

Basis of Computer Programming (java A)

Tutorial 3

[Experimental Objective]

1. Learn how to use command line parameters to input values, which is (*String[] args*) in *public static void main()*.and review to input values by **Scanner**.
2. Learn how to use the *if* and *if...else* selection statements to choose among alternative actions.
3. Learn how to use the *while* repetition statement to execute statements in a program repeatedly.

[Exercises]

1. (*String[] args*) is a set of values to input to *public static void main()*, which is a String array we will learn later. There may be several values, 0 or 1 or more. It should be parsed in the *main()* method, to many different data types.

Rewrite an application to input name, age, weight and grade in command line, and prints out in a specific format, which achieves almost the same effects like exercise 2 in Lab2.

Sample code:

```
public class Lab3_E1 {  
    public static void main(String[] args) {  
        String name = args[0];  
        int age = Integer.parseInt(args[1]);  
        float weight = Float.parseFloat(args[2]);  
        char grade = args[3].charAt(0);  
  
        System.out.printf("You are %s. \nYou are %d years old. \n", name, age);  
        System.out.printf("You weigh %.1f KG. \n The highest grade you got is %c. \n", weight, grade);  
    }  
}
```

It can be compiled and run by different methods.

Notice: the parameters should be separated by Blank space.

(1) in command line

```
C:\workspace\javaLab\src\Lab3>javac Lab3_E1. java  
  
C:\workspace\javaLab\src\Lab3>java Lab3_E1 zhangsan 20 60.1 A  
You are zhangsan.  
You are 20 years old.  
You weigh 60.1 KG.  
The highest grade you got is A.  
  
C:\workspace\javaLab\src\Lab3>
```

(2) in IDEA

Settings:

Name: Lab3_E1

Configuration Code Coverage Logs

Main class: Lab3_E1

VM options:

Program arguments: zhangsan 20 60.1 A

Working directory: C:\workspace\javaLab

Environment variables:

☐ Redirect input from:

Use classpath of module: javaLab

☐ Include dependencies with "Provided" scope

JRE: Default (1.8 - SDK of 'javaLab' module)

Shorten command line: user-local default: none - java [options] class

☐ Enable capturing form snapshots

Sample output:

```

Lab3_E1 x
"C:\Program Files\Java\jdk1.8.0_191\bin\java.exe" ...
You are zhangsan.
You are 20 years old.
You weigh 60.1 KG.
The highest grade you got is A.

Process finished with exit code 0

```

What will happen if you input more or less than 4 parameters, or input wrong data type?

- Write an application which can convert the grades on 100-point scale into GPA according to the following table.

Grade	GPA
100~90	4.0
89~80	3.0
79~70	2.0
69~60	1.0
59~0	0

Sample code:

```
public class Lab3_E2 {  
    public static void main(String[] args) {  
        float gpa;  
        float score = Float.parseFloat(args[0]);  
  
        if(score >= 60)  
            System.out.println("You passed the exam.");  
        else  
            System.out.println("You failed in the exam.");  
  
        if(score >= 90)  
            gpa = (float) 4.0;  
        else if(score >= 80)  
            gpa = 3.0f;  
        else if(score >= 70)  
            gpa = 2.0f;  
        else if(score >= 60)  
            gpa = 1.0f;  
        else if(score > 0)  
            gpa = 0.0f;  
        else  
        {  
            System.out.println("Invalid grade");  
            return;  
        }  
        System.out.printf("Your score is %.1f, the GPA is %.1f\n", score, gpa);  
    }  
}
```

Sample output:

```
C:\workspace\javaLab\src\Lab3>javac Lab3_E2. java  
  
C:\workspace\javaLab\src\Lab3>java Lab3_E2 96  
You passed the exam.  
Your score is 96.0, the GPA is 4.0  
  
C:\workspace\javaLab\src\Lab3>java Lab3_E2 85  
You passed the exam.  
Your score is 85.0, the GPA is 3.0  
  
C:\workspace\javaLab\src\Lab3>java Lab3_E2 77  
You passed the exam.  
Your score is 77.0, the GPA is 2.0  
  
C:\workspace\javaLab\src\Lab3>java Lab3_E2 60  
You passed the exam.  
Your score is 60.0, the GPA is 1.0  
  
C:\workspace\javaLab\src\Lab3>java Lab3_E2 59  
You failed in the exam.  
Your score is 59.0, the GPA is 0.0
```

3. Write a program to print 99 multiplication table, by using the *while* repetition statement.

Notice: printf using %02d, "3" will be "03"; %2d, "3" will be " 3", with a blank space on the left.

Sample output:

```

1 * 1 = 1
1 * 2 = 2  2 * 2 = 4
1 * 3 = 3  2 * 3 = 6  3 * 3 = 9
1 * 4 = 4  2 * 4 = 8  3 * 4 = 12  4 * 4 = 16
1 * 5 = 5  2 * 5 = 10  3 * 5 = 15  4 * 5 = 20  5 * 5 = 25
1 * 6 = 6  2 * 6 = 12  3 * 6 = 18  4 * 6 = 24  5 * 6 = 30  6 * 6 = 36
1 * 7 = 7  2 * 7 = 14  3 * 7 = 21  4 * 7 = 28  5 * 7 = 35  6 * 7 = 42  7 * 7 = 49
1 * 8 = 8  2 * 8 = 16  3 * 8 = 24  4 * 8 = 32  5 * 8 = 40  6 * 8 = 48  7 * 8 = 56  8 * 8 = 64
1 * 9 = 9  2 * 9 = 18  3 * 9 = 27  4 * 9 = 36  5 * 9 = 45  6 * 9 = 54  7 * 9 = 63  8 * 9 = 72  9 * 9 = 81

```

4. Create a class called `GuessingNumber`. In the main method, you should generate a random integer `magicNum` between 0 and 9, then keep asking the user to input an integer between 0 and 9 until the input number is equal to the attribute `magicNum`. When the input number is greater than the attribute `magicNum`, the method should output "Too big!Please try again:". When the input number is less than the attribute `magicNum`, the method should output "Too small!Please try again:". Then the method waits for the user to input a new integer. When the input number is equal to the attribute `magicNum`, the method should output "Congratulations!" and terminate.

Sample code:

```
import java.util.Random;
```

```

public static void main(String[] args) {

    Random random = new Random();
    int magicNum = random.nextInt(10);
    int inputNum;
    Scanner sc = new Scanner(System.in);

    System.out.println("Please input an Integer in
{0,1,2,...,9}:");
    inputNum = sc.nextInt();

    while( ) { // to finish it
        if ( ) // to finish it
            System.out.println("Too big!Please try again:");
        else
            System.out.println("Too small!Please try again:");
        inputNum = sc.nextInt();
    }

    System.out.println("Congratulations!");
    sc.close();
}

```

Sample output:

```
Please input an Integer in {0,1,2,...,9}:  
3  
Too small!Please try again:  
5  
Too small!Please try again:  
7  
Congratulations!
```

5. Calculate the value of π from the infinite series

$$\pi = 4 - \frac{4}{3} + \frac{4}{5} - \frac{4}{7} + \frac{4}{9} - \frac{4}{11} + \dots$$

Input an integer n which presents the times to compute as above. It is precise when n is bigger. Using *while* repetition statements to show the value of π .

sample output:

```
Please input n:  
10000  
The estimatioin of Pi is 3.141498
```

Advanced:

Input a double which presents a precision threshold. And the program will terminate when the difference between two successive values being smaller than the precision threshold. Print the value of the estimation, and the iteration numbers.

Sample running:

```
Please input the precision:  
0.0001  
The estimatioin of Pi is 3.141547  
It computed 19998 times
```

Tips: Math.abs()

<https://docs.oracle.com/javase/8/docs/api/java/lang/Math.html>