

Java学习架构(JDK11)

主要学习途径: leetcode, runoob, juejin.im, cnblogs

1. 基础知识

1. Java运行环境熟悉, 了解基本数据类型, 运算, 流程控制, 数组(leetcode官网练习)

2. 面向对象设计思想

1. 继承、多态、封装
2. 类的结构
3. 通过类改造1.1的学习案例
4. 枚举、包

4. 接口、抽象类

1. 通过接口改造1.2.3学习案例
2. 通过抽象类改造1.2.3学习案例

6. 异常

1. 总体error和exception架构学习
2. 内置异常、自定义异常
3. 捕获、抛出、记录异常
4. 通过异常规范3.x学习案例

8. 集合

1. 总体集合架构概览
2. 自顶而下学习常用集合(列表、map、tree、链表)
3. 常用排序、查找算法
4. 概览集合源码自身的排序、查找方式
5. 通过集合改造4.4案例

10. I/O

1. 总体I/O架构概览
2. 自顶而下学习常用I/O(字节流和字符流)
3. 完成一个小玩意, 诸如文件复制、文件搜索

12. XML / JSON文件操作

1. 使用JDK自带的类完成XML、JSON文件的读取、修改、写入磁盘

14. 注解 / 反射

1. 使用反射结合注解完成模板代码的优化

16. 多线程

1. 从操作系统到用户态的多线程模型概览
2. 使用多线程
3. 同步多线程
4. 网上参考消费者模型学习
5. 使用JDK自带的多线程Executor

18. 数据库

1. 关系型数据库、非关系型数据库的基本概念
2. 关系型数据库的基本使用
3. JDK Driver操作数据库
4. 参考Apache DBUtils包优化10.3案例
5. 阅读Apache DBUtils包源码, 并理解其设计思想

20. MyBatis

1. OR映射理念

2. MyBatis自身设计架构
3. 使用MyBatis将10.3案例改造
4. 使用MyBatis将7.1案例改造成文件和数据库数据的两两交互

3. 应用知识

4. 1. Maven / Gradle

2. 1. 包管理
2. 内置命令
3. 插件

3. Junit

4. 1. 单元测试理念
2. 应用

5. IOC

6. 1. 依赖注入理念
2. 学习spring-ioc和google-ioc, 并比较一下
3. 学习spring core(xml、注解)
4. 静态代理、动态代理、CGLIB代理
5. 学习spring aop

7. HTTP

8. 1. 参考《HTTP权威指南》

9. Servlet

10. 1. 理念, 以及现代对其的替代品
2. 生命周期
3. 过滤器、监听器
4. 参考《架构探险》手写一个Servlet服务器

11. Spring MVC

12. 1. MVC的前世今生
2. Spring MVC的总体架构
3. 学习内置的Handler、Adapter
4. 完成路径映射、参数传递、REST

5. 高级应用

6. 1. Spring全家桶
2. 缓存
3. NoSQL
4. 反向代理
5. Linux常用操作
6. 设计模式及应用
7. 数据库中间件

4. 网络(TCP, Socket, HTTP, RPC)