3.33

Total: 456.13
High : 96.72
Low : 3.33
Mean : 45.61


Windows PowerShell
PS C:\Users\delrocco\Desktop> javac Main.java
PS C:\Users\delrocco\Desktop> java Main
How many sales do you want to load? 20

37.12
38.67
89.09
63.91
12.90
46.13
59.93
26.52
4.35
55.15
95.52
77.40
61.19
74.11
17.90
25.65
53.87
58.50
40.06
48.21

Total: 986.15
High : 95.52
Low : 4.35
Mean : 49.31