

Lab01: Java 语法 & 面向对象初步

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
9. 理解、掌握 Java 语言的流程控制（重点 `switch` `case` `break` 语句）并能灵活使用
0. 理解、掌握 Java 语言的数组的声明、赋值及使用
1. 理解掌握面向对象思维方式，确定问题域，使用面向对象思维解决实际问题

2. Class 初步

Question01: 阅读下列 Java 源文件，并回答问题。

```
1 public class Person {
2     void speakHello() {
3         System.out.print("您好, 很高兴认识您");
4         System.out.println(" nice to meet you");
5     }
6 }
7
8 class Xiti {
9     public static void main(String args[]) {
10        Person zhang = new Person();
11        zhang.speakHello();
12    }
13 }
```

- (a) 上述源文件的名字是什么？
- (b) 编译上述源文件将生成几个字节码文件？这些字节码文件的名称都是什么？
- (c) 在命令行执行 `java Person` 得到怎样的错误提示？执行 `java xiti` 得到怎样的错误提示？执行 `java Xiti.class` 得到怎样的错误提示？执行 `java Xiti` 得到怎样的输出结果？

Question02: 编写程序输出自己的学号姓名，运行并截图。

3. Unicode 编码

Question03: 看程序输出结果,并提交结果内容或截图

```

1 public class Unicode {
2     public static void main(String args[]) {
3         char c = '大';
4         System.out.println("\'" + c + "\'的Unicode编码: " + (int) c);
5         int num = 23398;
6         System.out.println("Unicode编码为" + num + "的字符是: " + (char) num);
7     }
8 }

```

4. 数组

Question04: 看程序输出结果,并提交结果内容或截图

```

1 public class UseIntArray {
2     public static void main(String[] args) {
3         int intArray[] = { 8, 9, 12, 13, 14 };
4         int sum = 0;
5         for (int i = 0; i < intArray.length; i++)
6             sum += intArray[i];
7         for (int i = 0; i < intArray.length; i++)
8             System.out.println("intArray[" + i + "]= " + intArray[i]);
9         System.out.println();
10        System.out.println("sum=" + sum);
11    }
12 }

```

Question05: 看程序输出结果,并提交结果内容或截图:

```

1 class TwoDimensionArray {
2     public static void main(String args[]) {
3         int b[][] = { { 11 }, { 21, 22 }, { 31, 32, 33 } };
4         int sum = 0;
5         b[0][0] = 1000;
6         for (int i = 0; i < b.length; i++) {
7             for (int j = 0; j < b[i].length; j++)
8                 sum += b[i][j];
9         }
10        System.out.println("b[0][0]= " + b[0][0] + '\n' + "sum=" + sum + '\n'
11            + "b.length=" + b.length);
12
13        int arr1[][] = new int[3][4];
14        int arr2[][] = new int[3][];
15        int arr3[][] = { { 0, 1, 2 }, { 3, 4, 5 }, { 6, 7, 8 } };

```

```

16     int i, j, k = 0;
17
18     // arr1[][]赋值
19     for (i = 0; i < 3; i++)
20         // arr1为3行4列
21         for (j = 0; j < 4; j++)
22             arr1[i][j] = k++;
23
24     // arr2[][]开辟内存空间
25     for (i = 0; i < 3; i++)
26         // arr2每一行是变长的，元素个数为3、4、5
27         arr2[i] = new int[i + 3];
28     // arr2[][]赋值
29     for (i = 0; i < 3; i++)
30         for (j = 0; j < arr2[i].length; j++)
31             arr2[i][j] = k++;
32
33     // 输出arr1
34     System.out.println(" arr1:");
35     for (i = 0; i < 3; i++)
36         for (j = 0; j < 4; j++)
37             System.out.print(" " + arr1[i][j]);
38     System.out.println();
39
40     // 输出arr2
41     System.out.println(" arr2:");
42     for (i = 0; i < 3; i++)
43         for (j = 0; j < arr2[i].length; j++)
44             System.out.println(" " + arr2[i][j]);
45
46     // 输出arr3
47     System.out.println(" arr3:");
48     for (i = 0; i < 3; i++)
49         // arr3为3行3列
50         for (j = 0; j < 3; j++)
51             System.out.print(" " + arr3[i][j]);
52 }
53 }

```

5. switch

Question06: 阅读下面程序，给出程序的输出结果，为什么？

```

1 public class E {

```

```

2      public static void main (String args[ ]) {
3          char c = '\0';
4          for(int i=1;i<=4;i++) {
5              switch(i) {
6                  case 1: c = 'J';
7                          System.out.print(c);
8                  case 2: c = 'e';
9                          System.out.print(c);
10                     break;
11                 case 3: c = 'p';
12                         System.out.print(c);
13                 default: System.out.print("好");
14             }
15         }
16     }
17 }

```

6. 循环

Question07: 请实现程序输出以下 5*5 数字方格。

```

1  1 2 3 4 5
2  6 7 8 9 10
3 11 12 13 14 15
4 16 17 18 19 20
5 21 22 23 24 25

```

需要在 main 函数的输入参数中设置 5，输出 5 * 5 的数字方格。如果是输入 7，则是 7 * 7 的数字方格。需要提交代码及运行结果截图。

Question07: 请实现程序输出以下星塔。

需要在 main 函数的输入参数中设置 5，输出 5 层星塔。如果是输入 7，则输出 7 层星塔。假设输入参数都是奇数，且都大于等于 5，小于等于 11，提交代码及运行结果截图。

```

1      *
2     ***
3    *****
4   ***
5      *

```

Question09: 编写静态方法 double getPi(int n)，利用公式 $\pi = 4 \sum_{i=0}^n \frac{(-1)^i}{2i+1}$ ，求 pi 的近似值，需求如下

```

1  /**
2  * 利用公式\pi = 4\sum_{i=0}^n\{\frac{\left(-1\right)^{i}}{2i+1}\}求指定精度的圆周率
3  * 如果n是负数，视其为0
4  * @param n 公式中的i的上限
5  * @return 圆周率的近似值
6  */
7  public static double getPi(int n);

```

题外话：

如果 n 是负数，视其为 0

有穷求和的区间是闭区间，上限 n 对应的级数项也是要计算的。

Question10：快速排序

编写静态方法 `int[] qsort(int[] arr)`，需求如下：

```

1  /**
2  * 对一个int数组进行快速排序，结果是升序的，并且不会开辟额外的数组空间
3  * @param arr 待排序的数组
4  * @return 升序排列好的arr，如果arr==null，则返回null
5  */
6  public static int[] qsort(int[] arr);
7

```

约束：

返回值必须是原数组，即 `return arr`。

可以任意次创建指向原数组的引用，但是请不要在排序过程中创建新的数组。

可以根据需要定义新的函数。

如果数组是 `null`，返回 `null`。

对于空数组 (`arr.length==0`)，原样返回，而不是 `null`。

这一次请不要使用 `java.util.Array.sort()` 之类的操作。

Question13：现有一组数据分别是：12,45,67,89,123,-45,67，实现以下两步：

- (1) 请对这组数据排序后，按照从小到大的顺序打印；
- (2) 从键盘输入任意整数，程序判断该整数是否存在该数组中；

Question11：阅读下面代码并查看输出结果

```

1  public class ForInString {
2      public static void main(String[] args) {
3          for(char c : "An African Swallow".toCharArray())
4              System.out.print(c + " ");
5      }
6  }

```

7. 引用

Question12: 阅读下面这段代码:

```

1  class Point {
2      int x,y;
3      void setXY(int m,int n){
4          x = m;
5          y = n;
6      }
7  }
8  public class Example {
9      public static void main(String args[]) {
10         Point p1,p2;
11         p1=new Point();
12         p2=new Point();
13         System.out.println("p1的引用:"+p1);
14         System.out.println("p2的引用:"+p2);
15         p1.setXY(1111, 2222);
16         p2.setXY(-100, -200);
17         System.out.println("p1的x,y坐标:"+p1.x+","+p1.y); // 1
18         System.out.println("p2的x,y坐标:"+p2.x+","+p2.y); // 2
19         p1 = p2;
20         p1.setXY(0, 0);
21         System.out.println("p1的引用:"+p1);
22         System.out.println("p2的引用:"+p2);
23         System.out.println("p1的x,y坐标:"+p1.x+","+p1.y); // 3
24         System.out.println("p2的x,y坐标:"+p2.x+","+p2.y); // 4
25     }
26 }

```

请指出代码中注释标注出的四行输出的内容会是什么。

题外话:

意在理解 Java 对象实例是作为引用存在的。

如果你要为一个类实现 `copy()` 方法, 思考如何避免引用间赋值导致的浅拷贝? 或者说, 如何确保进行的是深拷贝。

你需要查一下什么是浅拷贝? 什么是深拷贝?

同样的问题也存在于 String 的 `==` 和 `equals()` 方法, 对于不同的背景, 相等的定义可能也不同。

Question13: 阅读下面程序, 给出程序的输出结果, 为什么?

```

1  public class Test{
2      public static void main(String args[]) {
3          int a[]={1,2,3,4};
4          int b[]={100,200,300};
5          System.out.println("数组a的元素个数="+a.length);
6          System.out.println("数组b的元素个数="+b.length);
7          System.out.println("数组a的引用="+a);

```

```

8      System.out.println("数组b的引用="+b);
9      a=b;
10     System.out.println("数组a的元素个数="+a.length);
11     System.out.println("数组b的元素个数="+b.length);
12     System.out.println("a[0]="+a[0]+" ,a[1]="+a[1]+" ,a[2]="+a[2]);
13     System.out.print("b[0]="+b[0]+" ,b[1]="+b[1]+" ,b[2]="+b[2]);
14 }
15 }
16

```

8. 可变参数

Question14: 看程序输出结果,并提交结果内容或截图:

```

1 public class E {
2     public static void main(String args[]) {
3         f(1,2);
4         f(-1,-2,-3,-4); //给参数传值时, 实参的个数很灵活
5         f(9,7,6) ;
6     }
7     public static void f(int ... x){ //x是可变参数的代表, 代表若干个int型参数
8         for(int i=0;i<x.length;i++) { //x.length是x代表的参数的个数
9             System.out.println(x[i]); //x[i]是x代表的第i个参数(类似数组)
10        }
11    }
12 }

```

Question15: 编写静态方法 `String strscat(String... args)` , 需求如下:

```

1 /**
2  * 将任意个字符串顺序连接, 不应该改变任意一个原有参数
3  * @param args 字符串们
4  * @return args中的字符串顺序连接组成的新字符串
5  */
6 public static String strscat(String... args);

```

样例:

调用方法	返回值
<code>strscat("a", "b", "c", "", "e")</code>	<code>"abce"</code>
<code>strscat("str")</code>	<code>"str"</code>

回答:

请尝试理解 Java 可变参数的实现机制。

调用 `strscat(new String[]{"a", "b"})` 能通过编译吗？

如果还有静态方法 `String strscat(String[] args)` 同时存在，代码能编译通过吗？这时传入的参数是什么？又会返回什么？

如果我们声明的是 `String strscat(String[] args)`，`strscat()` 这样的调用还能通过编译吗？如果能，这时传入参数和返回值的情况和 `String strscat(String... args)` 通过相同吗？

9. 面向对象（复习并完成课堂知识点）

Question16：减肥。小王本来体重 70Kg，经过减肥，体重降到 45Kg，试从这个问题领域中识别对象、类、属性、行为、状态，和状态变化。尝试将你抽象出来的类用 Java 代码实现出来，并编写一个测试类，来描述上述过程。

Question17：制作一桌丰盛的年夜饭。定义采购员类和厨师类的属性和行为，在测试类中产生采购员的对象：姑姑，妈妈和你，在测试类中产生出厨师的对象：爸爸和婶婶，协同制作年夜饭。