

JAVA程序员面试之《葵花宝典》 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/167/2021_2022_JAVA_E7_A8_8B_E5_BA_8F_c104_167000.htm 程序员面试之葵花宝典 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2. 无关的那些方面，3. 以便更充分地注意与当前目标4. 有关的方面。抽象并不5. 打算了解全部问题，而6. 只是选择其中的一部分，7. 暂时不8. 用部分细节。抽象包括两个方面，9. 一是过程抽象，10. 二是数据抽象。 11. 继承：继承是一种联结类的层次模型，12. 并且允许和鼓励类的重用，13. 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，14. 这个过程称为类继承。新类继承了原始类的特性，15. 新类称为原始类的派生类（子类），而16. 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，17. 并且类可以修改或增加新的方法使之更适合特殊的需要。 18. 封装：封装是把过程和数据包围起来，19. 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，20. 即现实世界可以被描绘成一系列完全自治、封装的对象，21. 这些对象通过一个受保护的接口访问其他对象。 22. 多态性：多态性是指23. 允许不同24. 类的对象对同25. 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，26. 很好的解决了应用程序函数同27. 名28. 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。java.lang.String类是final类型的，因此不可以继承这个类、不能修改这个类

。为了提高效率节省空间，我们应该用StringBuffer类

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
boolean Boolean
char Character
byte Byte
short Short
int Integer
long Long
float Float
double Double

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

JAVA平台提供了两个类：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Servlet的生命周期，并说出Servlet和CGI的区别。

Servlet被服务器实例化后，容器运行其init方法，请求到达时运行其service方法，service方法自动派遣运行与请求对应的doXXX方法（doGet，doPost）等，当服务器决定将实例销毁的时候

调用其destroy方法。与cgi的区别在于servlet处于服务器进程中，它通过多线程方式运行其service方法，一个实例可以服务于多个请求，并且其实例一般不会销毁，而CGI对每个请求都产生新的进程，服务完成后就销毁，所以效率上低于servlet。

7、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。

8、EJB是基于哪些技术实现的？并说出SessionBean和EntityBean的区别，StatefulBean

和StatelessBean的区别。EJB包括Session Bean、Entity Bean

、Message Driven Bean，基于JNDI、RMI、JAT等技术实现。

SessionBean在J2EE应用程序中被用来完成一些服务器端的业务操作，例如访问数据库、调用其他EJB组件。EntityBean被

用来代表应用系统中用到的数据。对于客户机，SessionBean是一种非持久性对象，它实现某些在服务器上运行的业务逻辑。

对于客户机，EntityBean是一种持久性对象，它代表一个存储在持久性存储器中的实体的对象视图，或是一个由现有

企业应用程序实现的实体。Session Bean还可以再细分为

Stateful Session Bean 与 Stateless Session Bean，这两种的Session

Bean都可以将系统逻辑放在method之中执行，不同的是

Stateful Session Bean 可以记录呼叫者的状态，因此通常来说，

一个使用者会有一个相对应的 Stateful Session Bean 的实体。 Stateless Session Bean 虽然也是逻辑组件，但是他却不负责记录使用者状态，也就是说当使用者呼叫 Stateless Session Bean 的时候， EJB Container 并不会找寻特定的 Stateless Session Bean 的实体来执行这个 method。换言之，很可能数个使用者在执行某个 Stateless Session Bean 的 methods 时，会是同一个 Bean 的 Instance 在执行。从内存方面来看， Stateful Session Bean 与 Stateless Session Bean 比较， Stateful Session Bean 会消耗 J2EE Server 较多的内存，然而 Stateful Session Bean 的优势却在于他可以维持使用者的状态。

9、 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。

10、 `&` 是位运算符，表示按位与运算， `&&` 是逻辑运算符，表示逻辑与 (and)。

11、 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 Java 1.2 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。 Hashtable

和HashMap采用的hash/rehash算法都大概一样，所以性能不会有很大差异。

12、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。

13、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。

14、Overload和Override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。

100Test 下载频道开通，各类考试题目直接下载。详细请访问

www.100test.com