# Hands-on Experiment # 7 : Worksheet

Section\_\_\_\_\_\_2\_\_\_\_\_\_\_\_ Date\_\_\_\_\_\_15/3/2018\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

No more than 3 students per one submission of this worksheet.

Student ID \_\_\_\_\_\_\_6031848721\_\_\_\_\_\_ Name\_\_\_\_\_Watcharin Kriengwatana\_\_\_\_\_\_\_\_\_\_\_\_\_

Student ID \_\_\_\_\_6031847021\_\_\_\_\_\_\_\_\_\_\_ Name\_\_\_\_\_\_\_\_\_Wasuthon Klyhirun\_\_\_\_\_\_\_\_\_\_

Student ID \_\_\_\_\_6031851521\_\_\_\_\_\_\_\_\_\_\_ Name\_\_\_\_\_\_\_\_Sarun Nuntaviriyakul\_\_\_\_\_\_\_\_\_\_\_

## Part A: Loop Writing Practice

In *MathPowLoop.java*, write Java statements using “loops” to calculate result2 so that its value is similar to result1 (which is calculated from *Math.pow()* ) for every double a and int b.

No methods in the *Math* class is allowed.

List your code here.

import java.util.Scanner;

public class MathPowLoop{

public static void main(String [] args){

Scanner sc = new Scanner(System.in);

System.out.print("a=");

double a = sc.nextDouble();

System.out.print("b=");

int b = sc.nextInt();

double result1 = Math.pow(a,b);

double result2=1;

if (b<0)

{

b=-b;

for (int i=0;i<b;i++)

{

result2=result2\*a;

}

result2=1/result2;

}

else

{

for (int i=0;i<b;i++)

{

result2=result2\*a;

}

}

System.out.println("Math.pow("+a+","+b+") = "+result1);

System.out.println("Your loop a^b = "+result2);

}

}

Test your code with the following test data set.

|  |  |  |  |
| --- | --- | --- | --- |
| a | b | Math.pow(a,b) | Your code |
| 2.0 | 8 | 256.0 | 256.0 |
| 2.5 | 3 | 15.625 | 15.625 |
| -2.0 | 8 | 256.0 | 256.0 |
| 1.0 | 1 | 1.0 | 1.0 |
| 1.0 | 0 | 1.0 | 1.0 |
| 2.0 | 30 | 1.073741824E9 | 1.073741824E9 |
| -2.0 | 30 | 1.073741824E9 | 1.073741824E9 |
| 2.0 | -1 | 0.5 | 0.5 |
| 2.0 | -4 | 0.0625 | 0.0625 |

## Part B: Text File Processing

The file *score.csv* contains scores from the midterm examination of a programming course, which has 5 questions (Q1-Q5). The file is in the “Comma-separated Value” format (<http://en.wikipedia.org/wiki/Comma-separated_values>) with the first line being the header labels describing the order of data on the other lines.

* Read <http://docs.oracle.com/javase/7/docs/api/java/util/Scanner.html> to learn how to read a text file using an instance of the Scanner class.
* Open the file in a spreadsheet application (such as MS Excel). If you do not have any spreadsheet application on your machine, try using Google Spreadsheet.
  + Use the application to find the average score, the maximum score, and the minimum score of each question (Q1-Q5).
  + Find the average of the total score and its corresponding standard deviation.
* Fill the results in the following table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| From Spreadsheet | Average | Standard Deviation | Max | Min |
| Q1 | 5.081 | 3.1632966104122 | 10 | 0 |
| Q2 | 5.014 | 3.2441221198235 | 10 | 0 |
| Q3 | 5.586 | 2.260910373216 | 9 | 2 |
| Q4 | 7.499 | 1.737532163160 | 10 | 5 |
| Q5 | 5.478 | 1.733932675314 | 8 | 3 |
| Total | 5.7316 | 2.67548 | 10 | 0 |

* Write a Java program to:
  + Compute the average score, the maximum score, and the minimum score of each question (Q1-Q5).
  + Compute the average of the total score and its corresponding standard deviation.
* Fill the results in the following table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| From Your Java App | Average | Standard Deviation | Max | Min |
| Q1 | 5.081 | 3.1632966104122144 | 10 | 0 |
| Q2 | 5.014 | 3.2441221198235346 | 10 | 0 |
| Q3 | 5.586 | 2.260910373216001 | 9 | 2 |
| Q4 | 7.499 | 1.737532163160736 | 10 | 5 |
| Q5 | 5.478 | 1.7339326753142763 | 8 | 3 |
| Total | 5.7316 | 2.675479971634197 | 10 | 0 |

List your code here.

import java.io.File;

import java.io.FileNotFoundException;

import java.util.Scanner;

public class Iteration{

public static void main(String[] args) throws FileNotFoundException{

Scanner sc1 = new Scanner( new File ("score.csv"));

sc1.nextLine();

int q1max=0;

int q2max=0;

int q3max=0;

int q4max=0;

int q5max=0;

int q1min=10;

int q2min=10;

int q3min=10;

int q4min=10;

int q5min=10;

int maxt = 0;

int mint = 0;

int line = 0;

double sum1 = 0;

double sum2 = 0;

double sum3 = 0;

double sum4 = 0;

double sum5 = 0;

while(sc1.hasNextLine()){

String str = sc1.nextLine();

Scanner sc2 = new Scanner(str).useDelimiter(",");

sc2.next();

int q1 = Integer.parseInt(sc2.next());

int q2 = Integer.parseInt(sc2.next());

int q3 = Integer.parseInt(sc2.next());

int q4 = Integer.parseInt(sc2.next());

int q5 = Integer.parseInt(sc2.next());

sum1+=q1;

sum2+=q2;

sum3+=q3;

sum4+=q4;

sum5+=q5;

if(q1 > q1max) q1max = q1;

if(q2 > q2max) q2max = q2;

if(q3 > q3max) q3max = q3;

if(q4 > q4max) q4max = q4;

if(q5 > q5max) q5max = q5;

if(q1 < q1min) q1min = q1;

if(q2 < q2min) q2min = q2;

if(q3 < q3min) q3min = q3;

if(q4 < q4min) q4min = q4;

if(q5 < q5min) q5min = q5;

line++;

}

double avg1 = sum1/line;

double avg2 = sum2/line;

double avg3 = sum3/line;

double avg4 = sum4/line;

double avg5 = sum5/line;

double avgt = (sum1+sum2+sum3+sum4+sum5)/(5\*line);

maxt = q1max>maxt?q1max:maxt;

maxt = q2max>maxt?q2max:maxt;

maxt = q3max>maxt?q3max:maxt;

maxt = q4max>maxt?q4max:maxt;

maxt = q5max>maxt?q5max:maxt;

mint = q1min<mint?q1min:mint;

mint = q2min<mint?q2min:mint;

mint = q3min<mint?q3min:mint;

mint = q4min<mint?q4min:mint;

mint = q5min<mint?q5min:mint;

sc1.close();

double sd1 = 0;

double sd2 = 0;

double sd3 = 0;

double sd4 = 0;

double sd5 = 0;

double sdt = 0;

Scanner sc3 = new Scanner( new File ("score.csv"));

sc3.nextLine();

while(sc3.hasNextLine()){

String str = sc3.next();

Scanner sc2 = new Scanner(str).useDelimiter(",");

sc2.next();

int q1 = Integer.parseInt(sc2.next());

int q2 = Integer.parseInt(sc2.next());

int q3 = Integer.parseInt(sc2.next());

int q4 = Integer.parseInt(sc2.next());

int q5 = Integer.parseInt(sc2.next());

sd1 += Math.pow(q1-avg1,2);

sd2 += Math.pow(q2-avg2,2);

sd3 += Math.pow(q3-avg3,2);

sd4 += Math.pow(q4-avg4,2);

sd5 += Math.pow(q5-avg5,2);

sdt += Math.pow(q1-avgt,2)+Math.pow(q2-avgt,2)+Math.pow(q3-avgt,2)+Math.pow(q4-avgt,2)+Math.pow(q5-avgt,2);

}

sd1 = Math.sqrt(sd1/(line-1));

sd2 = Math.sqrt(sd2/(line-1));

sd3 = Math.sqrt(sd3/(line-1));

sd4 = Math.sqrt(sd4/(line-1));

sd5 = Math.sqrt(sd5/(line-1));

sdt = Math.sqrt(sdt/((5\*line)-1));

sc3.close();

System.out.println("max = " + q1max + ", " + q2max + ", " + q3max + ", " + q4max + ", " + q5max + ", " + maxt);

System.out.println("min = " + q1min + ", " + q2min + ", " + q3min + ", " + q4min + ", " + q5min + ", " + mint);

System.out.println("average = " + avg1 + ", " + avg2 + ", " + avg3 + ", " + avg4 + ", " + avg5 + ", " + avgt);

System.out.println("sd(population) = " + sd1 + ", " + sd2 + ", " + sd3 + ", " + sd4 + ", " + sd5 + ", " + sdt);

}

}

Submit this worksheet (by only one member of the group) via <http://www.myCourseVille.com> (Assignments > Hands-on Experiment # 7) before noon of the day after your lecture.