

Java 面试前准备

感谢您参加我们的面试：

以下是我们总结一些面试前准备和技术参考题，希望对你有所帮助。

首先，你需要先准备下工作相关的**自我介绍**，包括以下内容：

1. 几年 **Java** 开发经验，大概概括下做过什么类型的项目，分别多久经验（如互联网金融，互联网电商，传统后台等）
2. 选择 1~2 个近期关键项目具体介绍：
 - 主要担任什么岗位，工作内容是什么，也可以简单介绍下团队人员组成，你在其中是属于什么角色（如架构搭建，核心代码编写，主要做接口开发等。。。）
 - 项目的技术是什么（请具体到开发环境，框架，开发工具，数据库等），如有一些比较创新的技术可以具体说明
 - 项目成果怎么样（是否上线，用户使用效果如何等）
 - 个人技术总结/自己成就突出（如你们怎么处理技术难题；你自己有什么建设性意见改进项目；项目技术是否有不足，你觉得要是你会怎么做）

其次，你需要看下您的简历，把你简历里面写的技术点（尤其写了精通，熟悉的）再过一下。

- 很多时候技术面试官会根据你简历里面写的技术点，有浅到深地提问。

最后，以下是我们给你提供的一些 **Java** 基础技术参考题，没有答案，需要你自己去准备哦~

祝面试成功，加油！

- **数据库相关：求平均成绩大于 60 分的学生名字和他们个人的平均分。**
- 值传递与引用传递的区别；
- 重载和重写的区别
- **spring** 框架的 **IOC** ,**AOP** 个人的理解
- **struts** **struts2** **springmvc** 框架底层实现和他们的区别；
- **hibernate** 框架的一级缓存和二级缓存
- **java** 类加载器是什么，**http** 协议
- **java** 的内存机制
- **java** 的数据结构
- **redis** 基本数据类型
- **spring** 事务的传播机制
- **bean** 的作用

- 用过什么设计模式

- 单例模式多例的区别
- 什么是工厂模式，工厂模式都有哪几种，什么情况下用
- java 的反射机制是什么
- 什么是动态代理
- http 的 7 层实现（或者叫 3 次握手是什么）
- 防止 sql 注入用哪几种方式
- 高并发如何处理
- 什么是同步锁
- 什么是乐观锁
- 什么是悲观锁
- sql 如何优化；sql 的执行方式是什么样的；如何知道 sql 是怎么执行的？
- spring 的配置文件
- 数据交互的加密方式
- 内存分配

1.说出数据连接池的工作机制是什么？

J2EE 服务器启动时会建立一定数量的池连接，并一直维持不少于此数目的池连接。客户端程序需要连接时，池驱动程序会返回一个未使用的池连接并将其标记为忙。如果当前没有空闲连接，池驱动程序就新建一定数量的连接，新建连接的数量有配置参数决定。当使用的池连接调用完成后，池驱动程序将此连接标记为空闲，其他调用就可以使用这个连接。

实现方式，返回的 **Connection** 是原始 **Connection** 的代理，代理 **Connection** 的 **close** 方法不是真正关闭连接，而是把它代理的 **Connection** 对象还回到连接池中。

2.接口和抽象类的区别是什么？

Java 提供和支持创建抽象类和接口。它们的实现有共同点，不同点在于：

接口中所有的方法隐含的都是抽象的。而抽象类则可以同时包含抽象和非抽象的方法。

类可以实现很多个接口，但是只能继承一个抽象类

类如果要实现一个接口，它必须要实现接口声明的所有方法。但是，类可以不实现

抽象类声明的所有方法，当然，在这种情况下，类也必须得声明成是抽象的。

抽象类可以在不提供接口方法实现的情况下实现接口。

Java 接口中声明的变量默认都是 **final** 的。抽象类可以包含非 **final** 的变量。 □

Java 接口中的成员函数默认是 **public** 的。抽象类的成员函数可以是 **private**，**protected** 或者是 **public**。

接口是绝对抽象的，不可以被实例化。抽象类也不可以被实例化，但是，如果它包含 **main** 方法的话是可以被调用的。

3.概括的解释下线程的几种可用状态?

线程在执行过程中，可以处于下面几种状态：

- 就绪(**Runnable**):线程准备运行，不一定立马就能开始执行。
- 运行中(**Running**)：进程正在执行线程的代码。
- 等待中(**Waiting**):线程处于阻塞的状态，等待外部的处理结束。
- 睡眠中(**Sleeping**)：线程被强制睡眠。
- I/O 阻塞(**Blocked on I/O**)：等待 I/O 操作完成。
- 同步阻塞(**Blocked on Synchronization**)：等待获取锁。
- 死亡(**Dead**)：线程完成了执行。

4.HashMap 和 Hashtable 有什么区别?

HashMap 和 Hashtable 都实现了 Map 接口，因此很多特性非常相似。但是，他们

有以下不同点：

- HashMap 允许键和值是 **null**，而 Hashtable 不允许键或者值是 **null**。
- Hashtable 是同步的，而 HashMap 不是。因此，HashMap 更适合于单线程环境，而 Hashtable 适合于多线程环境。
- HashMap 提供了可供应用迭代的键的集合，因此，HashMap 是快速失败的。另一方面，Hashtable 提供了对键的枚举(**Enumeration**)。
- 一般认为 Hashtable 是一个遗留的类

5. jsp 九大内置对象?

application page request response session exception out config
pageContext