1. 回调函数：老板A，让员工B做一件复杂的事，老板去做其他事，员工做完报告老板。

A{

askquestion{ new treard{ B.work()} }

tellmeAnswer{ get answer, do something}(一般这里是个接口)

}

B{

work{ work on the question; finish; A.tellmeAnswer() }

}

main{

A.askquestion(B)

}

1. 面向对象的特征有哪些方面？

- 抽象：抽象是将一类对象的共同特征总结出来构造类的过程.

- 继承(extends)：继承是从已有类得到继承信息创建新类的过程。

- 封装：封装就是隐藏一切可隐藏的东西，只向外界提供最简单的编程接口。

- 多态: 不同子类型的对象对同一消息作出不同的响应。

重写（override）一个引用在不同情况下的多种状态，不同子类重写了父类的不同方法。

Animal an = new Cat()

an.cry()

an = new Dog()

an.cry()

重载（overloading）一个类多个方法，参数不同。一个类本身的多态性表现

Animal an = new Animal()

public void eat(int a){}

public void eat(int a, string b){}

1. 访问修饰符public,private,protected

修饰符 当前类 子类 其他包

public √ √ √

protect √ √ ×

private √ × ×

1. float f=3.4;是否正确？

答:不正确。float f =(float)3.4; 或者写成float f =3.4F;。

1. Java有没有goto？

答：目前版本的Java中没有使用。

1. int和Integer有什么区别？

答 Integer是int的包装类，自动装箱/拆箱机制，使得二者可以相互转换。

- 原始类型: boolean，char，byte，short，int，long，float，double

- 包装类型：Boolean，Character，Byte，Short，Integer，Long，Float，Double

1. Math.round(11.5) 等于多少？Math.round(-11.5)等于多少？

Math.round(11.5)的返回值是12，Math.round(-11.5)的返回值是-11。

1. 数组

int intArray[] = new int[5];

intArray[i];

数组有没有length()方法？String有没有length()方法？

数组没有length()方法，有length 的属性。String 有length()方法。

1. 在Java中，如何跳出当前的多重嵌套循环？

方法一： 利用标签

tag:for(int i=0;i<10;i++){

for(int j=0;j<10;j++){

System.out.println("i="+i+",j="+j);

break tag;//跳到循环外的tag出，即终止整个循环 } }

方法二： 增加循环条件判断

for(int i=0;i<arr.length && !found;i++){

for(int j=0;j<arr[i].length;j++){

break;//跳出当前循环 } }

1. 构造器（constructor）是否可被重写（override）？

构造器不能被继承，因此不能被重写，但可以被重载（多参数）。

1. 是否可以继承String类？

String 类是final类，不可以被继承。

1. 到底是值传递还是引用传递？

Java的参数传递都是值传递！！！ （除非是数组或者是类 传的地址）

1. string 的比较

String s1 = new String("java");

String s2 = new String("java");

System.out.println(s1==s2); //false

System.out.println(s1.equals(s2)); //true

\*\*\*

String s1 = new String("java");

String s2 = s1;

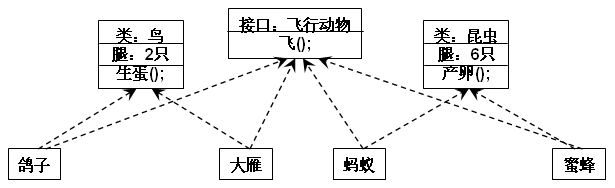
System.out.println(s1==s2); //true

System.out.println(s1.equals(s2)); //true

String s = new String(“xyz”);创建了几个字符串对象？

答：两个对象，一个是静态区的”xyz”，一个是用new创建在堆上的对象。

1. 接口 和 抽象类



在类的继承中，只能做单重继承，而实现接口时，一次则可以实现多个接口。

接口就是函数声明，类就是函数实现

接口有点象一个规范、一个协议，是一个抽象的概念；而类则是实现了这个协议，满足了这个规范的具体实体，是一个具体的概念。

抽象类的成员可以具有访问级别， 接口的成员全部public级别

在类的继承中，只能做单重继承，而实现接口时，一次则可以实现多个接口。

抽象的子类可以选择性实现， 接口的子类必须实现

接口是抽象类的一种特例！

1. 一个”.java”源文件中是否可以包含多个类（不是内部类）？有什么限制？

答：可以，但一个源文件中最多只能有一个公开类（public class）而且文件名必须和公开类的类名完全保持一致。

1. 静态方法 静态类

静态方法：直接类名调用 只能访问静态变量

静态类： main就是一个静态类

1. 内部类可以引用它的包含类（外部类）的成员吗？有没有什么限制？

一个内部类对象可以访问创建它的外部类对象的成员，包括私有成员。（Outer.this.\*\*\*）

1. final关键字有哪些用法？

(1)修饰类：表示该类不能被继承；

(2)修饰方法：表示方法不能被重写；

(3)修饰变量：表示变量只能一次赋值以后值不能被修改（常量）。

1. 什么是线程安全？

如果你的代码所在的进程中有多个线程在同时运行，而这些线程可能会同时运行这段代码。如果每次运行结果和单线程运行的结果是一样的，而且其他的变量的值也和预期的是一样的，就是线程安全的

1. Java写一个单例类 (打印机)

①将构造器私有，不允许外界通过构造器创建对象；

②通过公开的静态方法向外界返回类的唯一实例。

懒汉式单例 ()比较懒，只有当调用getInstance的时候，才回去初始化这个单例。它是线程不安全的，并发环境下很可能出现多个Singleton实例

public class Singleton {

private static Singleton instance = null;

private Singleton() {}

public static synchronized Singleton getInstance(){

if (instance == null) instance ＝ new Singleton();

return instance;

}

}

饿汉式单例(饿汉就是类一旦加载，就把单例初始化完成,保证getInstance的时候，单例是已经存在的了) 天生是线程安全的。

public class Singleton {

private Singleton(){}

private static Singleton instance = new Singleton();

public static Singleton getInstance(){

return instance;

}

}

1. java反射

反射就是一种机制，可以让你仅知道类的名字的情况下，可以了解整个类的内部的结构，并且访问内部的成员和方法等。

类是程序的一部分，每个类都有一个Class对象。每当编写并且编译了一个新类，就会产生一个Class对象。Class是Reflection的起源。

Class的方法

getName()：获得类的完整名字。 getFields()：获得类的public类型的属性。 等

1.getName

Foo f = new Foo();

System.out.println(f.getClass().getName());

2.调用未知对象的方法，不知道一个对象的类型，找到这个对象是否有个方法名叫print并且调用它

try {

Method method = f.getClass().getMethod("print", new Class<!--?-->[0]);

method.invoke(f);

} catch (Exception e) {

e.printStackTrace();

}

3.可以修改访问检查（field） 绕过private protect，访问。 从Class实例创建对象。

4. runtime需动态调整数组长度大小，又在调整的代码处只能确定它是个数组，不能确定类型等

public static Object changeArraySize(Object obj, int len) {

Class<?> arr = obj.getClass().getComponentType(); ？这里可以确定数组类型 int？ string？假设是int

Object newArray = Array.newInstance(arr, len); 新建一个int数组， 增加长度

int co = Array.getLength(obj); 原数组的长度

System.arraycopy(obj, 0, newArray, 0, co); copy原数组的内容到新数组

return newArray;

}

1. 创建对象时构造器的调用顺序是：

先初始化静态成员 static int a

然后调用父类构造器 super（）{}，

再初始化非静态成员 int b，

最后调用自身构造器 {}。

1. 日期和时间

java.util.Calendar

1. Error 和Exception 有什么区别?

Error 表示系统级的错误和程序不必处理的异常 （内存溢出）

Exception 表示需要捕捉程序进行处理的异常，是一种设计或实现问题；

1. try{}里有一个return语句，

那么紧跟在这个try后的finally{}里的code会不会被执行，什么时候被执行，在return前还是后?

会执行，在方法返回调用者前执行。

1. 不是java基础类型

String为引用类型。

1. 装箱、拆箱操作发生在:

引用类型与值类型之间

装箱——值类型转换为引用类型；拆箱——引用类型转换为值类型。

1. 数组定义后，默认元素为0
2. Java 多线程有几种实现方法

继承Thread类

实现Runnable接口

1. Java的跨平台特性是因为JVM的存在， 它可以执行.class字节码文件,不同平台开发出不同jvm（java虚拟机）
2. int[] x=new int[25];

若访问x[25]，程序将抛出异常。(java中没有'\0'这一说)

1. public > protected > 同包（default） > private
2. 实现接口的类（除抽象类），必须实现接口中所有方法，且不能降低方法的运用域，即必须显示声明为public
3. 要在session对象中保存属性

session.setAttribute(“key”，”value”)

1. 方法是可以和类名同名的，和构造方法唯一的区别就是，构造方法没有返回值。
2. 要导入java/awt/event下面的所有类

只能是import java.awt.event.\*

不可以import java.awt.\*

1. int a=10 ; double b=3.14 ; 则表达式 'A'+a+b 值的类型是（double ）

char < short < int < float < double 不同类型运算结果类型向右边靠齐。

1. Thread类包含start()和run()方法，有些人以为这两个方法作用一致，那就就大错特错了

start() 方法的作用是启动一个新线程，新线程会执行相应的run()方法，start()不能被重复调用。

而run()方法则只是普通的方法调用，在调用线程中顺序运行而已。

run()方法用来执行线程体中具体的内容

start()方法用来启动线程对象，使其进入就绪状态

sleep()方法用来使线程进入睡眠状态

suspend()方法用来使线程挂起，要通过resume()方法使其重新启动

WWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW