

Lab 1: 认识 Java

!!! Error ""

本次 Lab 时长为 3 周，DDL 为 **9 月 25 日 24:00**。😬

写在前面

每次实验的问答报告主要包括问答和编码两部分，具体提交内容请以题面说明为准。

对于每次实验的问答报告部分，你需要在**云平台**上提交一个压缩包，命名格式为 `学号-姓名-Labxx.zip`，如 `22373000-张三-Lab01`。所有问答题的回答和截图放在一个 PDF 文件中，可通过 Markdown 或 Word 导出。如有需要提交代码的问题，则新建一个文件夹用于存放所有代码，并在相应题目的回答中做出说明。提交的文件结构如下（仅供参考）：

```
22373000-xxx-Lab01
|-- 22373000-xxx-Lab01.pdf
|-- Q2
|   |-- Main.java
|-- Q3
|   |-- Test.java
|   |-- Hello.java
\-- Q6
     |-- Test.java
```

!!! Info ""

内容模板可参考 [Lab Solution 报告模板](#)。😬

1. 实验目的

1. 掌握 Java 环境配置。
2. 使用 CMD 编译运行 Java 程序。
3. 使用 IDE 编译运行 Java 应用程序。
4. 理解 Java 语言中的关键字、标识符并能灵活使用。
5. 理解、掌握 Java 语言的简单数据类型并能灵活使用（重点强制类型转换）。
6. 理解引用数据类型，理解简单数据类型和引用数据类型的特点。
7. 理解、掌握 Java 语言中的运算符并能灵活使用。
8. 理解、掌握常量与变量的概念并能灵活使用。

2. Hello, World!

Question01

借助身边的工具，学习 `java`，`javac` 等命令的使用，并尝试编译运行一个 Java 程序。

掌握基本使用即可，不需要提交任何内容。

Question02

学习一个新语言的第一步，自然是输出 `Hello, world!`。

但在这里我们略微增加一点点难度 ~ 你需要输出很多 `Hello, world!`

本题需要你实现一个 `Test` 类：

- 程序开始运行，进入 `main` 方法，并输出你的学号和姓名
- 当终端输入 `QUIT` 时，系统退出，并在终端打印一行字符：

```
----- Good Bye! -----
```

- 对于其他的输入，在终端中输出一行 `Hello, world!`，等待下一行输入

你需要用**命令行**运行该程序，并**提交运行时的截图**。😊

程序退出

和 C 语言的 `return 0` 类似，Java 程序也有其退出码，整个程序的正常的退出码应该为 0，而不应该为 -1、1 等。

在测评机中，退出状态非 0 则会被认为是程序未正常结束，很容易导致你的作业被错判，因此请务必注意在循环等进行程序退出时，确保退出状态为 0

下面给出 Java 中退出的一个例子：

```
class Example {
    public static void main(String[] args) {
        int x = 0;

        while (true) {
            x++;
            if (x > 100) {
                System.out.println("get 100");
                System.exit(0);
            }
        }
    }
}

// 这只是退出程序的一种方法
// 你也可以直接使用 break，跳出 while 循环，直接到达整个程序的退出区域（默认退出状态为 0）
```

参考实现

Java 打印字符串与其他语言有所区别，一般不直接调用 `print` 函数，可以用如下语句：

```
String str = "Hello world!"
System.out.println(str);
```

Java 连续读取输入行的一种实现：

```
Scanner in = new Scanner(System.in);
String argStr;
while (true) {
    argStr = in.nextLine();
}
```

!!! Note !!!

当然，`System.out` 中也有其他的打印方法，如 `print` `printf` 等，同学们也可以按需使用。

3. IDE 的使用

Question03

- (1) 编写以下程序，并运行。将运行结果截图提交。
- (2) 将断点设置在第八行，查看变量 `a` 的值，提交此刻的屏幕截图。

```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello world!");
        int a = 1;
        a = a + 1;
        a = a + 2;
        System.out.println("a is " + a);
        a = a + 3; //断点行
        a = a + 4;
        System.out.println("a is " + a);
    }
}
```

4. 基本数据类型

Question04

复制下列代码至 IDE 中，查看程序输出结果，并提交结果内容或截图。

```
public class App1 {
    public static void main(String args[]) {
        // 定义整形变量a和双精度浮点变量b、c
        int a = 126;
        double b = 0.2;
        double c = 1.6;
        // 创建字符串对象
        String str1 = "Hello world";

        String str = new String("字符串不是基本数据类型");
        // 输出各个变量
        System.out.println("a =" + a);
        System.out.println("b + c = " + (b + c));
        System.out.println("b + c = " + b + c);
        System.out.println(str);
        System.out.println(str1);
    }
}
```

```
}  
}
```

Question05

看程序输出结果，并提交结果内容或截图。

```
public class VarDemo {  
    public static void main(String[] args) {  
        byte b = 0x55;  
        boolean B = true;  
        short s = 0x55ff;  
        int i = 1000000;  
        long l = 0xfffffL;  
        char c = 'c';  
        float f = 0.23F;  
        double d = 0.7E-3;  
        String S = "This is a string";  
        System.out.println("字节型变量 b = " + b);  
        System.out.println("短整型变量 s = " + s);  
        System.out.println("整型变量 i = " + i);  
        System.out.println("长整型变量 l = " + l);  
        System.out.println("字符型变量 c = " + c);  
        System.out.println("浮点型变量 f = " + f);  
        System.out.println("双精度变量 d = " + d);  
        System.out.println("布尔型变量 B = " + B);  
        System.out.println("字符串对象 S = " + S);  
    }  
}
```

Question06

编写一段代码验证 Java 中类的基本数据类型字段（成员变量）的默认初始化值，请按照对应类型的字面量填写。在报告中填写下面的表格。

基本类型	默认值	基本类型	默认值
byte		boolean	
short		char	
int		float	
long		double	

6. 三元运算符

Question07

看程序输出结果，并提交结果内容或截图

```
public class FindMinMax {
```

```

public static void main(String[] args) {
    double temp, max, min;
    double d1 = 1, d2 = -9.9, d3 = 96.9;
    temp = d1 > d2 ? d1 : d2;
    temp = temp > d3 ? temp : d3;
    max = temp;
    temp = d1 < d2 ? d1 : d2;
    temp = temp < d3 ? temp : d3;
    min = temp;
    System.out.println("max = " + max);
    System.out.println("min = " + min);
}
}

```

7. 短路

Question08

阅读下面这段代码：

```

// ShortCircuit.java
public class ShortCircuit {
    static void print(String s) {
        System.out.println(s);
    }
    static boolean test1(int val) {
        print("1");
        return val < 1;
    }
    static boolean test2(int val) {
        print("2");
        return val < 2;
    }
    static boolean test3(int val) {
        print("3");
        return val < 3;
    }
    public static void main(String[] args) {
        boolean b = test1(0) && test2(2) && test3(2);
        print("b is " + b);
    }
}

```

运行该程序，程序的输出是什么？**给出结果或截图。**

请解释 Java 执行串联逻辑运算时的流程，**可以用文字、流程图、伪代码描述。**

思考如何利用短路这个机制来优化程序，**写出你的想法。**

8. 全局变量

Question09

查看程序输出结果，并提交结果内容或截图。

```
public class GlobalVar {
    int a = 10; // 全局变量,下同
    double b = 20;

    public static void main(String[] args) {
        GlobalVar globalVar = new GlobalVar();
        System.out.println("全局变量 a = " + globalVar.a);
        // System.out.println("全局变量 a = " +a); 错误写法
        globalVar.print();
        System.out.println("全局变量变化后 a = " + globalVar.a);
    }

    public void print() {
        System.out.println("在 print() 中, 全局变量 a = " + a + ", b = " + b);
        a = 30;
        System.out.println("在 print() 中, 全局变量 a = " + a + ", b = " + b);
    }
}
```

9. 局部变量

Question10

查看程序输出结果，并提交结果内容或截图。

```
public class LocalVar {
    public static void main(String[] args) {
        LocalVar localVar = new LocalVar();
        // System.out.println("局部变量 a = " + localVar.a); 引用错误, 下同
        localVar.print();
        // System.out.println("变化后的局部变量 a = " + localVar.a);
    }

    public void print() {
        int a = 10; // 局部变量, 下同
        double b = 20;
        System.out.println("在 print() 中的局部变量 a = " + a + ", b = " + b);
        a = 30;
        System.out.println("在 print() 中的局部变量 a = " + a + ", b = " + b);
    }
}
```

