Jack Utzerath
CST-105 9am
Exercise 4
10/17/21

Loom Video :
https://www.loom.com/share/d9838925110f441abd218c7ad24327bb

Text From Program:

```
/*
 * Jack Utzerath
 * CST-105 (9am)
 * Exercise 4
 * 10/17/21
 *
*/

package nerd;

//Import java pre made classes
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.PrintWriter;
import java.math.BigDecimal;
import java.util.Random;
import java.util.Scanner;

public class Exercise4 {

	public static void main(String[] args) {

		//create random class
		Random rnd = new Random();
		try
		{
			//create the input file
			FileOutputStream fos = new FileOutputStream("input.in");
			//create instance of the printwriter class
```

```
		PrintWriter write = new PrintWriter(fos);

		//Write Random numbers to a file
		for (int i = 0; i < 1000; i++)
		{
			double randomValue = 0 + (100000-0) *rnd.nextDouble();

			write.printf("%6f  ",randomValue );
		}
		//Close file
		write.close();
}
catch (Exception exp)
{
		System.out.println("There was an Exception" + exp);
}


//-------------------------------------------------------------------


try

{
		// Create an instance of the text file
		File inputFile = new File("input.in");
		// Make scanner read from file
		Scanner myReader = new Scanner(inputFile);
		//create result file
		FileOutputStream fos2 = new FileOutputStream("results.out");
		//create instance of the print file
		PrintWriter write2 = new PrintWriter(fos2);


		//Initialize the variables
		double maximum = 0, minimum = 0;
		double lastDonation = 0, lastDonation2 = 10000000;
		int numberOfDonations = 0;
		double limit = 10000000;
		double total = 0;
		double average = 0;
		boolean finished = false;

		//create while loop
```

```
                while (myReader.hasNextDouble() && !finished) {
                        // Read the line from the file

                        numberOfDonations++;

                        double currentDonation = myReader.nextDouble();

                        total = total + currentDonation;

                        average = (average + currentDonation)/2;

                        if (currentDonation > lastDonation)
                        {
                                lastDonation = currentDonation;
                                maximum = currentDonation;
                        }

                        if (currentDonation < lastDonation2)
                        {
                                lastDonation2 = currentDonation;
                                minimum = currentDonation;
                        }


                        if (total > limit)
                        {
                                finished = true;
                        }


                }

                //Convert values to BigDecimal
                BigDecimal maximum2 = new BigDecimal(maximum);
                BigDecimal minimum2 = new BigDecimal(minimum);
                BigDecimal average2 = new BigDecimal(average);
                BigDecimal total2 = new BigDecimal(total);

                //Write the sentences in text file
                write2.printf("It took %d contribution to reach the goal of $10,000,000\n",
numberOfDonations);
                write2.printf("The maximum contribution received was $%,f\n",
maximum2);
                write2.printf("The minimum contribution received was $%,f\n", minimum2);
```

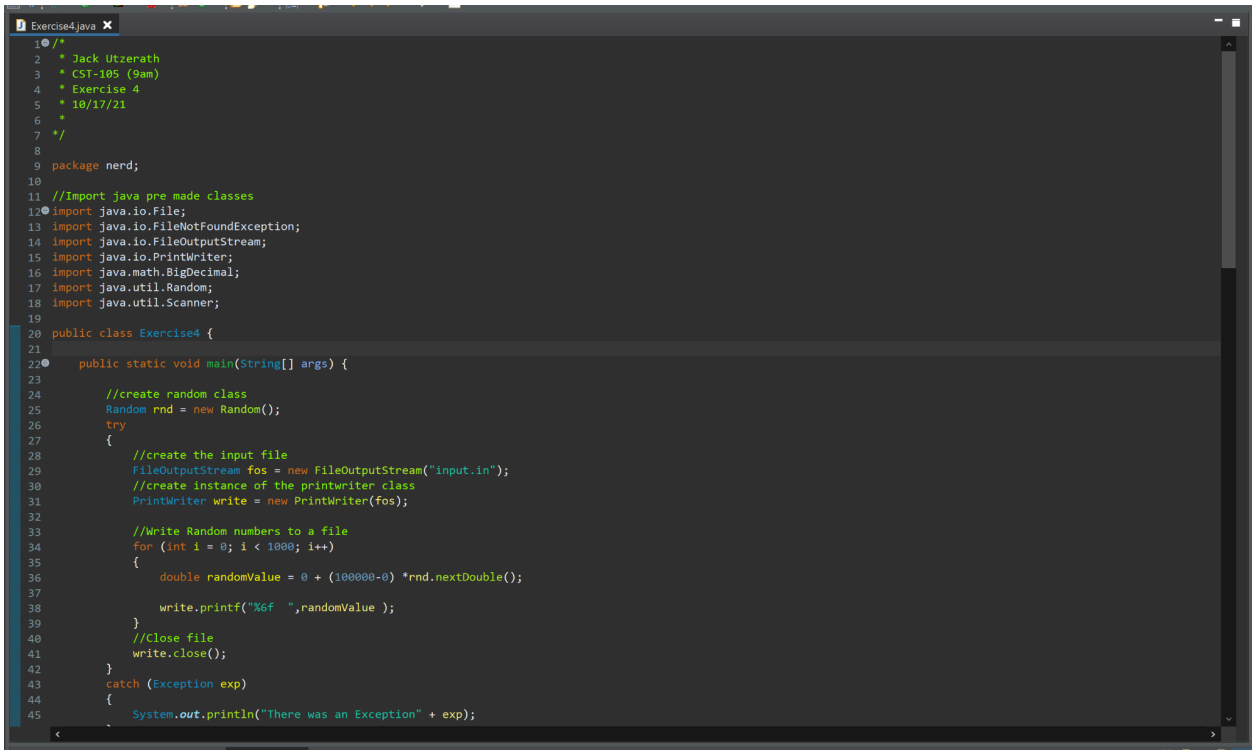
```
                write2.printf("The average contribution was $%,f\n", average2);
                write2.printf("A total of $%,f was collected", total2);


                //Close the reader and writer
                myReader.close();
                write2.close();
            }
            catch (FileNotFoundException e) // exception raised when there is an error
reading file
            {
                System.out.println("An Error Occured");

                e.printStackTrace();
            }


        }

}
```



Exercise4.java

```
        }
        catch (Exception exp)
        {
            System.out.println("There was an Exception" + exp);
        }


        //-----------------------------------------------------------------


        try

        {
            // Create an instance of the text file
            File inputFile = new File("input.in");
            // Make scanner read from file
            Scanner myReader = new Scanner(inputFile);
            //create result file
            FileOutputStream fos2 = new FileOutputStream("results.out");
            //create instance of the print file
            PrintWriter write2 = new PrintWriter(fos2);


            //Initialize the variables
            double maximum = 0, minimum = 0;
            double lastDonation = 0, lastDonation2 = 10000000;
            int numberOfDonations = 0;
            double limit = 10000000;
            double total = 0;
            double average = 0;
            boolean finished = false;

            //create while loop
            while (myReader.hasNextDouble() && !finished) {
                // Read the line from the file

                numberOfDonations++;

                double currentDonation = myReader.nextDouble();

                total = total + currentDonation;

                average = (average + currentDonation)/2;

                if (currentDonation > lastDonation)
                {
```

```
            }

            //Convert values to BigDecimal
            BigDecimal maximum2 = new BigDecimal(maximum);
            BigDecimal minimum2 = new BigDecimal(minimum);
            BigDecimal average2 = new BigDecimal(average);
            BigDecimal total2 = new BigDecimal(total);

            //Write the sentences in text file
            write2.printf("It took %d contribution to reach the goal of $10,000,000\n", numberOfDonations);
            write2.printf("The maximum contribution recieved was $%,f\n", maximum2);
            write2.printf("The minimum contribution recieved was $%,f\n", minimum2);
            write2.printf("The average contribution was $%,f\n", average2);
            write2.printf("A total of $%,f was collected", total2);


            //Close the reader and writer
            myReader.close();
            write2.close();
        }
        catch (FileNotFoundException e) // exception raised when there is an error reading file
        {
            System.out.println("An Error Occured");
            // The error is stored in e (printStackTrace() is a method).
            // printStaceTrace will pinpoint the exact line in which the method raised the
            // exception
            e.printStackTrace();
        }


    }
}
```

Problems Javadoc Declaration Console Debug

<terminated> Exercise4 [Java Application] C:\Users\jacku\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_16.0.2.v20210721-1149\jre\bin\javaw.exe (Oct 17, 2021, 8:15:47 PM – 8:15:47 PM)