

Readme

知识储备主要包含以下几个模块（按重要性排序）

- **Java Core**（重点：**Java 集合&工具类、多线程和并发包**）
- **JVM**
- **排序算法**
- **数据库**：Redis、Elastic Search、MySQL；
- **框架**：Spring, SpringMVC, Spring Boot, Hibernate
- **分布式**：Dubbo
- **设计模式**（要结合项目具体的去看，比如**Spring**里有哪些设计模式）
- **计算机网络**
- **算法和数据结构**
- **工具**（Maven、Git、Linux、Docker）

要准备到什么程度？

数据库方面（事物、锁、索引等），其实讲清楚一个问题即可：如何进行SQL调优，比如通过索引，看执行计划即可，如果有其它的优化点，说清楚即可。

Java Core方面，这里给出些诀窍：

1. 能结合ConcurrentHashMap的源代码，说出final,volatile,transient的用法，以及在其中如何用Lock对象防止写并发。
2. 结合一个项目实际，说下设计模式的实践。
3. 多线程方面，能说出Lock或volatile等高级知识点的用法。
4. 这块最取巧：说下GC的流程，以及如何通过日志和Dump文件排查OOM异常，如果再高级些的话，说下如何在代码中优化内存代码。

诀窍点归结成一个：能结合源代码或项目实际，说出些比较资深的问题。

其他需要注意的

Java后台开发千万不要把眼光仅仅局限于三层架构和后台框架, 而要把目光转向

- **分布式系统(非常重要)**
- **大数据处理**
- **多线程编程**
- **数据库性能调优**
- **编译原理**

以Java Core和框架为重点，**Java Core重原理及源码细节**，框架重掌握用法和整体了解（必须结合项目）

作为应届生，**Java**核心的内容>框架

所有知识储备「看书，记录」。其中「框架」和「算法」必须动手（框架必须结合项目，看企业版或用框架改自己的）

具体实施

Java、数据库、网络、操作系统、Linux知识可以边看边练 <https://www.nowcoder.com/plan/springintern>

算法用刷题来积累