

1、下列关于构造方法的叙述中，错误的是（ ）

- A、Java 语言规定构造方法名与类名必须相同
- B、Java 语言规定构造方法没有返回值，但不用 void 声明
- C、Java 语言规定构造方法不可以重载
- D、Java 语言规定构造方法通常通过 new 自动调用

正确答案：C

扩展：

构造方法：用于完成对象属性的初始化工作，有如下特点：

- （1）构造方法的名字必须和类名完全一样（包括大小写）；
- （2）构造方法没有返回值，连 void 也不会出现；
- （3）如果认为没有给类创建一个构造方法，java 编译器会自动给我们添加一个没有参数且方法体为空的构造方法。当然，如果已经人为的添加了一个构造方法，java 编译器就不会给添加这个构造方法了。
- （4）我们不能显式调用类的构造方法，构造方法通常只是通过 new 关键字隐式调用；

2、以下关于 final 的描述中，错误的是（ ）

- A、final 可以用来修饰接口以及抽象类
- B、final 修饰的变量不允许被再次赋值
- C、final 修饰的类无法被继承
- D、final 修饰的变量需要初始化

正确答案：A

扩展：

final 不能修饰抽象类和接口。很明显，C 选项说了 final 修饰的类无法被继承，修饰抽象类和接口毫无意义。

final 关键字可以用来修饰类、方法、变量、参数。

一，修饰类

- 1、该类不能被继承。
- 2、类中的方法不会被覆盖，因此默认都是 final 的。

二，修饰方法

- 1、该方法可以被继承，但是不能被覆盖。
- 2、好处：可以防止继承它的子类修改该方法的意义和实现；更为高效，编译器在遇到调用 final 方法转入内嵌机制，提高了执行效率。
- 3、注意：父类中的 private 成员方法不能被子类覆盖，因此，private 方法默认是 final 型的。

三，修饰变量

- 1、用 final 修饰后变为常量。包括静态变量、实例变量和局部变量这三种。
- 2、特点：可以先声明，不给初值，这种叫做 final 空白。但是使用前必须被初始化。一旦被赋值，将不能再被改变。

四，修饰参数

- 1、用 final 修饰参数时，可以读取该参数，但是不能对其作出修改

3、以下关于被 private 修饰的成员变量的说法正确的是（ ）

- A、只能被同一个包中的类访问
- B、只能被该类自身所访问和修改
- C、可以被两种类访问和引用：该类本身、该类的所有子类
- D、可以被三种类所引用：该类自身、与它在同一个包中的其他类、在其他包中的该类的子类

正确答案：B

扩展：

public：

具有最大的访问权限，可以访问任何一个在 classpath 下的类、接口、异常等。它往往用于对外的情况，也就是对象或类对外的一种接口的形式。

protected：

主要的作用就是用来保护子类的。它的含义在于子类可以用它修饰的成员，其他的不可以，它相当于传递给子类的一种继承的东西

default：

有时候也称为 friendly，它是针对本包访问而设计的，任何处于本包下的类、接口、异常等，都可以相互访问，即使是父类没有用 protected 修饰的成员也可以。

private：

访问权限仅限于类的内部，是一种封装的体现，例如，大多数成员变量都是修饰符为 private 的，它们不希望被其他任何外部的类访问。

4、什么是 `LinkedHashSet` ()

- A、是一个 `HashSet`，保留了对对象插入的顺序
- B、是一个 `HashSet`，拥有一个链表的性能
- C、是一个 `HashSet`，能很容易地转换为一个链表
- D、`HashSet` 对象的超类

正确答案：A

扩展：

`LinkedHashSet` 是 `HashSet` 的一个“扩展版本”，`HashSet` 并不管什么顺序，不同的是 `LinkedHashSet` 会维护“插入顺序”。`HashSet` 内部使用 `HashMap` 对象来存储它的元素，而 `LinkedHashSet` 内部使用 `LinkedHashMap` 对象来存储和处理它的元素。

5、关于以下程序代码的说明正确的是 ()

```
public class HasStatic {  
    private static int x = 100;  
  
    public static void main(String args[]) {  
        HasStatic hs1 = new HasStatic();  
        hs1.x++;  
        HasStatic hs2 = new HasStatic();  
        hs2.x++;  
        hs1 = new HasStatic();  
        hs1.x++;  
        HasStatic.x--;  
        System.out.println("x=" + x);  
    }  
}
```

- A、5 行不能通过编译，因为引用了私有静态变量
- B、10 行不能通过编译，因为 `x` 是私有静态变量
- C、程序通过编译，输出结果为：`x=103`
- D、程序通过编译，输出结果为：`x=102`

正确答案：D