第五周笔试题

## 一、单选题（每题0.5分，共20分）

1. **下列关于类和对象的说法中，哪一种说法是错误的？（）**
2. 类是对具有相同特征和行为的一类事物的抽取和封装
3. 对象是具体的，类是抽象的

C. 创建并引用对象的语法是：

类名 对象名 = new 类名() ;

对象名.属性

对象名.方法名()

D. 属性是指对象执行的操作，方法是指对象具有的各种特征

1. **局部变量vs成员变量，下述那种说法是错误的？（）**
2. 定义位置：成员变量直接在类中定义，局部变量定义在方法中
3. 声明赋值：成员变量可以在声明时赋初始值，若不赋值，会有默认的初始值；

局部变量在使用时需要赋值，没有默认的初始值

C. 变量同名：局部变量可以和成员变量同名，并且在使用时，成员变量具有更高的优先级

D. 作用域：成员变量在整个类内部都是可见的，所有成员方法都可以使用它；

局部变量仅限于 定义它的方法，在该方法外部无法访它

1. **在Java中,以下关于构造方法描述错误的是（）。**
2. 构造方法的返回类型只能是void
3. 构造方法是类的一种特殊方法,它的方法名必须与类名相同
4. 构造方法的主要作用是完成对象的初始化操作
5. 一般创建新对象时,系统会白动调用构造方法
6. **有一个Student类，下面关于定义长度为3的Student类型数组，正确的是（）**
7. Student stus[3]; B. Student stus[]= new Student [3];

C. Student stus = Student(3); D. Student stus[3]= new Student [];

1. **以下关于static关键字的描述，错误的是（）**
2. 使用static关键字修饰的属性和方法升级为类属性和类方法
3. static修饰的属性或方法采用类名加“.”调用
4. 在非静态方法中不可以使用静态方法
5. 静态方法中不可以使用this和super关键字
6. **在Java中，关于类的继承，下面描述错误的是（）**
7. 子类与父类是is-a（什么是一种什么）关系

B. Java中只支持单继承，即一个类只能有一个直接父类

C. 继承是Java中实现代码重用的重要手段之一

D. 当前类要使用另一个类中的属性和方法，则该类应该继承另一个类

1. **关于Object类中的方法，下面描述错误的是（）**
2. Object不是一个抽象类，可以实例化
3. Object类是所有类的父类
4. Object类中的hashCode()方法返回该对象的哈希码值。
5. Object类中的equals(Object obj)方法，默认比较的是内容
6. **关于抽象类，下面描述错误的是（）**
7. 普通类可以实例化，抽象类不能实例化
8. 若一个类是抽象类，那么，在该类中至少有一个抽象方法
9. 抽象类要用abstract修饰

D. 抽象类是一个不能实例化的类，它可以具有抽象方法或者普通方法

1. **下面关于接口的描述，错误的是 （C）**
2. 接口中只能声明常量和抽象方法
3. 一个类可以实现多个接口，一个接口可以继承多个接口
4. 一个抽象类实现一个接口，则必须实现接口中的所有方法

D. 接口提供一种约定或者标准，使得实现接口的类在形式上保持一致

1. **关于抽象类与接口的区别，下面说法错误的是（）**

A．抽象类使用abstract声明，接口使用interface声明

B．接口中必须有内容，不允许出现空接口体

C．抽象类可以包含抽象方法和普通方法

D．接口只能包含抽象方法

1. **下面关于多态的描述，错误的是（）**

A．父类的引用指向子类的实例是多态的体现方式

B．接口的引用指向实现类的实例是多态的体现

C. 同一个引用类型，使用不同的实例而执行不同操作，就是多态。

D．父类的引用指向子类的对象时，可以调用子类特有的方法。

1. **下面关于内部类的说法，错误的是（）**

A．内部类就是定义在类内部的类

B．定义非静态成员内部类对象的语法是：

外部类类名.内部类类名 inner=new 外部类类名().new 内部类类名();

C．可以在方法体内部，定义静态内部类

D．非静态内部类可以访问外部类所有属性

1. **以下关于数组和集合的描述，正确的是（）**
2. 数组被创建好后，与集合一样，它的长度可以根据情况动态改变
3. 集合中存放的是引用数据类型

C．在一个数组中，可以存放多种类型的数据

D．数组元素的索引是从1开始，当索引值≧该数组长度时会产生数组越界异常

1. **下面关于异常的描述，错误的是（A）**
2. Error是Exception的子类，属于JVM级别的错误
3. Java使用异常处理机制为程序提供了错误处理的能力

C．在程序运行过程中如果发生异常，程序则会出现中断

D．在安排catch语句的顺序时，先应该捕获最特殊的异常，然后再逐渐一般化，即先子类后父类。

1. **下面关于throws和throw的说法，错误的是（）**

A．throw用在方法体中，throws用于方法的声明部分

B．throw关键字用在方法代码中主动抛出一个异常对象，若该异常对象属于Checked异常，那么，在该方法的声明部分必须使用throws关键字声明抛出该类型的异常或是该异常类型的父类

C．如果一个方法声明的是运行时异常，则在调用这个方法之处必须处理这个异常

D．throws声明某个方法可能抛出的各种异常

1. **对于下段代码，运行结果正确的是（）**

public class Test {

public int num;

public void calc(int num,int num2){

num += num2;

System.out.print(num+num2+”,”);

}

public static void main(String[] args) {

Test test = new Test();

test.num=10;

test.calc(11,3);

System.out.println(test.num);

}

}

A．14 ,10

B．13 ,10

C．13,13

D．17,10

1. **以下代码运行后，控制台显示的结果是（）**

class Student{

public int age;

}

public class Test {

public void calc1(int num){

num=num+1;

}

public void calc2(Student student){

student.age=student.age+1;

}

public static void main(String[] args){

Test test=new Test();

int num=8;

test.calc1(num);

Student stu=new Student();

stu.age=18;

test.calc2(stu);

System.out.println(num+"---"+stu.age);

}

}

A．9---19

B．9---18

C．8---19

D．8---18

1. **以下关于字符串String和StringBuffer及StringBuilder的描述，错误的是（）**

A．String是指不可变的字符序列，而StringBuffer是指可变的字符序列

B．如果程序中需要对存储的字符串序列进行不断的改变，建议采用String，而不是StringBuffer或StringBuilder

C．StringBuffer是线程安全的，而StringBuilder是线程不安全的

D．String类的split()方法，用于字符串的切割，返回的结果是一个String类型的数组。

1. **以下关于集合的描述，错误的是（）**

A．Map接口存储一组成对的键－值对象，提供key(键)到value(值)的映射

B．List 接口存储一组可重复，有序（插入顺序）的对象

C．Set接口存储一组唯一，无序的对象

D．List，Set,Map都是Collection的子接口

1. **以下关于集合的描述，不正确的是（）**

A．HashSet类实现了Set接口，底层数据结构是哈希表。

B．TreeSet类实现了Set接口，可以对Set集合中的元素进行排序，底层数据结构是二叉树。

C．ArrayList集合在遍历的时候，可以使用迭代器迭代遍历，也可以使用普通for循环遍历，

还可以使用增强for循环来遍历

D．Collections是集合框架中的一个接口。

1. **main方法如下所示，该程序的运行结果是（）。**

public static void main(String [] args){

int i=2;

System.out.print(i--);

System.out.print(--i);

}

1. 10
2. 21
3. 20
4. 运行时出现异常
5. **在Java中，如何得到数组的长度（）。**

A、 array.length();

B、 array.length;

C、 array.size();

D、 array.size;

1. **给定如下代码片段，编译运行时的结果是（）。**

int i = 2;

switch(i){

case 0:

System.out.println (”zero”);

break;

case 1:

System.out.println (”one”);

case 2:

System.out.println (“two”);

default:

System.out.println (“default”);

}

A、 输出：zero

B、 输出：zero

default

C、输出：two

default

D、输出：two

1. **下列赋值语句中错误的是 ( )**

A、float f=11.1f

B、double d=5.3E12;

C、char c='\r';

D、byte bb=443;

1. **分析如下的代码，编译运行时将输出（）。**

public class Test {

public static void main(String[] args) {

Student s = new Student();

}

}

class Person{

public Person(){

System.out.print("Person");

}

}

class Student extends Person{

public Student(){

System.out.print("Student");

}

}

A、Person B、Student C、PersonStudent D、StudentPerson

1. **下面的方法，正确的调用是：（）**

public  float  add（double  a，double  b）{   }

A、add（10，20，30）

B、add（10.0，20.5）

C、add（’a’ , ‘b’）

D、add（"10.0" , "20.0"）

1. **正确的输出结果是（B）。**

int i ;

for(i = 0;i<4; i+=2){

System.out.print (i+＂,＂);

}

System.out.print (i);

A、 0,2

B、 0,2,4

C、 0,2,4,6

D、 0,1,2,3

1. **在Java中，如果要在字符串类型s="java"中，得到字母 'v' 出现的位置，选（）语句。**
2. s.matches('v');
3. s.charAt('v');
4. s.indexOf('v');
5. s.substring('v');
6. **分析下面的Java程序段，编译运行后的输出结果是（B）。**

public class Test {

public static void changeString(StringBuffer sb) {

sb = new StringBuffer();

sb.append("stringbuffer2");

}

public static void main(String[] args) {

StringBuffer sb = new StringBuffer("stringbuffer1");

changeString(sb);

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

}

}

A、sb = stringbuffer2stringbuffer1

B、sb = stringbuffer1

C、sb = stringbuffer2

D、sb = stringbuffer1stringbuffer2

1. **以下能对一维数组a正确初始化的语言是（）**
2. int[2] a={1},{2};
3. int[1] a={10,20};

C、int[] a={0};

D、int[2] a={10.0};

1. **下列代码执行之后，输出的结果为\_\_\_\_\_。**

public class ex24 {

public static void main(String[] args) {

int x=12;

int m=11;

int y=13;

int temp= x > y ? x : y;

temp= temp > m ? temp : m;

System.out.println (temp);

}

}

A 、1

B、 12

C、 13

D、 11

1. **分析下面的源程序，编译后的运行结果是（）。**

public class Test{

public static void main (String[] args){

int[] numbers=new int[]{2,1,3};

Arrays.sort(numbers);

for(int index=1 ; index < numbers.length ; index++){

System.out.print (numbers[index]);

}

}

}

A、21

B、23

C、13

D、运行时报数组越界错误

1. **编译运行后的输出结果是（）。**

public static void main (String[] args){

int a=100;

while(a != 0){

System.out.print (a);

a=a/5；

}

}

A、 10020

B、 100204

C、 100

D、 20

1. **java.io包中提供的用于筛选文件名的接口是（）**

A、FileInputStream

B、FileOutputStream

C、FileNameFilter

D、RandomAccessFile

1. **下面的说法错误的是（）**

A、一个线程就是一个Thread实例

B、启动线程的正确方法是调用start方法

C、当线程调用start方法之后，线程将立即进入运行状态

D、主线程和子线程默认的优先级是一样的

1. **关于final关键字，下面说法错误的是（）**

A、用final修饰的变量即为常量，只能赋值一次

B、final修饰类，则类不能被继承

C、final修饰代码块，该代码块只能执行一次

D、final修饰方法，则该方法不能被子类重写

1. **通常情况下，下列哪一个类的对象可以作为BufferedReader构造方法的参数（A）**

A、InputStreamReader

B、PrintStream

C、FileInputStream

D、OutputStreamReader

1. **以下关于数组的描述，正确的是（）**

A、在一个数组中，可以存放多种数据类型的数据

B、数组被创建好后，它的大小可以根据情况进行改变

C、在JAVA语言中，最多只能声明二维数组，三维数组Java不支持

D、数组元素的索引是从0开始，当索引值大于该数组长度时会产生数组越界异常

1. **以下代码运行后，控制台显示的结果是（）**

public class A{

public static void main(String[] args){

String str ="aa,bb,cc,dd,";

String[] ss = str.split(",");

System.out.println(ss.length);

}

}

A、3

B、4

C、5

D、6

1. **以下哪些异常是非运行时即编译时（checked Exception）异常？（B）**

A、NullPointerException

B、IOException

C、ArrayIndexOutOfBoundsException

D、ArithmeticException

## 二、多选题（每题1分，共10分）

**注意：多选题的正确答案至少是两个或两个以上,少选多选不得分**

1. **关于构造方法，说法错误的是（）**
2. 构造方法与类同名
3. 一个类只能有一个构造方法
4. 构造方法必须是公有的
5. 构造方法无返回值
6. **以下关于方法重写的原则，描述正确的是（ABD）**
7. 重写方法和被重写方法必须具有相同的方法名和参数列表 正确
8. 所谓方法重写：即在子类中根据需求对从父类继承的方法进行重新编写
9. 重写方法可以抛出比父类方法更多的异常
10. 重写方法的不能缩小被重写方法的访问权限
11. **以下关于多线程的说法，正确的是（AD）**
12. 线程是进程中的更小单位，共享着进程的资源
13. Java中实现多线程，可以通过实现Thread类或继承Runnable接口的方式来实现
14. 当调用线程的start方法启动线程后，线程将会立即进入执行状态，执行run方法

D．在多线程环境下，并且存在多个线程操作同一个数据资源时，我们需要考虑线程的安全问题，即加入同步处理。

1. **以下的描述，正确的是（ABD）**
2. 构造方法、静态方法、私有方法、final方法不能被声明为抽象的方法 正确
3. 如果一个子类没有实现抽象父类中所有的抽象方法，则子类也需要定义为抽象类
4. 抽象方法既有方法声明，也有方法的实现
5. 抽象类可以没有抽象方法
6. **以下关于集合的描述，正确的是（AD）**
7. HashSet集合中不可以存储重复元素。
8. TreeMap中存储键值对(key-value)时，需要对键(key)进行排序。所以要求键(key)要实现Comparable接口或者Comparator接口。 构造方法中传入comparator
9. ArrayList集合可以通过get(int index)的方法获取元素，LinkedList集合没有。
10. Stack(栈)是以后进先出(LIFO)的方式进行存取元素的。
11. **以下关于转换流的说法正确的是（）**
12. InputStreamReader是将字节转换为字符
13. InputStreamReader是将字符转换为字节
14. OutputStreamWriter是将字节转换为字符
15. OutputStreamWriter是将字符转换为字节
16. **以下说法，正确的是（ BCD）**
17. break只能在switch循环语句中使用，用于跳出循环
18. break只能跳出一层循环 多层循环中只能跳出一层
19. return可以用于结束方法的执行，表示方法执行到此结束
20. continue用于跳过本次循环，但并不是结束整个循环
21. **以下说法，正确的是（）**
22. 对象转型分为向上转型和向下转型
23. 向上转型是不安全的，向下转型是安全的
24. 向上转型是安全的，向下转型是不安全的
25. 多态就是一种向下转型的体现
26. **以下说法，正确的是（）**
27. Math.ceil(7.1)的结果为7.0
28. Math.ceil(7.9)的结果为8.0
29. Math.floor(7.1)的结果为7.0
30. Math.floor(7.9)的结果为8.0
31. **以下说法，正确的是（）**
32. HashMap是以hash表来对数据进行存储，要求其存放的对象必须实现Comparable接口
33. TreeMap是以hash表来对数据进行存储，要求其存放的对象必须实现Comparable接口
34. HashSet是以hash表来对数据进行存储，所以输出的顺序和存放的顺序不一定是一致的
35. TreeSet会对存放到里面的数据进行排序，可以让存放的对象实现Comparable接口或者在创建TreeSet对象的时候，传递一个Comparator比较器对象