

CS102A Introduction to Computer Programming

Fall 2020

Lab 3

Objectives

1. Learn how to obtain user input from the command line arguments or via the `Scanner` class.
2. Learn how to use the `if` and `if...else` conditional statements to choose the next block of code to be executed from an optional set of alternatives.
3. Learn how to use the `while` loop to repeatedly execute statements in a block of code.

1 Exercises

1.1 Exercise 1

Given the command `java Hello I Love Programming`, the Java loader will assume `Hello` to be a class name, and will therefore attempt to locate and run a `Hello.class` file in the current directory. The string following the class name `Hello`, i.e., `I Love Programming`, comprises the command line arguments. When JVM receives command line arguments, it wraps and passes them to `String[] args` in your `main` method. To confirm this, you may check the length of `args` using `args.length`.

Write a Java program that gets a user's name, age, weight, and grade from the command line arguments, then prints the above information using a specific format. Your program should have similar output to Exercise 2 in Lab 2:

```

1 public class Lab3_E1 {
2     public static void main(String[] args) {
3         String name = args[0];
4         int age = Integer.parseInt(args[1]);
5         float weight = Float.parseFloat(args[2]);
6         char grade = args[3].charAt(0);
7
8         System.out.printf("You are %s.\nYou are %d years old.\n",
9             name, age);
10        System.out.printf("You weigh %.1f KG.\nThe highest grade
11            you got is %c.\n", weight, grade);
12    }
13 }

```

Command line arguments can be obtained from the IDE or the command line:

1. Using the command line:

```

> javac Lab3_E1.java

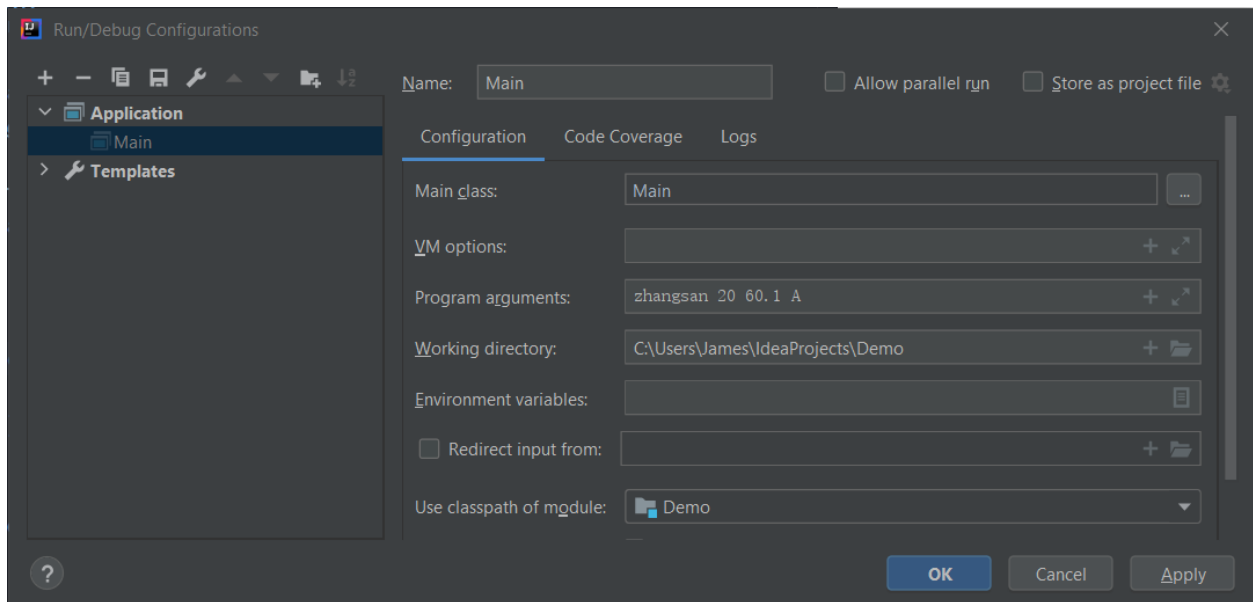
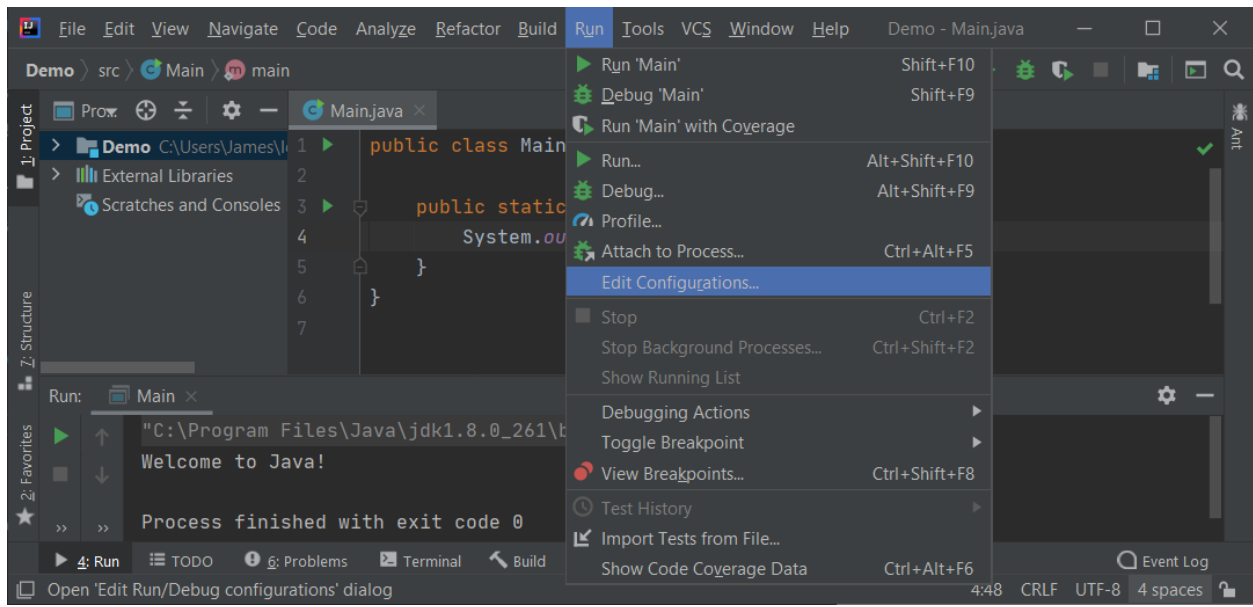
> java Lab3_E1 zhangsan 20 60.1 A
You are zhangsan.
You are 20 years old.
You weigh 60.1 KG.
The highest grade you got is A.

```

2. Using IDEA: Input `zhangsan 20 60.1 A` in the *Program arguments* box, located under the *Configuration* tab of your project settings.

? Question

What will happen if you input more or fewer than four arguments, or if you input the wrong data type for one or more of the arguments?



1.2 Exercise 2

Write a program that converts grades on a 100-point scale into a 4-point GPA scale 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:

```

1 public class Lab3_E2 {
2     public static void main(String[] args) {
3         float gpa;
4         float score = Float.parseFloat(args[0]);
5
6         if(score >= 60)
7             System.out.println("You passed the exam.");
8         else
9             System.out.println("You failed in the exam.");
10
11        if(score >= 90)
12            gpa = (float) 4.0;
13        else if(score >= 80)
14            gpa = 3.0f;
15        else if(score >= 70)
16            gpa = 2.0f;
17        else if(score >= 60)
18            gpa = 1.0f;
19        else if(score > 0)
20            gpa = 0.0f;
21        else {
22            System.out.println("Invalid grade");
23            return;
24        }
25        System.out.printf("Your score is %.1f, the GPA is %.1f\n"

```

```
        , score, gpa);  
26     }  
27 }
```

Sample output:

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

1.3 Exercise 3

Write a program to print a 9×9 multiplication table using the `while` loop.



Note

If you use `printf` with `%02d`, 3 will be printed as `03`; with `%2d`, 3 will be printed as `3` (with a 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
```

1.4 Exercise 4

Create a class called `GuessingNumber`. In the `main` method, you should generate a random integer `magicNum` between 0 and 9, then repeatedly ask the user to input an integer between 0 and 9 until the input number is equal to `magicNum`. When the input number is greater than `magicNum`, the method should output *Too big! Please try again:*. When the input number is less than `magicNum`, the method should output *Too small! Please try again:*. When the user finally inputs a number equal to `magicNum`, the method should output *Congratulations!* and terminate.

Sample code:

```
1 import java.util.Random;
2 import java.util.Scanner;
3
4 public class GuessingNumber {
```

```

5     public static void main(String[] args) {
6         Random random = new Random();
7         int magicNum = random.nextInt(10);
8
9         int inputNum;
10
11        Scanner sc = new Scanner(System.in);
12
13        System.out.println("Please input an Integer in
14        {0,1,2,...,9}:");
15        inputNum = sc.nextInt();
16
17        while (                ){ // Finish this line
18            if (                ) // Finish this line
19                System.out.println("Too big! Please try again:");
20            else
21                System.out.println("Too small! Please try again:");
22            );
23            inputNum = sc.nextInt();
24        }
25        System.out.println("Congratulations!");
26        sc.close();
27    }
28 }

```

Sample output:

```

Please input an Integer in {0,1,2,...,9}:
1
Too small! Please try again:
5
Too small! Please try again:
8
Too small! Please try again:

```

9

Congratulations!

1.5 Exercise 5

Write a program that calculates the value of π from the infinite series $\pi = 4 - 4/3 + 4/5 - 4/7 + 4/9 - 4/11 + \dots$. The program should ask the user to input an integer `n` corresponding to the number of terms in the formula above. The calculation of π is more precise when `n` is bigger. Use `do...while` or `while` loops to compute the value of π .

Sample output:

```
Please input the precision:
10000
The estimation of Pi is 3.141498
```

Next, modify your program as follows: the user is asked to input a double value representing a precision threshold. Your program should terminate when the difference between two successive iterations is smaller than the precision threshold. Print the value of π and the iteration numbers. Sample output:

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



Tip

Use `Math.abs()`.