# 三套真题答案

真题1答案

一

1．B 2.A 3.C 4.B 5.D 6.A 7.B 8.C 9.B 10.B 11. C

二

1.BD 2.AE 3.AC 4.AE 5.AD 6.DEF 7.AC 8.BC 9.AB 10.CD 11.BCD 12. ACD 13.ABD

三、

1、a=3,b=4,c=5,x=10

2. Java.util

3. public static void main(String[] args)

4. 1

5. 26

四

1.

|  |
| --- |
| class Test  {  public static void main(String[] args)  {  System.out.println("Hello world");  }  } |
| **import** java.applet.Applet;  **import** java.awt.Graphics;  **class** Test **extends** Applet  {  **private** String str;  **public** **void** init()  {  str="Hello world";  }  **public** **void** paint(Graphics g)  {  g.drawString(str, 10, 10);  }  } |

2.

|  |
| --- |
| **import** java.io.BufferedOutputStream;  **import** java.io.File;  **import** java.io.FileNotFoundException;  **import** java.io.FileOutputStream;  **import** java.io.IOException;  **import** java.util.Scanner;  **class** Test  {  **public** **static** **void** main(String[] args)  {  String name;  String phone;  Scanner scanner = **new** Scanner(System.*in*);  BufferedOutputStream buff=**null**;  FileOutputStream fos;  **try** {  fos = **new** FileOutputStream(**new** File("phone.txt"));  buff=**new** BufferedOutputStream(fos);    **while**(**true**)  {  System.*out*.println("请输入名字，或输入done退出");  name = scanner.nextLine();  **if**(name.trim().equalsIgnoreCase("done"))  **break**;  System.*out*.println("请输入手机号，或输入done退出");  phone = scanner.nextLine();  **if**(phone.trim().equalsIgnoreCase("done"))  **break**;  buff.write( (phone+","+name+"\n").getBytes());  }  } **catch** (Exception e)  {  e.printStackTrace();  } **finally**  {  **if**(buff!=**null**)  **try** {  buff.close();  } **catch** (IOException e) {  // **TODO** Auto-generated catch block  e.printStackTrace();  }  }  }  } |

真题2答案

一、

1.BCDE 2.B 3.A 4.B 5.C 6.A 7.D 8D 9.A 10.B 11.A 12.A 13.D 14.A 15.B

二、

1、答案：静态成员变量，成员变量，构造函数。

2、答案：final， abstract。

3、答案：4，2。

4、重载，多态。

5、答案：继承Thread类，实现Runnable接口。

**三、**

**1、**请参考4.2.2节

**2、**接口可以继承接口，抽象类可以实现接口，而抽象类不能继承实体类。

**3、请参考4.5.3节**

**4、请参考4.9.3和4.9.4节**

**5、请参考4.3.3节**

**四、**

**1、**请参考8.3节

**2、**

在Java语言中，默认使用的Unicode编码方式，即每个字符占用两个字节，因此可以用来存储中文。虽然String是由char所组成的，但是它采用了一种更加灵活的方式来存储，即英文占用一个字符，中文占用两个字符，采用这种存储方式的一个重要作用就是可以减少所需的存储空间，提高存储效率。根据这个特点，可以采用如下代码来完成题目的要求：

|  |
| --- |
| **public** **class** Test {  **public** **static** **boolean** isChinese(**char** c) {  String sb = String.*valueOf*(c);  **return** sb.getBytes().length > 1 ? **true** : **false**;  }  **public** String truncateStr(String str, **int** len) {  **if** (str == **null** || str.equals("") || len == 0)  **return** "";  **char**[] chrArr = str.toCharArray();  StringBuilder sb = **new** StringBuilder("");  **int** count = 0;  **for** (**char** cc : chrArr) {  **if** (count < len) {  **if** (*isChinese*(cc)) {  **if** (count + 1 == len)  **return** sb.toString();  count = count + 2;  sb = sb.append(cc);  } **else** {  count = count + 1;  sb = sb.append(cc);  }  } **else** {  **break**;  }  }  **return** sb.toString();  }  **public** **static** **void** main(String[] args) {  Test splitStr = **new** Test();  String sb = "我b们c";  System.*out*.println(splitStr.truncateStr(sb, 6));  }  } |

程序运行结果为：

|  |
| --- |
| 我b们c |

**真题3答案**

**一、单项选择题**

**1.**B **2.**B **3.**C  4.A 5.A 6.C 7.D 8.A 9.D

 二、

1．BCD 2．ACD 3．BCD 4．ACD 5．ACD 6．ACD  7．ABC 8．BCD

三、

1．正确  2.正确  3.错误

四、

1、 “hello”。

**2、**ABDCBDCB。

五、简答题

**1、**

以下将分别对&与&&进行说明。

&是按位与操作符，a&b是把a和b都转换成二进制数后，然后再进行与的运算。而&&为逻辑与操作符，a&&b就是当且仅当两个操作数均为 true时，其结果才为 true，只要有一个为false，a&&b的结果就为false。

此外，&&还具有短路的功能，在参与运算的两个表达式中，只有当第一个表达式的值为true的时候，才会去计算第二个表达式的值，如果第一个表达式的值为false，则此时与运算的结果就为false，同时不会去计算第二个表达式的值，例如：if(i!=0 && i++>10)，当i=1的时候，表达式i!=0的返回值为true，因此，此时将不会执行第二个表达式i++>10的判断。

**2、**

参考4.9.4节

**3、**

参考4.9.6节

**4、**

参考4.2.8节

**5、**

参考4.3.3节

  六、加分题

**1、什么是Spring MVC？什么是IoC？什么是AOP？**

答案：以下将分别对SpringMVC、IoC、AOP进行解释说明。

（1）SpringMVC

Spring MVC是在Spring框架上发展起来的框架，它提供了构建 Web 应用程序的全功能 MVC 模块，使用了 Spring 可插入的 MVC 架构，可以自由地选择各个模块所使用的架构，非常灵活。例如可以选择使用内置的 Spring Web 框架，也可以选择使用 Struts 的 Web 框架。通过策略接口，Spring 框架是高度可配置的。它包含多种视图技术，例如 JavaServer Pages（JSP）技术和Velocity模板引擎等。对于Spring MVC 框架而言，它并不知道具体使用了哪个视图。Spring MVC把控制器、模型、分派器以及处理程序对象的角色进行了分离，因此Spring MVC有更好的可定制性。

对于IoC，请参考5.3.11节；对于AOP，请参考5.3.12节。

**2、什么是设计模式？常用的设计模式有哪些？**

对于设计模式的理解，请参考第7章。