**Java笔试题**

**-----------------------------------------Part I ，总计70分----------------------------------------------------**

1. **单选题（共12题，每题2分，共24分）**

1．下面关于Java的表述有错误的是\_\_\_\_\_\_\_\_

A．Java中没有sizeof操作符，所有数据类型的长度都是确定的，与平台无关。

B．在Java中，局部变量不会被自动初始化，需要显式的赋初值。

C．Java中的instanceof运算符，可以判断一个对象是否属于某个类的实例。

D．一个Java文件里可以有多个public类，但Java文件名必须和包含main方法的那个public类名精确一致。

2．下面关于Java中static关键字的表述错误的是\_\_\_\_\_\_\_\_

A．被static修饰的类成员变量在编译时由内存分配一块空间，直到程序停止运行才会被释放。

B．被static修饰的类方法只能访问其自身类未被static修饰的实例方法，而不能访问其他类的。

C．被static修饰的类方法不能使用this关键字。

D．在没有创建该类的对象时，该类中用static修饰的类方法就可以被调用。

3．下面关于Java中abstract关键字的表述有错误的是\_\_\_\_\_\_\_\_

A．用abstract修饰不在类内实现的方法，称为抽象方法；抽象方法必须在子类派生时被重载。

B．用abstract修饰的类为抽象类；抽象类可以同时被static或final关键字修饰。

C．抽象类中不一定包含抽象方法，但包含抽象方法的类一定要声明为抽象类。

D．如果一个类被定义为抽象类，则该类不能进行实例化。

4．下面关于Java中垃圾回收（gc）的表述有错误的是\_\_\_\_\_\_\_

A．在String str1 = "Hello";String str2 = str1;str1 = null;中，对象"Hello"不会被垃圾回收。

B．调用Runtime.getRuntime().gc()会立即启动垃圾回收线程对当前程序进行垃圾回收。

C．调用Systmer.gc()只是提示垃圾回收器进行垃圾回收操作，但并不保证gc线程立即启动。

D．当垃圾回收器将要释放无用对象的内存时，先调用该对象的finalize()方法。

5．下面关于Java中接口(interface)的表述有错误的是\_\_\_\_\_\_\_\_

A．在接口中定义的常量可以被多个类共享，具有public,final,static属性，可被任意类访问。

B．接口中只对方法进行声明而没有实现体，而且用分号结尾，具有public和abstract属性

C．子接口中定义了与父接口同名的常量或者相同的方法，则父接口的常量被隐藏，方法被重写

D．实现接口的类必须实现接口和其父接口中的所有方法。

6．下面哪个方法不属于String类提供的\_\_\_\_\_\_

A．IndexOf()

B．getChar2()

C．replace()

D．append()

7．在Java多线程中，调用下面哪种方法不会使线程进入阻塞状态？\_\_\_\_\_\_\_\_

A．sleep()

B．suspend()

C．wait()

D．yield()

8．下面程序被编译运行时，下面哪些描述是正确的\_\_\_\_\_\_\_\_\_。

**class** B{

**public** **int** f(**int** i){

**return** i+i;

}

**public** **int** f(**int** i, **int** j){

**return** i+j;

}

**public** **int** f(**char** c){

**return** c;

}

}

**public** **class** Test **extends** B{

**public** **int** f(**int** i, **int** j){

**return** **super**.f(i,j);

}

**public** **int** f(**int** j){

**return** -j;

}

**public** **static** **void** main(String[] args){

B x = **new** Test();

System.*out*.println(x.f(5,10)+" "+x.f(4)+" "+x.f('A'));

}

}

A．编译不通过，B x = new Test() 出错，类型不匹配

B．编译不通过，原因是B.f(char c)，一个char类型不能直接当做int类型返回

C．编译通过，运行输出结果是15 -4 65

D．编译通过，运行输出结果是15 8 65

9．下面哪个类或接口定义了wait()、notify()、notifyAll()方法\_\_\_\_\_\_\_

A．java.lang.Thread

B．java.lang.Runnable

C．java.lang.Object

D．java.lang.TreadGroup

10．在Java API文档中，下面的哪部分未被包括在内\_\_\_\_\_\_\_

A．类的用途的描述

B．类层次

C．父类方法的列表

D．成员变量的列表

11．下面列出的修饰符中，哪一个用来修饰方法(method)要求此方法必须显式地获得某个对象才能执行\_\_\_\_\_\_\_\_

A．final

B．synchronized

C．static

D．public

12．下面的各种异常中，哪种异常在一个方法中没有声明抛出此异常时也可以在其方法体中直接抛出\_\_\_\_\_\_\_\_

A．java.lang.ClassNotFoundException

B．java.lang.Exception

C．java.lang.NullPointerException

D．java.ioException

1. **多选题（共5题，每题3分，共15分）**

1．下面对操作系统中线程和进程的表述有错误的是\_\_\_\_\_\_\_

A．进程是程序的运行实例，它包含内存和资源，而线程是由进程进一步派生出来的一组代码的执行过程

B．每个进程都有一个主线程，还可以建立另外的线程。进程中的线程是并行执行的，每个线程占用CPU的时间由此进程来调度。

C．进程中的所有线程共享进程的虚拟地址空间，这意味着所有线程都可以访问进程的全局变量和资源。

D．进程中的每个线程的声明周期都在该进程中，每个线程都有自己的私有虚拟地址空间。

2．下面关于HashMap与Hashtable区别的表述正确的是\_\_\_\_\_\_

A．HashMap与Hashtable采用的hash/rehas算法基本一致。

B．HashMap允许将null作为一个entry得key或value，而Hashtable不允许。

C．HashMap与Hashtable都是Map interface的实现。

D．Hashtable的方法是syncronized的，而HashMap不是，因此HashMap非线程安全

3．对于一个abstract的method，不能用下面哪个Java关键字同时修饰\_\_\_\_\_\_\_

A．static

B．native

C．synchronized

D．final

4．下面关于Java中Vector、ArrayList、LinkedList的表述正确的是\_\_\_\_\_\_\_

A．这三个类都实现了java.util.List接口，但只有Vector是线程安全的

B．ArrayList与Vector在从指定位置用index检索一个元素的时候，时间开销是O(1)

C．LinkedList在插入、删除任意位置元素的时间开销都是O(1)，但它在索引一个元素的时候比较慢，为O(i)，其中i是索引的位置。

D．当向这几种类型中增加元素时，如果数目超出了其目前的长度，Vector缺省情况下自动增长原来一倍的长度，而ArrayList与LinkedList则增长50%的长度。

5．在J2EE中，JSP可以使用下列哪些技术对客户端会话状态进行跟踪\_\_\_\_\_\_\_\_\_\_

A．建立匿名的表格字段

B．使用URL重写

C．使用持久化Cookies

D．使用session会话机制

1. **填空题（共5题，每题3分，共15分）**

1．下面程序的运行结果为\_\_\_\_\_\_\_\_\_

**public** **class** Test{

**public** **static** **void** main(String[] args){

B obj = **new** B();

}

}

**class** A{

**static** {System.*out*.print("X");}

**public** A(){System.*out*.print("A");}

}

**class** B **extends** A{

**static** {System.*out*.print("Y");}

**public** B(){System.*out*.print("B");}

}

2．下面程序的运行结果为\_\_\_\_\_\_\_\_\_

**public** **class** Test{

**public** **static** **void** main(String[] args)

{

Sub obj = **new** Sub();

obj.print();

}

}

**class** Base{

**public** String s = "X";

**public** **void** print()

{

System.*out*.print(s);

printAgain();

}

**public** **void** printAgain()

{

System.*out*.print("A");

}

}

**class** Sub **extends** Base{

**public** String s = "Y";

**public** **void** printAgain()

{

System.*out*.print("B");

}

}

3．下面程序的运行结果为\_\_\_\_\_\_\_\_

**public** **class** Test{

**static** **int** func(**int** i, **int** j){

**try**{

**return** i/j;

}**catch**(ArithmeticException e){

System.*out*.print("A");

**return** 0;

}**catch**(Exception e){

System.*out*.print("B");

**return** 0;

}**finally**{

System.*out*.print("C");

}

}

**public** **static** **void** main(String[] args){

System.*out*.println(*func*(10,0));

}

}

4．下面程序的运行结果为\_\_\_\_\_\_\_

**class** A{

**void** func1(){

System.*out*.print("A");

}

}

**class** B **extends** A{

**void** func2(){

System.*out*.print("B");

}

}

**class** C **extends** B{

**void** func2(){

System.*out*.print("C");

}

}

**public** **class** Test{

**public** **static** **void** main(String[] args){

B obj = **new** C();

obj.func1();

obj.func2();

}

}

5．请填写下面程序的输出结果

**class** A{**int** a = 1;}

**public** **class** Test{

**public** **static** **void** main(String[] args){

A obj1 = **new** A();

A obj2 = **new** A();

A obj3 = obj1;

String s1 = **new** String("abc");

String s2 = **new** String("abc");

String s3 = "abc";

System.*out*.println(obj1 == obj2);

System.*out*.println(obj1.equals(obj2));

System.*out*.println(obj1 == obj3);

System.*out*.println(s1 == s2);

System.*out*.println(s1.equals(s2));

System.*out*.println(s1.equals(s3));

}

}

1. **简答题（共3题，前两题每题5分，最后一题6分，共16分）**

1．请编写一个函数func()，将此函数的输入参数（int型）逆序输出显示，如54321 –> 12345，要求使用递归，并且函数体代码不超过8行。

**public** **class** Test{

**public** **void** func(**int** i){

}

**public** **static** **void** main(String[] args){

Test t=**new** Test();

t.func(987);

}

}

2．请用两种方式实现Java中的Singletone模式的类的基本框架，并分析其是否线程安全。

3. 请用Java实现一个多线程的程序，其中一个线程对某静态全局变量进行100次自加，另一个线程对该变量进行100次自减，要求这两个线程的实现方式不同。

**-----------------------------------------Part II，总计30分--------------------------------------------------**

**一、选择题(共7题，每题2分，共14分)**

1.下面哪个是java的关键字

A NULL

B new

C instanceOf

D wend

2.下面语句哪个是正确的

A char s='abc';

B long l=oxfff;

C float f=0.45;

D double d =0.8E-3;

3.public class ExampleStringBuffer {

public static void main(String[] args) {

StringBuffer sb = new StringBuffer("mytest");

System.out.println("buffer=" + sb);

System.out.println("length=" + sb.length());

}

}

程序的运行结果中在“length=”后输出的值是

A 10

B 6

C 20

D 30

4.在下列的整数数据类型中，需要内存空间最小的是

A short

B long

C int

D byte

5. public class MyFor {

static boolean out(char input){

System.out.print(input);

return true;

}

public static void main(String arg[]){

int i=0;

for(out('A');out('B')&&i<2;out('C')){

i++;

out('D');

}

}

}

程序的运行结果是

A ABDCBDCB

B ABCDABCD

C 程序编译时出错

D 程序运行时异常

6.在Eclipse中程序代码自动排版的热键是：

A Ctrl+Shift+O

B Ctrl+O

C Ctrl+Shift+F

D Ctrl+F

7.编译运行以下程序后，关于输出结果的说明正确的是

public class Test{

public static void main(String args[ ]){

int x=6;

System.out.println("value is "+ ((x>6) ? 99.9 :9));

}

}

A、 输出结果为：value is 99.99

B、 输出结果为：value is 9

C、 输出结果为：value is 9.0

D、 编译错误

**三、问答编程题(共5题，共16分)**

1.spring配置文件中 (2分)

<bean id="TimerBeatTask"

class="org.springframework.scheduling.timer.ScheduledTimerTask">

<property name="timerTask">

<ref bean="TimerBeat" />

</property>

<property name="period">

<value>10000</value>

</property>

<property name="delay">

<value>10000</value>

</property>

</bean>

period属性配置的含义是：

delay属性配置的含义是：

2. http头中 transfer-encoding: chunked 的含义是？（2分）

3.写一程序将"张三", "李四", "王五" 按汉字拼音首字母排序 （4分）

4.写一段JAVA程序按一定格式输出系统当日期和时间。 （4分）

5. 有一自然数满足如下条件 x%2=1 x%3=2 x%5=4 x%6=5 x%7=0 ，请写一java程序算出x值。 （4分）