



类 Test1 定义如下:

```
1 public class Test1{//1
2     public float aMethod(float a,float b) {}//2
3     //3
4 }//4
```

将以下哪种方法插入行 3 是不合法的。

正确答案: C 你的答案: 空 (错误)

```
public int aMethod(int a,int b){}
private float aMethod(int a,int b,int c){}
public float aMethod(float a,float b){}
public float aMethod(float a,float b,float c){}
```

以下关于 java 封装的描述中, 正确的是:

正确答案: A 你的答案: 空 (错误)

封装的主要作用在于对外隐藏内部实现细节, 增强程序的安全性

封装的意义不大, 因此在编码中尽量不要使用

如果子类继承父类, 对于父类中进行封装的方法, 子类仍然可以直接调用

只能对一个类中的方法进行封装, 不能对属性进行封装

以下程序执行的结果是:

```
1 class X{
2     Y y=new Y();
3     public X(){
4         System.out.print("X");
5     }
6 }
7 class Y{
8     public Y(){
9         System.out.print("Y");
10    }
11 }
12 public class Z extends X{
13     Y y=new Y();
14     public Z(){
15         System.out.print("Z");
16     }
17     public static void main(String[] args) {
18         new Z();
19     }
20 }
```

正确答案: C 你的答案: 空 (错误)





ZYXX

ZYXY

YXYZ

XYZX

`instanceof` 运算符能够用来判断一个对象是否为:

正确答案: C 你的答案: 空 (错误)

一个类的实例

一个实现指定接口的类的实例

全部正确

一个子类的实例

如果一个方法或变量是"`private`"访问级别, 那么它的访问范围是:

正确答案: D 你的答案: 空 (错误)

在当前类, 或者子类中

在当前类或者它的父类中

在当前类, 或者它所有的父类中

在当前类中

`init` 启动进程需要读取()配置文件

正确答案: D 你的答案: 空 (错误)

/sbin/init

/bin/sh

/etc/sysvinit

/etc/inittab

有关线程的叙述正确的是()

正确答案: C D 你的答案: 空 (错误)

可以获得对任何对象的互斥锁定

通过继承 `Thread` 类或实现 `Runnable` 接口, 可以获得对类中方法的互斥锁定

线程通过调用对象的 `synchronized` 方法可获得对象的互斥锁定

线程调度算法是平台独立的

在重新启动 `Linux` 系统的同时把内存中的信息写入硬盘, 应使用 ( ) 命令实现

正确答案: A 你的答案: 空 (错误)

`#shutdown -r now`

`#halt`

`#reboot`

`#init3`





实现单例模式的类具有?

正确答案: A 你的答案: 空 (错误)

- 没有公有构造方法, 一个公有静态工厂方法, 和一个静态实例变量
- 没有公有构造方法, 有一个私有静态工厂方法, 和一个静态实例变量
- 有公有构造方法, 没有静态工厂方法和静态实例变量
- 没有公有构造方法, 没有公有静态工厂方法, 也没有静态实例变量

哪个变量用来指定一个远程 X 应用程序将输出放到哪个 X server 上

正确答案: B 你的答案: 空 (错误)

- TERM
- DISPLAY
- ECHO
- OUTPUT

接口的作用是;

正确答案: B 你的答案: 空 (错误)

- 减小堆大小
- 契约式设计以封装实现
- 实现抽象工厂模式
- 使抽象类具化

给定代码:

```
1 public class SwitchTest{//1
2     public static void main(String[] args) {//2
3         System.out.println("value="+switchit(4));//3
4     }//4
5     public static int switchit(int x) {
6         int j=1;
7         switch (x) {
8             case 1:j++;
9             case 2:j++;
10            case 3:j++;
11            case 4:j++;
12            case 5:j++;
13            default:j++;
14        }
15        return j+x;
16    }
17 }
```

第三行将输出什么?

正确答案: B 你的答案: 空 (错误)





```
value=6
value=8
value=3
value=5
value=4
```

有以下类定义:

```
1  abstract class Animal{
2      abstract void say();
3  }
4  public class Cat extends Animal{
5      public Cat() {
6          System.out.printf("I am a cat");
7      }
8      public static void main(String[] args) {
9          Cat cat=new Cat();
10     }
11 }
```

运行后:

正确答案: B 你的答案: 空 (错误)

I am a cat

Animal 能编译, Cat 不能编译

Animal 不能编译, Cat 能编译

编译能通过, 但是没有输出结果

以下代码输出的是:

```
1  public class SendValue{
2      public String str="6";
3      public static void main(String[] args) {
4          SendValue sv=new SendValue();
5          sv.change(sv.str);
6          System.out.println(sv.str);
7      }
8      public void change(String str) {
9          str="10";
10     }
11 }
```

正确答案: A 你的答案: 空 (错误)

6  
10  
都不对





16

以下代码在编译和运行过程中会出现什么情况

```
1 public class TestDemo{
2     private int count;
3     public static void main(String[] args) {
4         TestDemo test=new TestDemo(88);
5         System.out.println(test.count);
6     }
7     TestDemo(int a) {
8         count=a;
9     }
10 }
```

正确答案: A 你的答案: 空 (错误)

编译运行通过，输出结果是 88

编译时错误，count 变量定义的是私有变量

编译时错误，System.out.println 方法被调用时 test 没有被初始化

编译和执行时没有输出结果

