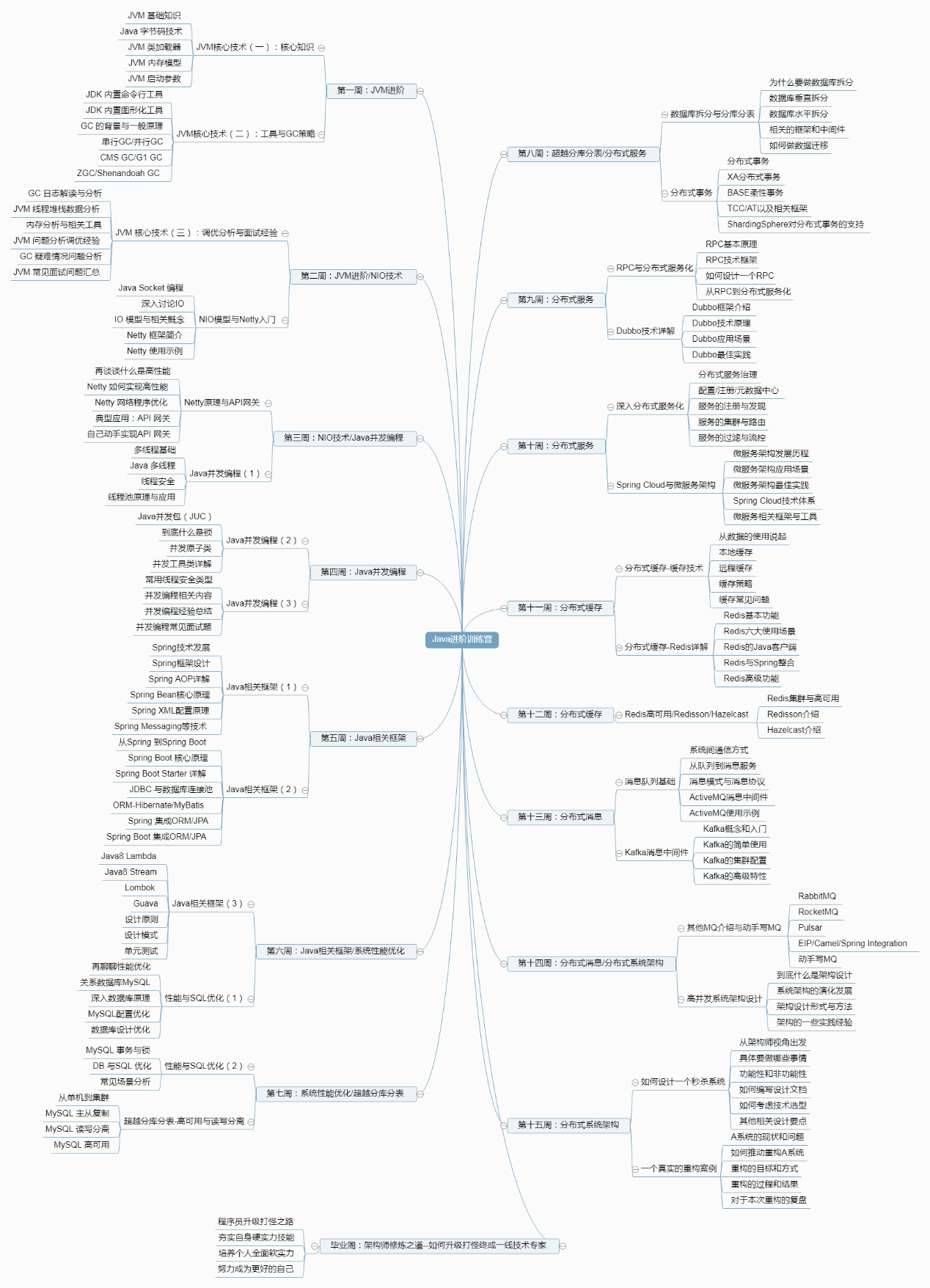
# 一、知识点梳理



# 二：知识点总结与思考

1. JVM进阶知识总结与思考

我们总共用了三节课学习了JVM相关知识，第一节课介绍了JVM基础知识，Java字节码技术，JVM类加载器，JVM内存模型，JVM启动参数，第二节课介绍了相关的工具以及GC算法的介绍，第三节课讲解了具体的GC日志分析等以及一些相关面试问题。让我们系统化的对JVM这一知识点有了系统的了解，系统的去学习了解一个东西，这是在工作过程中很难做到的，所以它给我们在日后工作中起到一个很大的作用，比如：如何在测试或者生产中设置启动参数，OOM后如何抓取日志，用什么样的工具，如何用工具抓取并分析它，最后用已学的知识优化它。

1. NIO技术知识总结与思考

对于NIO这一个模块的学习，说实话印象不是很深刻哈，学的东西还是有点云里雾里的感觉，可能跟实际中我们都用的比较少有关吧，一般情况下我们创建一个http请求啥的，我们要么就是直接用httpClient的方式，要么就是直接用okhttp，要么就是直接用的restTemplate的方式，所以在运行netty通信的示例过程中，总感觉这东西太麻烦，然后可能也介绍的比较抽象些，没有加入一些市面上的通信工具做一些简单比较，这一块东西目前是这种感觉。但网关还是比较印象深刻的，这在后面起到了一个很好的铺垫作用。

1. Java并发编程

并发编程是一块很重要的，也是作为一个开发者来说必不可少的一块掌握的内容。老师在课上带我们从多线程的基础知识了解，再到多线程编程以及线程安全的了解，然后到线程池的掌握，最后面又了解了并发包，锁，原子类，相关的工具以及类型的介绍，很好的让我们对并发编程有了一个基本的认识，在工作过程中掌握这些基础知识再去学习应用能够让我们对自己的并发编程有一定的提高。

1. Java相关框架

Java框架作为java开发者的入门知识之一，相信大家多多少少都会有些了解了。老师从spring背景介绍开始，到spring框架设计，然后介绍了spring核心的aop和ioc等等。之后介绍了orm框架，开发过程中的一些工具类，设计模式等，很实用的介绍了开发过程中需要掌握的一些东西，很全面，开发者必备，要反复查缺补漏的知识点。

1. 性能与SQL优化以及分库分表

性能在工作中是容易被忽略的东西，特别是在中后台管理系统中，几乎是被忽略不计了。它是一个非功能性需求，但随着业务量上来，或者作为一个核心交易系统，那么这个东西就显得尤为重要了，因为这时候几十毫秒和几十秒将是直接致命的影响。老师从性能出发，介绍到数据库的原理，之后介绍到mysql以及数据库设计优化，sql优化，再到现在互联网时代的分布式集群的概念，让我们有了个系统性的了解，并且给我们打开了现代互联网分布式的一层面纱，以至于不再一脸懵逼。

1. 分布式服务

介绍了分布式事务，XA分布式事务以及柔性事务相关概念，有了个基本了解。需要多多考虑在实际工作中是怎么应用的。比如：一个系统如果用到了多数据库，此时要保证数据的一致性，那么久需要多多研究这块专题了。再比如数据库重构，也需要用到该知识。再到分布式服务，随着应用服务的增加，我们都在讲微服务，那么微服务到底是啥？老师从RPC原理开始介绍，再到流行的Dubbo技术介绍，Spring Cloud介绍，以及微服务架构，一步步深入，带我们领受了一遍大方向。具体深入了解应用，也可以基于此深入研究。

1. 分布式缓存

缓存技术也是提升性能必备的知识之一。老师从数据的使用开始介绍，到本地缓存，远程缓存以及介绍了缓存策略，最后用redis详细介绍了分布式缓存的使用以及Redisson/Hazelcast等介绍。

1. 分布式消息

从系统间通信方式介绍开始，再到了解消息模式以及消息协议，然后介绍到当前热门的消息技术中间件ActiveMQ，Kafka，RocketMQ等，打开了该专题的一扇窗，无论是工作中又或是学习，都可以再深入进一步去学习，然后掌握它，真正成为自己的东西。对，手写一个MQ就是我们的目标，从学到用，到懂得如何用了，也就掌握的差不多了。

1. 分布式系统架构

最后做了一个升级，让我们站在更高的角度去看到如何去架构一个系统，设计一个系统，我们需要做哪些事情，功能性需求和功能性需求如何综合考虑，包括如何编写设计文档，考虑技术选择性，还举了个真实的重构案例，告诉我们如何打怪升级，加油，这是我们的大目标。