**[【计算机二级Java语言】卷005](https://www.cnblogs.com/mjios/p/12454967.html)**

**选择题**

**公共知识**

【1】下列叙述中正确的是()。

〖A〗线性表的链式存储结构与顺序存储结构所需要的存储空间是相同的

〖B〗线性表的链式存储结构所需要的存储空间一般要多于顺序存储结构

〖C〗线性表的链式存储结构所需要的存储空间一般要少于顺序存储结构

〖D〗线性表的链式存储结构与顺序存储结构在存储空间的需求上没有可比性

线性链式存储结构中每个结点都由数据域与指针域两部分组成, 增加了存储空间, 所以一般要多于顺序存储结构。

【2】下列叙述中正确的是()。

〖A〗栈是一种先进先出的线性表

〖B〗队列是一种后进先出的线性表

〖C〗栈与队列都是非线性结构

〖D〗以上三种说法都不对

栈是一种先进后出的线性表, 队列是一种先进先出的线性表, 栈与队列都是线性结构。

【3】软件测试的目的是()。

〖A〗评估软件可靠性

〖B〗发现并改正程序中的错误

〖C〗存改正程序中的错误

〖D〗发现程序中的错误

软件测试是为了发现错误而执行程序的过程, 测试要以查找错误为中心, 而不是为了演示软件的正确功能。  
不是为了评估软件或改正错误。

【4】在软件开发中, 需求分析阶段产生的主要文档是()。

〖A〗软件集成测试计划

〖B〗软件详细设计说明书

〖C〗用户手册

〖D〗软件需求规格说明书

需求分析阶段的工作可以概括为：需求获取、需求分析、编写需求规格说明书、需求评审四个方面。  
所以选择D。

【5】软件生命周期是指()。

〖A〗软件产品从提出、实现、使用维护到停止使用退役的过程

〖B〗软件从需求分析、设计、实现到测试完成的过程

〖C〗软件的开发过程

〖D〗软件的运行维护过程

通常, 将软件产品从提出、实现、使用维护到停止使用退役的过程称为软件生命周期。  
也就是说, 软件产品从考虑其概念开始, 到该软件产品不能使用为止的整个时期都属于软件生命周期。

【6】面向对象方法中, 继承是指()。

〖A〗一组对象所具有的相似性质

〖B〗一个对象具有另一个对象的性质

〖C〗各对象之间的共同性质

〖D〗类之间共享属性和操作的机制

继承是面向对象的方法的一个主要特征, 是使用已有的类的定义作为基础建立新类的定义技术。  
广义的说, 继承是指能够直接获得已有的性质和特征, 而不必重复定义它们, 所以说继承是指类之间共享属性和操作的机制。

【7】层次型、网状型和关系型数据库划分原则是()。

〖A〗记录长度

〖B〗文件的大小

〖C〗联系的复杂程度

〖D〗数据之间的联系方式

层次模型的基本结构是树形结构, 网状模型是一个不加任何条件限制的无向图, 关系模型采用二维表来表示, 所以三种数据库的划分原则是数据之间的联系方式。

【8】一个工作人员可以使用多台计算机, 而一台计算机可被多个人使用, 则实体工作人员与实体计算机之间的联系是()。

〖A〗一对一

〖B〗一对多

〖C〗多对多

〖D〗多对一

因为一个人可以操作多个计算机, 而一台计算机又可以被多个人使用, 所以两个实体之间是多对多的关系。

【9】数据库设计中反映用户对数据要求的模式是()。

〖A〗内模式

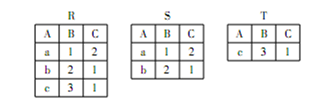
〖B〗概念模式

〖C〗外模式

〖D〗设计模式

数据库系统的三级模式是概念模式、外模式和内模式。  
概念模式是数据库系统中全局数据逻辑结构的描述, 是全体用户公共数据视图。  
外模式也称子模式或用户模式, 它是用户的数据视图, 给出了每个用户的局部数据描述, 所以选择C。  
内模式又称物理模式, 它给出了数据库物理存储结构与物理存取方法。

【10】有三个关系R、S和T如下：



则由关系R和S得到关系T的操作是()。

〖A〗自然连接

〖B〗差

〖C〗交

〖D〗并

关系T中的元组是R关系中有而S关系中没有的元组的集合, 所以进行的是差的运算。

**专业知识**

【11】下列关于Java语言特点的叙述中, 错误的是

〖A〗Java支持分布式计算

〖B〗Java是面向过程的编程语言

〖C〗Java是跨平台的编程语言

〖D〗Java支持多线程

Java是新一代编程语言, 具有很多特点。  
简单易学、利用面向对象技术、分布式计算、健壮性、安全性、跨平台(即体系结构中立)、可移植性、解释执行、高性能、多线程和动态性。  
本题答案为B选项。

【12】下列关于Java源程序结构的论述中, 正确的是

〖A〗一个文件包含的import 语句最多1个

〖B〗一个文件包含的public类最多1个

〖C〗一个文件包含的接口定义最多1个

〖D〗一个文件包含的类定义最多1个

一个文件的public类最多可以有一个, 但是可以包含多个import语句。  
包含的接口定义也可以有多个, 包含的class类可以有多个。  
本题答案为B选项。

【13】Java的反汇编命令是

〖A〗javap

〖B〗javac

〖C〗jdb

〖D〗java

Javap命令是java反汇编命令, javac命令是java语言编译器, jdb是基于文本和命令行的调试工具, java命令是Java解释器。  
选项A正确, 本题答案为A。

【14】下列语句执行后的输出结果是

复制代码

Java

|  |  |
| --- | --- |
|  | int i = 2, j = 6; |
|  | System.out.println((i + 5) > j); |

〖A〗0

〖B〗1

〖C〗true

〖D〗false

表达式"i+5=7"等于"7>6"返回值为true。  
本题答案为C选项。

【15】下列语句执行后的输出结果是

复制代码

Java

|  |  |
| --- | --- |
|  | int i = 2, j = 4; |
|  | boolean k; |
|  | k = (i++) > 0 && (++j - 4) < 0; |
|  | System.out.println(i \* j); |

〖A〗8

〖B〗10

〖C〗12

〖D〗15

后置"++"表示先参与运算再自身+1; 前置"++"表示先自身+1再参与运算。  
逻辑运算符"&&"在运算过程中会发生短路。  
当遇到false时, 后面的表达式不参与运算。  
在程序中, i++ = 2大于0, i自增1后等于3。  
因为表达式"( i++ ) > 0"为true, 继续运算逻辑运算符"&&"后面的式子, 此时j自增为5。  
题目要求打印"i\*j"表达式的值, 程序输出为15。  
本题答案为D选项。

【16】下列运算符中, 优先级最高的是

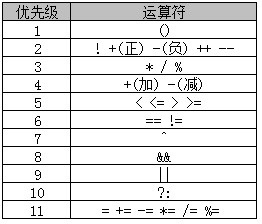
〖A〗+=

〖B〗==

〖C〗&&

〖D〗++

在Java中运算的优先级参照



。本题答案为D选项。

【17】下列程序的运行结果是

复制代码

Java

|  |  |
| --- | --- |
|  | public class Test { |
|  | public static void main(String[] args) { |
|  | int a = 3; |
|  | double b = 3.5; |
|  | int k = (int) b; |
|  | if (a == k) |
|  | System.out.println("0"); |
|  | else if (a > k) |
|  | System.out.println("1"); |
|  | else |
|  | System.out.println("2"); |
|  | } |
|  | } |

〖A〗程序不能运行

〖B〗0

〖C〗1

〖D〗2

double型数据强转为int型数据会丢失精度。  
变量b转为int类型结果为3, k = 3; 因为k = 3, a = 3, 所以表达式"a==k"为true, 执行if后面的语句, 输出0。  
本题答案为B选项。

【18】switch (expression) 语句中, 表达式expression不允许的类型是

〖A〗byte

〖B〗int

〖C〗boolean

〖D〗char

switch语句中表达式expression只能是下面这几种数据类型：int、byte、short和char。  
在JDK1.7之后还支持String类型。  
如果比较的数据类型是double和boolean型, 则不能使用 switch结构。  
选项C不满足条件。  
本题答案为C选项。

【19】下列程序的运行结果是

复制代码

Java

|  |  |
| --- | --- |
|  | public class Test { |
|  | public static void main(String[] args) { |
|  | int i = 1; |
|  | while (i < 7) { |
|  | if (i % 2 == 0) |
|  | System.out.print(i); |
|  | i++; |
|  | } |
|  | } |
|  | } |

〖A〗7

〖B〗135

〖C〗246

〖D〗247

运算符"%"为取余操作。  
本题中i % 2 == 0, i只要是偶数就会满足条件, 并打印i的值。  
while循环中, i的最大值为6, 有2, 4, 6三个偶数。  
本题答案为C选项。

【20】下列程序的运行结果是

复制代码

Java

|  |  |
| --- | --- |
|  | public class Test { |
|  | public static void main(String[] args) { |
|  | int count = 0; |
|  | for (int i = 1; i < 5; i += 2) |
|  | for (int j = 1; j <= 10; j += 3) |
|  | count++; |
|  | System.out.print(count); |
|  | } |
|  | } |

〖A〗0

〖B〗8

〖C〗10

〖D〗40

本题考查嵌套循环的应用, 第一层for循环执行2次, 第二层循环执行4次。  
内循环一共循环8次。  
每次循环自增变量count的值。  
循环结束后count的值为8。  
本题答案为B选项。

【21】继承是面向对象程序设计的一个重要特征, 它可降低程序的复杂性并使代码

〖A〗可读性好

〖B〗可重用

〖C〗可跨包访问

〖D〗运行更安全

继承性是面向对象方法的一个重要基本特性, 它使代码可重用, 可降低程序复杂性。  
继承是指在现有类(父类)的基础上构建一个新类(子类), 子类重用(继承)了父类的方法和状态, 同时还可以向新类中增添新的方法和状态。  
本题答案为B选项。

【22】阅读下列代码段

复制代码

Java

|  |  |
| --- | --- |
|  | abstract class Person { |
|  | public Person(String n) { |
|  | name = n; |
|  | } |
|  | public \_\_\_\_\_\_\_\_ String getDescription(); |
|  | public String getName() { |
|  | return name; |
|  | } |
|  | private String name; |
|  | } |

在下划线处应填入的修饰符是

〖A〗static

〖B〗abstract

〖C〗protected

〖D〗final

getDescription()只有方法名, 没有方法体, 属于抽象方法。  
抽象方法用关键字abstract修饰。  
本题答案为B选项。

【23】Java中所有类的父类是

〖A〗Father

〖B〗Lang

〖C〗Exception

〖D〗Object

java中, Object类是所有类的父类。  
本题答案为D选项。

【24】下列程序的运行结果

复制代码

Java

|  |  |
| --- | --- |
|  | public class Test { |
|  | String s = "One World One Dream"; |
|  | public static void main(String[] args) { |
|  | System.out.println(s); |
|  | } |
|  | } |

〖A〗args

〖B〗One World One Dream

〖C〗s

〖D〗编译时出错

s不属于静态变量, 访问时不能直接调用, 需用对象的引用进行访问, 程序编译时出错。  
本题答案为D选项。

【25】用于在子类中调用被重写父类方法的关键字是

〖A〗this

〖B〗super

〖C〗This

〖D〗Super

this调用本类相关的构造函数, super调用父类相关构造方法。  
本题答案为B选项。

【26】阅读下列程序片段

复制代码

Java

|  |  |
| --- | --- |
|  | public void test() { |
|  | try { |
|  | sayHello(); |
|  | System.out.println("Hello"); |
|  | } catch (ArrayIndexOutOfBoundsException e) { |
|  | System.out.println("ArrayIndexOutOfBoundsException "); |
|  | } catch (Exception e) { |
|  | System.out.println("Exception "); |
|  | } finally { |
|  | System.out.println("finally"); |
|  | } |
|  | } |

如果sayHello()方法正常运行, 则test()方法的运行结果将是

〖A〗Hello finally

〖B〗ArrayIndexOutOfBoundsException

〖C〗Exception finally

〖D〗Hello

sayHello()方法正常运行, try模块输出"Hello", 接着执行finally模块, 无论异常是否存在, 此模块都会执行, 输出"finally"。  
本题答案为A选项。

【27】下列程序的运行结果

复制代码

Java

|  |  |
| --- | --- |
|  | public class ConcatTest { |
|  | public static void main(String[] args) { |
|  | String str1 = "abc"; |
|  | String str2 = "ABC"; |
|  | String str3 = str1.concat(str2); |
|  | System.out.println(str3); |
|  | } |
|  | } |

〖A〗abc

〖B〗ABC

〖C〗abcABC

〖D〗ABCabc

本题考查字符串的使用。  
String类提供concat()方法, 该方法将当前字符串对象与指定str字符串相连。  
表达式str1.concat(str2)的结果为"abcABC"。  
本题的答案为C选项。

【28】下列程序的运行结果

复制代码

Java

|  |  |
| --- | --- |
|  | public class Arrays { |
|  | public static void main(String[] args) { |
|  | int[] a = new int[5]; |
|  | for (int i = 0; i < a.length; i = i + 1) { |
|  | a[i] = 10 + i; |
|  | } |
|  | for (int i = 0; i < a.length; i = i + 1) { |
|  | System.out.print(a[i] + " "); |
|  | } |
|  | String[] s = { "Frank", "Bob", "Jim" }; |
|  | for (int i = 0; i < s.length; i = i + 1) { |
|  | System.out.print(s[i] + " "); |
|  | } |
|  | s[2] = "Mike"; |
|  | System.out.print(s[2]); |
|  | } |
|  | } |

〖A〗10 11 12 13 14 Mike Bob Frank Jim

〖B〗11 12 13 14 15 Frank Bob Mike Jim

〖C〗10 11 12 13 14 Frank Bob Jim Mike

〖D〗11 12 13 14 15 Mike Jim Bob Frank

第一个for循环对整型数组a赋值时从"10+0"开始, 直到"10+4", 共5个数。  
第二个for循环直接将字符串数组内容依次打印出来, 循环结束后对s[2]重新赋值为"Mike", 故再次打印s[2]时输出"Mike"。  
本题答案为C选项。

【29】下列代码的功能是向文件score.dat中写入对象。在下划线处应填入的是

复制代码

Java

|  |  |
| --- | --- |
|  | ObjectOutputStream out = new ObjectOutputStream(new \_\_\_\_\_\_\_\_\_\_\_\_("score.dat")); |
|  | out.writeObject(fenshu); |
|  | out.close(); |

〖A〗ObjectOutputStream

〖B〗ObjectInputStream

〖C〗FileOutputStream

〖D〗FileInputStream

类ObjeetOutputStream是将一个对象写到一个流中, 其原型为public ObjectOutputStream(OutputStream out), 即参数为一个输出流对象。  
显然, 在空白处新建了一个输出流的对象。  
"score.dat"是一个文件名。  
本题的意思为将对象写入一个文件输出流。  
public FileOutputStream(String name), 创建一个输出文件流。  
本体答案为C选项。

【30】Java中的抽象类Reader和Writer所处理的流是

〖A〗图像流

〖B〗对象流

〖C〗字节流

〖D〗字符流

Reader / Writer所处理的流是字符流, Inputstream / OutputStream的处理对象是字节流。  
本题答案为D选项。

【31】下列不属于接口WindowListener的方法是

〖A〗windowClosing()

〖B〗windowClosed()

〖C〗windowMinimized()

〖D〗windowOpened()

接口WindowListener包括以下方法：windowActivated、windowDeactivated、windowClosing、windowClosed、windowIconified、windowDeiconified、windowOpened方法。  
本题答案为C选项。

【32】在下列程序的空白处, 应填入的正确选项是

复制代码

Java

|  |  |
| --- | --- |
|  | import java.io.\*; |
|  | public class ObjectStreamTest { |
|  | public static void main(String[] args)throws IOException { |
|  | ObjectOutputStream oos = new ObjectOutputStream(new FileOutputStream("serial.bin")); |
|  | java.util.Date d = new java.util.Date(); |
|  | oos.\_\_\_\_\_\_\_\_\_\_\_\_(d); |
|  | ObjectInputStream ois = new ObjectInputStream(new FileInputStream("serial.bin")); |
|  | try { |
|  | java.util.Date restoredDate = (java.util.Date)ois.readObject(); |
|  | System.out.println("read Object back from serial.bin file: " + restoredDate); |
|  | } catch (ClassNotFoundException cnf) { |
|  | System.out.println("class not found"); |
|  | } |
|  | } |
|  | } |

〖A〗WriteObject

〖B〗write

〖C〗BufferedWriter

〖D〗writeObject

本题考查的是输入/输出及文件操作, writeObject()方法直接写入对象。  
本题答案为D选项。

【33】下列包中, 包含JOptionPane类的是

〖A〗javax.swing

〖B〗java.lang

〖C〗java.util

〖D〗java.applet

Swing中提供了J0ptionPane类来实现类似Windows平台下的MessageBox的功能。  
利用JOptionPane类中的各个static方法来生成各种标准的对话框, 实现显示信息、提出问题、警告、用户输入参数等功能, 且这些对话框都是模式对话框。  
本题答案为A选项。

【34】下列程序中需要对Employee的对象进行存储, 下划线处应填入的正确选项是

复制代码

Java

|  |  |
| --- | --- |
|  | class Employee implements \_\_\_\_\_\_\_\_\_\_ { |
|  | …… |
|  | } |

〖A〗Comparable

〖B〗Serializable

〖C〗Cloneable

〖D〗DataInput

本题考查对象的序列化。  
如果想保存程序中的对象, 以便在该程序的后运行中使用, 可以使用对象的序列化。  
一个类只有实现了Serialiazable接口, 才是可序列化的。  
如果要序列化某些类的对象, 这些类就必须实现 Serialiazable接口。  
本题的答案为B选项。

【35】在下列Java语言的包中, 提供图形界面构件的包是

〖A〗java.io

〖B〗javax.swing

〖C〗java.net

〖D〗java.rmi

java.io包向程序提供文件输入/输出操作的类; javax.swing包提供构建和管理应用程序的图形界面的轻量级构件; java.net包向程序提供执行网络通信应用及URL处理的类; java.rmi包向程序提供远程方法调用所需的类。  
本题答案为B选项。

【36】Java API中支持线程的类或接口是  
I.java.lang.Thread  
II.java.lang.Runnable  
III.java.lang.ThreadGroup  
IV.java.io.Serializable

〖A〗I, II

〖B〗I, II, III

〖C〗I, II, IV

〖D〗I, II, III, IV

java.lang.Thread类和java.lang.Runnable是创建线程的两个方法, 分别是实现Thread类和继承Runnable接口。  
ThreadGroup类是管理一组线程的类。  
Serializable是序列化, 将一个对象的状态保存起来, 在适当的时候再获得, 它不支持线程。  
本题答案为B选项。

【37】阅读下面程序

复制代码

Java

|  |  |
| --- | --- |
|  | public class Test2 \_\_\_\_\_\_\_\_\_\_\_ { |
|  | public static void main(String[] args) { |
|  | Thread t = new Test2(); |
|  | t.start(); |
|  | } |
|  | public void run() { |
|  | System.out.println("How are you."); |
|  | } |
|  | } |

在程序下划线处填入正确选项是

〖A〗extends Thread

〖B〗extends Runnable

〖C〗implements Runnable

〖D〗implements Thread

创建线程有两种方法：实现java.lang.Runnable接口和继承Thread类并重写run()方法。  
本程序通过Thread t = new Test2(); 可以看出是通过继承Thread类来创建进程。  
本题答案为A选项。

【38】请阅读下列程序

复制代码

Java

|  |  |
| --- | --- |
|  | public class ThreadTest { |
|  | public static void main(String[] args)throws Exception { |
|  | int i = 0; |
|  | Hello t = new Hello(); |
|  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; |
|  | while (true) { |
|  | System.out.println("Good Morning" + i++); |
|  | if (i == 2 && t.isAlive()) { |
|  | System.out.println("Main waiting for Hello!"); |
|  | t.join(); |
|  | //等待t运行结束 |
|  | } |
|  | if (i == 5) |
|  | break; |
|  | } |
|  | } |
|  | } |
|  | class Hello extends Thread { |
|  | int i; |
|  | public void run() { |
|  | while (true) { |
|  | System.out.println("Hello" + i++); |
|  | if (i == 5) |
|  | break; |
|  | } |
|  | } |
|  | } |

为使该程序正确执行, 下划线处的语句应是

〖A〗t.sleep()

〖B〗t.yield()

〖C〗t.interrupt()

〖D〗t.start()

启动线程用start()方法。  
本题答案为D选项。

【39】在多线程并发程序设计中, 能够给对象x加锁的语句是

〖A〗x.wait()

〖B〗x.notify()

〖C〗synchronized(x)

〖D〗x.synchronized()

Java平台将每个由synchronized(Object)语句指定的对象设置一个锁, 称为对象锁。  
Java中的对象锁是一种独占的排他锁。  
本题答案为C选项。

【40】下列叙述中, 错误的是

〖A〗Applet事件处理机制与Java Application相同

〖B〗Applet事件处理机制与JApplet相同

〖C〗Applet事件处理机制采用监听器方式

〖D〗JApplet事件处理机制不是采用监听器方式

Swing的事件处理机制继续沿用AWT的事件处理机制。  
事件处理机制中仍旧包含三种角色：事件源、事件和事件处理者(事件监听程序)。  
事件源就是Swing的各种构件, 与之对应的就是事件监听器接口。  
选项D说法错误, 本题答案为D选项。

**编程题**

[编程题代码下载](https://github.com/CoderMJLee/NCRE)

【41】在考生文件夹中存有文件名为Java\_1.java的文件, 该程序是不完整的, 请在注释行"//\*\*\*\*\*\*\*\*\*\*Found\*\*\*\*\*\*\*\*\*\*"下一行语句的下划线地方填入正确内容, 然后删除下划线, 请勿删除注释行或改动其他已有语句内容。存盘时文件必须存放在考生文件夹下, 不得改变原有文件的文件名。  
程序的功能是在给定球的半径时计算出球的体积。当程序运行时, 在命令屏幕上给出程序功能提示, 并显示输入半径的窗口：



根据考生输入球半径的值计算体积, 并在命令屏幕上显示结果。

复制代码

Java

|  |  |
| --- | --- |
|  | (1) javax |
|  | (2) static |
|  | (3) input |

本题考查：java包的导入、main()函数语法以及数据类型转换。  
(1) swing是java的扩展程序包之一, javax中存储的都是扩展程序包。  
(2) java中程序的main方法为：public static void main(String[] args) { } 。  
(3) 将字符串类型转为double类型, Double引用类型提供parseDouble(String str)方法, 本程序要求在窗口中输入的字符串转为double类型。

【42】在考生文件夹中存有文件名为Java\_2.java的文件, 该程序是不完整的, 请在注释行"//\*\*\*\*\*\*\*\*\*\*Found\*\*\*\*\*\*\*\*\*\*"下一行语句的下划线地方填入正确内容, 然后删除下划线, 请勿删除注释行或改动其他已有语句内容。存盘时文件必须存放在考生文件夹下, 不得改变原有文件的文件名。  
程序的功能是：定义了一个简单的线程, 可以对其名称进行设置。该程序运行时的输出结果如下：  
Running the Testing\_Thread:  
-- - 0-- - Testing\_Thread  
-- - 1-- - Testing\_Thread  
-- - 2-- - Testing\_Thread  
-- - 3-- - Testing\_Thread  
-- - 4-- - Testing\_Thread

复制代码

Java

|  |  |
| --- | --- |
|  | (1) t.start() |
|  | (2) Thread |
|  | (3) setName(str) |
|  | (4) run |
|  | (5) catch |

本题考查：线程类的定义、线程的启动运行以及异常的捕获。  
(1) java中启动线程用start()方法, 根据本程序中的 Thread t = new SimpleThread("Testing\_Thread"); 启动线程用t.start()。  
(2) 实现线程的方式有两种：继承Thread类、实现Runnable接口。  
这里是继承Thread类。  
(3) 设置线程名的方法为：setName(String str), 将线程名设置为想要的名称。  
(4) 覆写run()方法, 作为线程 的操作主体 。  
(5) 处理异常的方式, 程序中处理方式为try catch语句, catch中处理捕获的异常。

【43】在考生文件夹中存有文件名为Java\_3.java的文件, 该程序是不完整的, 请在注释行"//\*\*\*\*\*\*\*\*\*\*Found\*\*\*\*\*\*\*\*\*\*"下一行语句的下划线地方填入正确内容, 然后删除下划线, 请勿删除注释行或改动其他已有语句内容。存盘时文件必须存放在考生文件夹下, 不得改变原有文件的文件名。  
本题的要求是：  
在JFrame窗口中, 显示出红色的2008北京奥运主题口号"同一个世界,同一个梦想", 同时在窗口的上方, 有三个按钮可以改变窗口的背景色。  
该程序的运行结果如下：



复制代码

Java

(1)JFrame(2)panel(3)implements(4)actionPerformed

本题考查：图形界面组件以及监听器的使用。  
(1) 程序要求在JFrame窗口中, 显示出红色的2008北京奥运主题口号"同一个世界,同一个梦想", 所以FontFrame类应继承JFrame类。  
(2) 通过add()方法将面板对象添加到容器中。  
(3) ActionListener是Java中关于事件处理的一个接口, 继承自EventListener。  
所以操作时应实现该接口, java中实现接口用关键字implements。  
(4) 实现ActionListener接口需实现里面的actionPerformed()抽象方法。