JAVA面试基础题及答案

1、面向对象的特征有哪些方面   
1.抽象：  
抽象就是忽略一个主题中与当前目标无关的那些方面，以便更充分地注意与当前目标有关的方面。抽象并不打算了解全部问题，而只是选择其中的一部分，暂时不用部分细节。抽象包括两个方面，一是过程抽象，二是数据抽象。  
2.继承：  
继承是一种联结类的层次模型，并且允许和鼓励类的重用，它提供了一种明确表述共性的方法。对象的一个新类可以从现有的类中派生，这个过程称为类继承。新类继承了原始类的特性，新类称为原始类的派生类（子类），而原始类称为新类的基类（父类）。派生类可以从它的基类那里继承方法和实例变量，并且类可以修改或增加新的方法使之更适合特殊的需要。  
3.封装：  
封装是把过程和数据包围起来，对数据的访问只能通过已定义的界面。面向对象计算始于这个基本概念，即现实世界可以被描绘成一系列完全自治、封装的对象，这些对象通过一个受保护的接口访问其他对象。  
4. 多态性：  
多态性是指允许不同类的对象对同一消息作出响应。多态性包括参数化多态性和包含多态性。多态性语言具有灵活、抽象、行为共享、代码共享的优势，很好的解决了应用程序函数同名问题。  
2、String是最基本的数据类型吗?  
基本数据类型包括byte、int、char、long、float、double、boolean和short。  
3、int 和 Integer 有什么区别  
Java 提供两种不同的类型：引用类型和原始类型（或内置类型）。Int是java的原始数据类型，Integer是java为int提供的封装类。Java为每个原始类型提供了封装类。  
原始类型封装类  
booleanBoolean charCharacter byteByte shortShort intInteger  
longLong floatFloat doubleDouble  
引用类型和原始类型的行为完全不同，并且它们具有不同的语义。引用类型和原始类型具有不同的特征和用法，它们包括：大小和速度问题，这种类型以哪种类型的数据结构存储，当引用类型和原始类型用作某个类的实例数据时所指定的缺省值。对象引用实例变量的缺省值为 null，而原始类型实例变量的缺省值与它们的类型有关。  
4、String 和StringBuffer的区别  
String和StringBuffer，它们可以储存和操作字符串，即包含多个字符的字符数据。这个String类提供了数值不可改变的字符串。而这个StringBuffer类提供的字符串进行修改。当你知道字符数据要改变的时候你就可以使用StringBuffer。典型地，你可以使用StringBuffers来动态构造字符数据。  
5、运行时异常与一般异常有何异同？  
异常表示程序运行过程中可能出现的非正常状态，运行时异常表示虚拟机的通常操作中可能遇到的异常，是一种常见运行错误。java编译器要求方法必须声明抛出可能发生的非运行时异常，但是并不要求必须声明抛出未被捕获的运行时异常。  
6、说出ArrayList,Vector, LinkedList的存储性能和特性  
ArrayList和Vector都是使用数组方式存储数据，此数组元素数大于实际存储的数据以便增加和插入元素，它们都允许直接按序号索引元素，但是插入元素要涉及数组元素移动等内存操作，所以索引数据快而插入数据慢，Vector由于使用了synchronized方法（线程安全），通常性能上较ArrayList差，而LinkedList使用双向链表实现存储，按序号索引数据需要进行前向或后向遍历，但是插入数据时只需要记录本项的前后项即可，所以插入速度较快。  
7、Collection 和 Collections的区别。   
　　Collection是集合类的上级接口，继承与他的接口主要有Set 和List.  
Collections是针对集合类的一个帮助类，他提供一系列静态方法实现对各种集合的搜索、排序、线程安全化等操作。  
8、&和&&的区别。

&和&&都是逻辑运算符，都是判断两边同时真则为真，否则为假；但是&&当第一个条件不成之后，后面的条件都不执行了，而&则还是继续执行，直到整个条件语句执行完为止。   
9、HashMap和Hashtable的区别。   
HashMap是Hashtable的轻量级实现（非线程安全的实现），他们都完成了Map接口，主要区别在于HashMap允许空（null）键值（key）,由于非线程安全，效率上可能高于Hashtable。  
HashMap允许将null作为一个entry的key或者value，而Hashtable不允许。  
HashMap把Hashtable的contains方法去掉了，改成containsvalue和containsKey。因为contains方法容易让人引起误解。   
Hashtable继承自Dictionary类，而HashMap是Java1.2引进的Map interface的一个实现。  
最大的不同是，Hashtable的方法是Synchronize的，而HashMap不是，在多个线程访问Hashtable时，不需要自己为它的方法实现同步，而HashMap 就必须为之提供外同步（如果是ArrayList：List lst = Collections.synchronizedList(new ArrayList());如果是HashMap：Map map = Collections.synchronizedMap(new HashMap());）。  
Hashtable和HashMap采用的hash/rehash算法都大概一样，所以性能不会有很大的差异。  
10、final, finally, finalize的区别。   
　　final 用于声明属性，方法和类，分别表示属性不可变，方法不可覆盖，类不可继承。  
finally是异常处理语句结构的一部分，表示总是执行。  
finalize是Object类的一个方法，在垃圾收集器执行的时候会调用被回收对象的此方法，可以覆盖此方法提供垃圾收集时的其他资源回收，例如关闭文件等。  
11、sleep() 和 wait() 有什么区别?   
sleep是线程类（Thread）的方法，导致此线程暂停执行指定时间，给执行机会给其他线程，但是监控状态依然保持，到时后会自动恢复。调用sleep不会释放对象锁。  
wait是Object类的方法，对此对象调用wait方法导致本线程放弃对象锁，进入等待此对象的等待锁定池，只有针对此对象发出notify方法（或notifyAll）后本线程才进入对象锁定池准备获得对象锁进入运行状态。  
12、Overload(重载)和Override(重写)的区别。Overloaded的方法是否可以改变返回值的类型?  
方法的重写Overriding和重载Overloading是Java多态性的不同表现。重写Overriding是父类与子类之间多态性的一种表现，重载Overloading是一个类中多态性的一种表现。如果在子类中定义某方法与其父类有相同的名称和参数，我们说该方法被重写 (Overriding)。子类的对象使用这个方法时，将调用子类中的定义，对它而言，父类中的定义如同被“屏蔽”了。如果在一个类中定义了多个同名的方法，它们或有不同的参数个数或有不同的参数类型，则称为方法的重载(Overloading)。Overloaded的方法是可以改变返回值的类型。  
13、error和exception有什么区别?  
error 表示恢复不是不可能但很困难的情况下的一种严重问题。比如说内存溢出。不可能指望程序能处理这样的情况。  
exception 表示一种设计或实现问题。也就是说，它表示如果程序运行正常，从不会发生的情况。  
14、同步和异步有何异同，在什么情况下分别使用他们？举例说明。  
如果数据将在线程间共享。例如正在写的数据以后可能被另一个线程读到，或者正在读的数据可能已经被另一个线程写过了，那么这些数据就是共享数据，必须进行同步存取。  
当应用程序在对象上调用了一个需要花费很长时间来执行的方法，并且不希望让程序等待方法的返回时，就应该使用异步编程，在很多情况下采用异步途径往往更有效率。  
15、abstract class和interface有什么区别?  
声明方法的存在而不去实现它的类被叫做抽象类（abstract class），它用于要创建一个体现某些基本行为的类，并为该类声明方法，但不能在该类中实现该类的情况。不能创建abstract 类的实例。然而可以创建一个变量，其类型是一个抽象类，并让它指向具体子类的一个实例。不能有抽象构造函数或抽象静态方法。Abstract 类的子类为它们父类中的所有抽象方法提供实现，否则它们也是抽象类为。取而代之，在子类中实现该方法。知道其行为的其它类可以在类中实现这些方法。  
接口（interface）是抽象类的变体。在接口中，所有方法都是抽象的。多继承性可通过实现这样的接口而获得。接口中的所有方法都是抽象的，没有一个有程序体。接口只可以定义static final成员变量。接口的实现与子类相似，除了该实现类不能从接口定义中继承行为。当类实现特殊接口时，它定义（即将程序体给予）所有这种接口的方法。然后，它可以在实现了该接口的类的任何对象上调用接口的方法。由于有抽象类，它允许使用接口名作为引用变量的类型。通常的动态联编将生效。引用可以转换到接口类型或从接口类型转换，instanceof 运算符可以用来决定某对象的类是否实现了接口。  
16、heap和stack有什么区别。  
栈是一种线形集合，其添加和删除元素的操作应在同一段完成。栈按照后进先出的方式进行处理。  
堆是栈的一个组成元素。  
17、Static Nested Class 和 Inner Class(内部类)的不同。   
Static Nested Class是被声明为静态（static）的内部类，它可以不依赖于外部类实例被实例化。而通常的内部类需要在外部类实例化后才能实例化。  
18、short s1 = 1; s1 = s1 + 1;有什么错? short s1 = 1; s1 += 1;有什么错?   
short s1 = 1; s1 = s1 + 1; （s1+1运算结果是int型，需要强制转换类型）  
short s1 = 1; s1 += 1;（可以正确编译）  
19、Math.round(11.5)等於多少? Math.round(-11.5)等於多少?   
Math.round(11.5)==12  
Math.round(-11.5)==-11  
round方法返回与参数最接近的长整数，参数加1/2后求其floor.  
20、String s = new String("xyz");创建了几个String Object?   
两个  
21、启动一个线程是用run()还是start()?  
启动一个线程是调用start()方法，使线程所代表的虚拟处理机处于可运行状态，这意味着它可以由JVM调度并执行。这并不意味着线程就会立即运行。run()方法可以产生必须退出的标志来停止一个线程。  
22、给我一个你最常见到的runtime exception。  
ArithmeticException, ArrayStoreException, BufferOverflowException, BufferUnderflowException, CannotRedoException, CannotUndoException, ClassCastException, CMMException, ConcurrentModificationException, DOMException, EmptyStackException, IllegalArgumentException, IllegalMonitorStateException, IllegalPathStateException, IllegalStateException, ImagingOpException, IndexOutOfBoundsException, MissingResourceException, NegativeArraySizeException, NoSuchElementException, NullPointerException, ProfileDataException, ProviderException, RasterFormatException, SecurityException, SystemException, UndeclaredThrowableException, UnmodifiableSetException, UnsupportedOperationException  
23、接口是否可继承接口? 抽象类是否可实现(implements)接口? 抽象类是否可继承实体类?  
接口可以继承接口。抽象类可以实现(implements)接口，抽象类是否可继承实体类，但前提是实体类必须有明确的构造函数。  
24、List, Set, Map是否继承自Collection接口?  
List，Set是，Map不是  
  
25、数组有没有length()这个方法? String有没有length()这个方法？  
数组没有length()这个方法，有length的属性。String有有length()这个方法。  
26、Set里的元素是不能重复的，那么用什么方法来区分重复与否呢? 是用==还是equals()? 它们有何区别?  
Set里的元素是不能重复的，那么用iterator()方法来区分重复与否。equals()是判读两个Set是否相等。  
equals()和==方法决定引用值是否指向同一对象equals()在类中被覆盖，为的是当两个分离的对象的内容和类型相配的话，返回真值。  
27、构造器Constructor是否可被override?  
构造器Constructor不能被继承，因此不能重写Overriding，但可以被重载Overloading。  
28、是否可以继承String类?  
String类是final类故不可以继承。  
29、swtich是否能作用在byte上，是否能作用在long上，是否能作用在String上?

在 [Java](https://www.baidu.com/s?wd=Java&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWP1cvrjDdPvnduhNWnvcL0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnW0dPjD3rjbLrjT3n1TkPjb3Ps) 7之前，switch 只能支持 byte、short、char、int或者其对应的封装类以及 Enum 类型。在 [Java](https://www.baidu.com/s?wd=Java&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dWP1cvrjDdPvnduhNWnvcL0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnW0dPjD3rjbLrjT3n1TkPjb3Ps) 7中，String支持被加上了。  
30、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return前执行。  
31、两个对象值相同(x.equals(y) == true)，但却可有不同的hash code，这句话对不对?  
不对，有相同的hash code。  
32、当一个对象被当作参数传递到一个方法后，此方法可改变这个对象的属性，并可返回变化后的结果，那么这里到底是值传递还是引用传递?   
是值传递。Java 编程语言只有值传递参数。当一个对象实例作为一个参数被传递到方法中时，参数的值就是对该对象的引用。对象的内容可以在被调用的方法中改变，但对象的引用是永远不会改变的。  
33、当一个线程进入一个对象的一个synchronized方法后，其它线程是否可进入此对象的其它方法?  
不能，一个对象的一个synchronized方法只能由一个线程访问。  
34、Java中的异常处理机制的简单原理和应用。  
当JAVA程序违反了JAVA的语义规则时，JAVA虚拟机就会将发生的错误表示为一个异常。违反语义规则包括2种情况。一种是JAVA类库内置的语义检查。例如数组下标越界,会引发IndexOutOfBoundsException;访问null的对象时会引发NullPointerException。另一种情况就是JAVA允许程序员扩展这种语义检查，程序员可以创建自己的异常，并自由选择在何时用throw关键字引发异常。所有的异常都是java.lang.Thowable的子类。  
35、请说出你所知道的线程同步的方法。  
wait():使一个线程处于等待状态，并且释放所持有的对象的lock。  
sleep():使一个正在运行的线程处于睡眠状态，是一个静态方法，调用此方法要捕捉InterruptedException异常。  
notify():唤醒一个处于等待状态的线程，注意的是在调用此方法的时候，并不能确切的唤醒某一个等待状态的线程，而是由JVM确定唤醒哪个线程，而且不是按优先级。  
Allnotity():唤醒所有处入等待状态的线程，注意并不是给所有唤醒线程一个对象的锁，而是让它们竞争。  
36、你所知道的集合类都有哪些？主要方法？  
最常用的集合类是 List 和 Map。 List 的具体实现包括 ArrayList 和 Vector，它们是可变大小的列表，比较适合构建、存储和操作任何类型对象的元素列表。 List 适用于按数值索引访问元素的情形。   
Map 提供了一个更通用的元素存储方法。 Map 集合类用于存储元素对（称作“键”和“值”），其中每个键映射到一个值。  
37、char型变量中能不能存贮一个中文汉字?为什么?   
能够定义成为一个中文的，因为java中以unicode编码，一个char占16个字节，所以放一个中文是没问题的  
38、多线程有几种实现方法,都是什么?同步有几种实现方法,都是什么?   
多线程有两种实现方法，分别是继承Thread类与实现Runnable接口   
同步的实现方面有两种，分别是synchronized,wait与notify  
39、线程的基本概念、线程的基本状态以及状态之间的关系  
线程指在程序执行过程中，能够执行程序代码的一个执行单位，每个程序至少都有一个线程，也就是程序本身。  
Java中的线程有四种状态分别是：运行、就绪、挂起、结束。   
40、排序都有哪几种方法？请列举。用JAVA实现一个快速排序。  
排序的方法有：插入排序（直接插入排序、希尔排序），交换排序（冒泡排序、快速排序），选择排序（直接选择排序、堆排序），归并排序，分配排序（箱排序、基数排序）  
快速排序的伪代码。  
/ /使用快速排序方法对a[ 0 :n- 1 ]排序  
从a[ 0 :n- 1 ]中选择一个元素作为m i d d l e，该元素为支点  
把余下的元素分割为两段left 和r i g h t，使得l e f t中的元素都小于等于支点，而right 中的元素都大于等于支点  
递归地使用快速排序方法对left 进行排序  
递归地使用快速排序方法对right 进行排序  
所得结果为l e f t + m i d d l e + r i g h t  
41、JAVA语言如何进行异常处理，关键字：throws,throw,try,catch,finally分别代表什么意义？在try块中可以抛出异常吗？  
Java通过面向对象的方法进行异常处理，把各种不同的异常进行分类，并提供了良好的接口。在Java中，每个异常都是一个对象，它是Throwable类或其它子类的实例。当一个方法出现异常后便抛出一个异常对象，该对象中包含有异常信息，调用这个对象的方法可以捕获到这个异常并进行处理。Java的异常处理是通过5个关键词来实现的：try、catch、throw、throws和finally。一般情况下是用try来执行一段程序，如果出现异常，系统会抛出（throws）一个异常，这时候你可以通过它的类型来捕捉（ca tch）它，或最后（finally）由缺省处理器来处理。  
用try来指定一块预防所有“异常”的程序。紧跟在try程序后面，应包含一个catch子句来指定你想要捕捉的“异常”的类型。  
throw语句用来明确地抛出一个“异常”。  
throws用来标明一个成员函数可能抛出的各种“异常”。  
Finally为确保一段代码不管发生什么“异常”都被执行一段代码。  
可以在一个成员函数调用的外面写一个try语句，在这个成员函数内部写另一个try语句保护其他代码。每当遇到一个try语句，“异常”的框架就放到堆栈上面，直到所有的try语句都完成。如果下一级的try语句没有对某种“异常”进行处理，堆栈就会展开，直到遇到有处理这种“异常”的try语句。  
42、一个“.java”源文件中是否可以包括多个类（不是内部类）？有什么限制？  
可以。必须只有一个类名与文件名相同。  
  
43、java中有几种方法可以实现一个线程？用什么关键字修饰同步方法? stop()和suspend()方法为何不推荐使用？  
有两种实现方法，分别是继承Thread类与实现Runnable接口  
用synchronized关键字修饰同步方法  
反对使用stop()，是因为它不安全。它会解除由线程获取的所有锁定，而且如果对象处于一种不连贯状态，那么其他线程能在那种状态下检查和修改它们。结果很难检查出真正的问题所在。suspend()方法容易发生死锁。调用suspend()的时候，目标线程会停下来，但却仍然持有在这之前获得的锁定。此时，其他任何线程都不能访问锁定的资源，除非被“挂起”的线程恢复运行。对任何线程来说，如果它们想恢复目标线程，同时又试图使用任何一个锁定的资源，就会造成死锁。所以不应该使用suspend()，而应在自己的Thread类中置入一个标志，指出线程应该活动还是挂起。若标志指出线程应该挂起，便用wait()命其进入等待状态。若标志指出线程应当恢复，则用一个notify()重新启动线程。  
44、java中有几种类型的流？JDK为每种类型的流提供了一些抽象类以供继承，请说出他们分别是哪些类？  
字节流，字符流。字节流继承于InputStream OutputStream，字符流继承于InputStreamReader OutputStreamWriter。在java.io包中还有许多其他的流，主要是为了提高性能和使用方便。  
45、java中会存在内存泄漏吗，请简单描述。  
会。如：int i,i2; return (i-i2); //when i为足够大的正数,i2为足够大的负数。结果会造成溢位，导致错误。  
46、java中实现多态的机制是什么？  
方法的重写Overriding和重载Overloading是Java多态性的不同表现。重写Overriding是父类与子类之间多态性的一种表现，重载Overloading是一个类中多态性的一种表现。  
47、静态变量和实例变量的区别？  
static i = 10; //常量  
class A a; a.i =10;//可变  
48、什么是java序列化，如何实现java序列化？  
序列化就是一种用来处理对象流的机制，所谓对象流也就是将对象的内容进行流化。可以对流化后的对象进行读写操作，也可将流化后的对象传输于网络之间。序列化是为了解决在对对象流进行读写操作时所引发的问题。  
序列化的实现：将需要被序列化的类实现Serializable接口，该接口没有需要实现的方法，implements Serializable只是为了标注该对象是可被序列化的，然后使用一个输出流(如：FileOutputStream)来构造一个ObjectOutputStream(对象流)对象，接着，使用ObjectOutputStream对象的writeObject(Object obj)方法就可以将参数为obj的对象写出(即保存其状态)，要恢复的话则用输入流。  
49、是否可以从一个static方法内部发出对非static方法的调用？  
不可以,如果其中包含对象的method()；不能保证对象初始化.  
50、在JAVA中，如何跳出当前的多重嵌套循环？  
用break; return 方法。  
51、List、Map、Set三个接口，存取元素时，各有什么特点？  
List 以特定次序来持有元素，可有重复元素。Set 无法拥有重复元素,内部排序。Map 保存key-value值，value可多值。  
52、说出一些常用的类，包，接口，请各举5个  
常用的类：BufferedReader BufferedWriter FileReader FileWirter String Integer  
常用的包：java.lang java.awt java.io java.util java.sql  
常用的接口：Remote List Map Document NodeList   
53、内部类可以引用他包含类的成员吗？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内容