

百战程序员 6000 集

影响 6000 万学习 IT 的中国人

6000 集涵盖：人工智能(350 集)、Python(900 集)、JAVA(1101 集)、大数据(750 集)、互联网架构(300 集)、C 语言(199 集)、PHP(700 集)、区块链(500 集)、前端开发(700 集)、中小学编程(500 集)等多个专业领域，也会随着行业发展推出更多系统化的课程。让初学者一套课程就能成为所学专业的高手。

市面上大量视频课程都属于零散性质，不利于真正掌握一门技术。6000 集本着系统化的思维，让零基础的同学也可以步步深入，跟着一集集系统学习，直达高手境界。

注重效率只讲干货，我们将用心做好每一集视频，甚至精确控制到“帧”为单位，让每个人在编程学习上节省 1000 小时。

CEO: 高 淇 gaoqi1110@163.com

学习要点和建议：

1. 《6000 集》集合尚学堂和百战程序员所有高端课程，根据开班时间持续更新，请关注网站：www.bjsxt.com(尚学堂官网)、www.sxt.cn(速学堂平台)
2. 普通人只需要选择一个领域即可。没有必要将所有领域全部学习一遍。当然，在校大学生时间充裕，未尝不可，成为超级大神迷倒身边众多妹子，不是坏事。成为超级大神，迷倒你的大学老师，更令人神往。
3. IT 时代，编程终将成为每个人需要的技能，希望大家分享给身边的人，让更多人受益！

JAVA 全套 1101 集(从零直达年薪 20 万)

预科阶段：职业规划和学习方法,不再迷茫（14 集）

本阶段不需要编程，是让初学者了解程序员这个行业，了解 JAVA 的技术体系，用通俗的示例告诉大家什么是编程，并且告诉大家职业如何规划，让初学者不再迷茫。如果已经对自身职业规划比较清晰，可以跳过直接进入第一阶段。

1. 程序员的基本素养和职业规划

1. 为什么需要学编程，什么样的人可以做程序员？
2. 流行的编程语言，应该怎么选择？
3. 本行不行，我可以转行到 IT 开发吗？
4. 程序员不是青春饭，程序员迈向百万年薪之路
2. JAVA 技术体系介绍和学习方法
 1. 零基础学员学习 JAVA 的秘诀？
 2. 十分钟秒懂编程，程序就是现实逻辑的表达
 3. 表格思想_理解数据库存储数据
 4. 表格思想_理解 JAVA 类和对象
 5. 表格思想_理解 XML 文档
3. 就业和找工作需要注意事项
 1. 写出一份吸引面试官的简历（六大要素）
 2. 面试前，三大步让你秒杀 90% 竞争者
 3. 面试中，秒吸引面试官的十大要点
 4. 技术面试中，不该问的 4 个问题
 5. 职场成功的核武器_7A+软实力理论

第一阶段：JAVA 快速入门（100 集）

本阶段是进入“程序员”的门槛，需要学习编程最基本的知识：变量、数据类型、控制语句、面向对象。我们通过实际的案例，让大家一开始就通过游戏项目进入学习状态，寓教于乐，引起大家的兴趣。“兴趣永远是最好的老师，高淇老师只能排第二”。

同时，我们也深入内存底层，打扎实大家的基本功。本阶段内容学完，你就明白各个语言底层其实差异都不大。大家再去学其他编程语言也很容易入门。

1. JAVA 入门和第一个项目
 1. 计算机发展史和未来方向
 2. 多种编程语言介绍
 3. JAVA 的发展历史和未来
 4. JAVA 的核心优势和生态体系
 5. JAVA 三大版本含义_JAVAME 和安卓开发的关系
 6. JAVA 的特性和优势
 7. JAVA 应用程序的运行机制
 8. JDK 和 JRE 和 JVM 的作用和区别
 9. JDK 下载和安装_JDK 目录介绍
 10. JDK 环境变量 PATH 设置_classpath 问题
 11. 控制台测试 JDK 安装和配置成功
 12. 写出第一个 JAVA 程序
 13. 第一个 JAVA 程序错误总结
 14. 第一个 JAVA 程序详细解析
 15. 常用 DOS 命令
 16. 常用开发工具介绍
 17. eclipse 开发环境使用原因
 18. eclipse 下载版本选择和使用
 19. eclipse 下建立 JAVA 项目_项目的结构

20. eclipse 开发和运行 JAVA 程序
21. 37 分钟零基础开发桌球小游戏项目
2. 变量和数据类型和运算符
 1. 注释(单行注释、多行注释、文档注释)
 2. 标识符规则_驼峰规则
 3. JAVA 的关键字保留字
 4. 变量的本质_声明格式和初始化问题
 5. 变量分类(局部变量、成员变量、静态变量)
 6. 常量和 final
 7. 基本数据类型介绍
 8. 整型变量和整型常量
 9. 浮点型变量和 BigDecimal 的使用
 10. 字符型类型_转义字符
 11. 布尔型_if 语句使用要点
 12. 运算符介绍
 13. 算术运算符(二元、自增和自减)
 14. 赋值和赋值扩展运算符
 15. 关系运算符
 16. 逻辑运算符_短路运算详解
 17. 位运算符
 18. 字符串连接符
 19. 条件运算符
 20. 运算符优先级_逻辑与或优先问题
 21. 自动类型转化
 22. 强制类型转换
 23. 基本类型常见错误_溢出_L 问题
 24. 使用 Scanner 获得键盘输入
3. 流程控制语句和方法
 1. 控制语句和现实逻辑对应
 2. if 单选择结构
 3. ifelse 双选择结构
 4. Ifelseif...else 多选择结构
 5. switch 语句的使用
 6. while 循环结构
 7. for 循环详解_dowhile 简介
 8. 嵌套循环_循环相关练习
 9. break 和 continue 语句
 10. 带标签的 break 和 continue
 11. 方法的定义_形参_实参_返回值_语句块
 12. 方法的重载 overload
 13. 递归算法_递归头_递归体
4. 面向对象初步
 1. 面向过程和面向对象区别
 2. 对象是什么_对象和数据管理

3. 对象和类的关系_属性_成员变量_方法
 4. 一个典型类的写法和调用_类的 UML 图入门
 5. 内存分析详解_栈_堆_方法区_栈帧_程序执行的内存变化过程
 6. 构造方法详解_构造方法重载
 7. 垃圾回收机制介绍_垃圾回收算法(引用计数和引用可达)
 8. 通用分代垃圾回收详解(年轻代, 年老代, 持久代)
 9. this 的本质_对象创建过程的 4 步
 10. static 关键字_内存分析静态变量和静态方法
 11. 静态初始化块_继承树的追溯
 12. JAVA 的参数传值机制_值传递
 13. java 包机制_package 的使用_JDK 提供的常见包的作用
 14. import 详解_静态导入详解
 15. 面向对象三大特征_继承_instanceOf 的使用
 16. 继承_方法的重写 override_返回值类型问题
 17. 继承_Object 类的用法_重写 toString 方法
 18. 继承_equals 方法的重写_阅读 Object 和 String 的源码
 19. 继承_super 父类对象引用_继承树追溯问题
 20. 面向对象三大特征_封装的使用_访问控制符
 21. 封装_使用细节_javabeen 的规则
 22. 面向对象三大特征_多态_多态的三个必要条件
 23. 对象的转型_向上自动转型_向下强制转型_ClassCastException 处理
 24. final 修饰变量和方法 and 类
5. 数组初步
1. 数组的使用_内存状态_动态初始化和遍历_数组越界异常
 2. 数组的 3 中初始化方式_静态初始化_动态初始化_默认初始化
 3. 数组的遍历_foreach 循环
6. 打飞机小项目训练
1. 游戏项目介绍_建立游戏主窗口
 2. 图形绘制_文本绘制_颜色改变_图像对象的加载
 3. 线程内部类实现动画
 4. 游戏物体根类的实现
 5. 面向对象思想重构飞机类设计
 6. 键盘控制游戏物体原理
 7. 面向对象重构飞机类的键盘控制代码
 8. 炮弹类设计_任意角度飞行
 9. 容器或数组产生多发炮弹
 10. 双缓冲解决闪烁问题_矩形检测原理
 11. 炮弹和飞机的碰撞检测_飞机死掉
 12. 爆炸类_图片数组轮播处理
 13. 主窗口画出爆炸
 14. 飞机死亡和计时功能
 15. 总结和期望

第二阶段：JAVA 基础深化和提高（130 集）

1. 面向对象深化

1. 抽象类和抽象方法
2. 接口详解
3. 接口的多继承
4. 内部类
5. String 类常用方法_API 文档阅读
6. String 类和常量池
7. 字符串相等的判断
8. 设计模式_回调的实现_模板方法模式
9. 设计模式_组合模式

2. 数组深化

1. 多维数组
2. Object 数组存储表格数据
3. 数组的拷贝
4. java.util.Arrays 工具类的使用
5. 冒泡排序的基础算法
6. 冒泡排序的优化算法
7. 二分法

3. 异常机制

1. 异常概念_分类
2. 异常处理方式之一_捕获异常
3. 异常处理方式之二_声明异常
4. 常见简单异常的解决办法
5. 自定义异常
6. 如何利用百度解决异常

4. 常用类

1. 包装类的基本用法
2. 自动装箱和拆箱
3. String 类底层分析_JDK 源码分析
4. StringBuilder、StringBuffer 用法_JDK 底层源码分析
5. 不可变字符序列和可变字符系列使用陷阱
6. java.util.Date 类
7. DateFormat 和 SimpleDateFormat 类
8. Calendar 日历类
9. Math 类
10. File 类
11. 递归算法遍历目录结果和树结构
12. 枚举

5. 容器

1. 泛型
2. 自定义泛型和容器中使用泛型
3. 基本概念_Collection_Set_List 接口介绍

4. List 接口_ArrayList_用法_详解
5. 模拟 ArrayList 容器的底层实现
6. ArrayList 的 JDK 源代码分析
7. LinkedList 特点和底层实现（链表）
8. Vector 用法和 ArrayList 区别_JDK 源码分析
9. Map 接口_HashMap_HashTable 用法详解
10. HashMap 底层实现
11. 二叉树和红黑树
12. TreeMap 的使用和底层实现
13. Set 接口_HashSet 的用法
14. HashSet 底层实现_JDK 源码分析
15. TreeSet 底层实现_JDK 源码分析
16. 迭代器_遍历 List_Set_Map
17. 容器遍历方式总结
18. Collections 工具类

6. IO 流技术

1. IO 流的基本概念
2. 第一个简单的 IO 流程序
3. 流概念细分和体系_四大抽象类
4. 文件字节流
5. 文件字符流
6. 缓冲字节流_缓冲字符流
7. 字节数组流
8. 数据流
9. 转换流
10. 对象流
11. 序列化和反序列化
12. 设计模式_装饰器模式_IO 流整体架构体系
13. Apache IOUtils 的使用
14. Apache FileUtils 的使用

7. 多线程技术

1. 多线程基本概念_程序_进程_线程
2. 通过继承 Thread 类实现多线程
3. 通过接口实现 Runnable 接口实现多线程
4. 线程状态
5. 终止线程的方法
6. 暂停线程执行 sleep/yield/stop
7. 获取线程基本信息的方法
8. 线程的优先级问题
9. 线程同步_具体实现
10. 死锁
11. 死锁的解决办法
12. 线程并发协作原理
13. 生产者消费者模式的实现

14. 任务定时调度

8. 网络编程

1. 网络编程基本概念
2. IP 和端口和 Socket 含义
3. TCP 和 UDP 协议区别
4. TCP 协议数据传递细节
5. UDP 协议数据传递细节
6. InetAddress 类
7. InetSocketAddress 类
8. URL 类
9. 基于 TCP 协议的 Socket 编程_单向通信实现
10. 基于 TCP 协议的 Socket 编程_双向通信实现
11. 结合多线程实现_客户端和服务段任意通信
12. UDP 通信_DatagramSocket 实现

9. XML 技术

1. XML 概念和体系
2. XML 基本语法
3. Schema 技术介绍
4. XML 存储表格数据
5. DOM 和 SAX 方式解析 XML 数据
6. JDOM 解析 XML 数据
7. XPATH 技术_快速获取节点

10. 反射技术

13. 反射机制介绍_Class 对象获取
14. 反射机制动态操作_方法_属性_构造器
15. 提高反射效率
16. 操作泛型
17. 注解_自定义注解
18. 反射读取注解信息

11. 多线程和并发编程

1. 线程知识回顾_并发编程模型
2. JAVA 内存模型
3. 线程同步
4. 线程通信
5. JAVA 中的锁机制
6. 阻塞队列
7. 线程池
8. ThreadLocal 是什么_JDK 底层原理
9. ThreadLocal 的使用场景

9. 手写服务器项目

1. 预备_Socket 知识回顾
2. 预备_多线程实现聊天室代码

3. 预备_html 简单入门
4. 预备_http 协议入门
5. Tomcat 服务器的快速使用
6. Tomcat 服务器运行原理
7. 预备_http 工具查看网络交互过程
8. 手写服务器_整体架构和接口
9. 封装 Response
10. 封装 Request_method_url
11. 封装 Request_储存参数_处理中文
12. 封装分发器
13. 多请求处理_多态
14. 多请求处理_反射
15. xml 配置文件_sax 解析
16. 整合最终版
17. 总结和期望

第三阶段：数据库编程和设计（100 集）

1. Mysql 数据库的使用

1. 数据库相关概念
2. MySQL 基本特点
3. MySQL 的安装和卸载
4. MySQL 客户端
5. MySQL 常用命令

2. Oracle 数据库的使用

1. Oracle 入门
2. Oracle 的安装和卸载
3. Oracle 目录结构和系统用户
4. Oracle 客户端
5. Oracle 常用命令
6. Oracle 的连接配置（listener.ora 和 tnsnames.ora）

3. SQL 语言

1. SQL 语言基础
2. select 子句
3. 去除重复行
4. where 条件查询
5. order by 排序
6. 使用函数
7. group by 分组
8. having 分组筛选
9. DML-insert+update+delete
10. SQL99 连接查询-交叉连接和自然连接
11. SQL99 连接查询-Using 连接和 on 连接
12. SQL99 连接查询-外连接-1
13. SQL99 连接查询-外连接-2

14. SQL99 连接查询-自连接
15. SQL92 连接查询
16. 子查询-单行子查询
17. 子查询-多行子查询
18. 用户 user 和权限
19. 数据库表 table
20. 数据库表的约束 constraints
21. 序列 sequence
22. 索引 index
23. 视图 view
24. 事务
25. rowid 和 rownum
26. 分页查询
27. 导入导出数据
28. 面试题中的 SQL 语句
29. 面试题中的 SQL 语句
30. 面试题中的 SQL 语句
4. 数据库设计范式
 1. 数据库设计范式入门
 2. 第一范式
 3. 第二范式
 4. 第三范式
 5. 数据库设计范式总结
5. JDBC 技术和 ORM 基本思想
 1. JDBC_设计架构_驱动类加载_建立 Connection_效率测试
 2. JDBC_statement 接口用法_SQL 注入
 3. JDBC_PreparedStatement 用法_占位符_参数处理
 4. JDBC_ResultSet 结果集用法_游标原理_关闭连接问题
 5. JDBC_批处理 Batch_插入 2 万条数据的测试
 6. JDBC_事务概念_ACID 特点_隔离级别_提交 commit_回滚 rollback
 7. JDBC_时间处理_Date_Time_Timestamp 区别_随机日期生成
 8. JDBC_时间操作_时间段和日期段查询
 9. JDBC_CLOB 文本大对象操作
 10. JDBC_BLOB_二进制大对象的使用
 11. JDBC_代码总结_简单封装_资源文件 properties 处理连接信息
 12. ORM 原理_使用 Object 数组存储一条记录
 13. ORM 原理_Map 封装一条记录_Map 和 List 封装多条记录
 14. ORM 原理_使用 javaBean 对象封装一条记录
 15. Apache DBUtils 类的使用_完成数据增加和修改和删除操作
 16. Apache DBUtils 类的使用_完成单行数据_多行数据查找
6. Mybatis 框架
 1. 框架是什么_为什么框架都要有配置文件_如何把项目导出成 jar
 2. 为什么企业开发都使用框架
 3. 著名的 IT 企业都在使用哪些框架_这些框架都能做什么

4. MyBatis 简介_什么是 MyBatis
5. MyBatis 环境搭建实现控制台打印查询结果
6. XML 映射文件_environments 标签及子标签配置
7. XML 映射文件 mappers 映射器配置
8. Mapper XML 文件 namespace 命名空间作用和查找规则
9. Mapper XML 文件<select>标签的使用及 resultType 解释
10. MyBatis 结果集映射 Auto Mapping 及列名和属性名不一致解决
11. MyBatis 实现查询_返回 Object
12. MyBatis 实现查询_返回 Map
13. MyBatis 实现查询_返回 List
14. MyBatis 映射文件_properties 标签实现软编码
15. MyBatis 映射文件_typeAliases 标签配置包别名和类别名
16. MyBatis 映射文件_settings 标签设置全局开关
17. MyBatis 对日志支持_打印运行过程中 SQL 命令
18. Mapper XML 文件 parameterType 属性传递不同类型参数
19. MyBatis 事务支持
20. Mapper XML 文件 insert 标签,delete 标签,update 标签
21. Mapper XML 文件 resultMap 标签_联合查询_关联类对象
22. Mapper XML 文件 resultMap 标签_联合查询_关联类集合对象
23. Mapper XML 文件 resultMap 标签_N+1_关联类对象
24. Mapper XML 文件 resultMap 标签_N+1_关联集合对象
25. MyBatis SqlSession 缓存
26. MyBatis 二级缓存
27. 动态 SQL_if 标签
28. 动态 SQL_choose 标签
29. 动态 SQL_trim 标签
30. 动态 SQL_where 标签
31. 动态 SQL_set 标签
32. 动态 SQLt_foreach 标签
33. 动态 SQL_bind 标签
34. Mapper XML 文件 sql_include 标签
35. Mapper XML 文件_selectKey 标签
36. MyBatis 接口绑定方案
37. MyBatis 常用注解
38. MyBatis 运行原理

第四阶段：网页编程和设计（88 集）

1. HTML5

1. 互联网三大基石
2. HTML 入门
3. Head 标签的子标签
4. 基本标签
5. 超链接标签

6. 图片标签
 7. 列表标签
 8. 表格标签
 9. 表单标签 1
 10. 表单标签 2
 11. Frameset
 12. iframe
 13. 层 div 标签
 14. HTML5 介绍
 15. HTML5 新增结构标签
 16. HTML5 增强表单标签
 17. HTML5 音频视频标签
 18. HTML5 绘图和多媒体
2. CSS3
1. CSS 入门
 2. CSS 选择器 1
 3. CSS 选择器 2
 4. 引入 CSS 样式
 5. CSS 常用属性 1
 6. CSS 常用属性 2
 7. CSS 常用属性 3
 8. 盒子模型
 9. CSS3 新增选择器 1
 10. CSS3 新增选择器 2
 11. CSS3 新增属性 1
 12. CSS3 新增属性 2
 13. CSS3 新增属性 3
 14. 案例：制作 12306 网页 1
 15. 案例：制作 12306 网页 2
 16. 案例：制作 12306 网页 3
3. Javascript 语言
1. JS 定义、特点、执行过程
 2. JS 的声明和引入
 3. 数据类型
 4. 变量
 5. 运算符
 6. 控制语句
 7. 自定义函数
 8. 内置函数
 9. String 对象
 10. Date 对象
 11. Array 对象
 12. 事件机制 event
 13. BOM 和 DOM 入门

14. BOM 对象-window
15. BOM 对象-location、history、navigator
16. DOM 结构 节点类型
17. DOM 编程-操作属性
18. DOM 编程-操作样式
19. DOM 编程-操作文本和值
20. DOM 编程-操作元素 1
21. DOM 编程-操作元素 2
22. DOM 编程示例 1
23. DOM 编程示例 2
24. 表单元素操作
25. 表单验证
26. 正则表达式
27. 案例：制作购物车页面 1
28. 案例：制作购物车页面 2
29. 案例：制作购物车页面 3
30. 自定义对象 prototype

4. JQuery

1. jQuery 概述
2. 认识 jQuery 的\$
3. jQuery 的选择器—基本选择器
4. jQuery 的选择器—属性选择器
5. jQuery 的选择器—层级和位置选择器
6. jQuery 的选择器—表单和表单对象选择器
7. 操作页面的样式 1
8. 操作页面的样式 2
9. 操作页面元素的属性 1
10. 操作页面元素的属性 2
11. 操作页面的文本和值 1
12. 操作页面的文本和值 2
13. 操作页面的元素 1
14. 操作页面的元素 2
15. 操作页面的元素 3
16. 事件处理 1
17. 事件处理 2
18. 动画功能 1
19. 动画功能 2
20. jQuery 插件
21. jQuery 封装原理
22. 案例：制作购物车页面 1
23. 案例：制作购物车页面 2
24. 案例：制作购物车页面 3

第五阶段：JAVAE 和项目开发（141 集）

1. Servlet 技术

1. HTTP 协议详解
2. HTTP 请求的基本格式
3. HTTP 请求方式 get 和 post 的区别
4. HTTP 响应的基本格式
5. Tomcat 服务器下载和初步使用
6. Myeclipse 下建立自己的第一个 web 项目并发布到服务器上
7. Servlet 体系架构和接口介绍
8. 第一个 Servlet 程序的编写配置和运行
9. Servlet 运行原理和服务过程
10. Servlet 常见配置错误
11. Request 对象基本用法
12. Response 对象基本用法
13. 请求转发和 request 作用域
14. 重定向和请求转发的区别
15. web.xml 配置文件
16. server.xml 配置文件
17. Session 对象原理
18. Session 对象使用练习
19. Session 作为作用域的保存信息
20. Cookie 对象原理
21. Cookie 对象使用练习
22. ServletConfig 对象用法
23. ServletContext 对象的用法
24. Request、Session、ServletContext 作为作用域的区别

3. JSP 技术

1. JSP 技术介绍_第一个 JSP 页面
2. 分析 JSP 生成的 java 源代码
3. Java 代码块的使用
4. JSP 运行原理深入剖析
5. Page 指令
6. 静态导入和动态导入的区别
7. 三种注释方式
8. include 和 forward 标签
9. pageContext、request、session、application 的区别
10. 九大对象的本质

4. 用户注册和登录项目案例【网站系统】

1. MVC 模式
2. 完成数据库的设计
3. Javabean 类_User 类的设计_封装数据
4. 页面模板介绍
5. 注册页面_reg.jsp

6. Controller 层_RegServlet 接收客户数据
7. DAO 层_RegDao 实现插入数据库操作
8. Service 层_RegService 用于应对业务逻辑变化
9. 项目三层架构运转流程
10. 中文乱码问题的解决
11. 登录页面_login.jsp
12. Controller 层_LoginServlet 接收登录数据
13. Service 层_LoginServlet 应对登录逻辑变化
14. DAO 层_LoginDao 实现对数据库的查找
15. 退出系统_功能实现
16. 总结和期望

2. 标签库

1. EL 表达式基本用法
2. EL 表达式读取作用域中的值
3. 标签库介绍_核心标签库
4. <c:if>标签
5. <c:choose>标签
6. <c:each>标签
7. <c:import>标签
8. 使用标签库生成全国所有省的下拉列表

3. 过滤器和监听器

1. 过滤器 Filter 原理
2. 过滤器的使用
3. 过滤器_解决中文问题
4. 过滤器_图片统一水印
5. 过滤器_权限统一管理
6. 监听器的原理和分类
7. 监听器的使用
8. 监听器_在线人数统计

4. Ajax 异步技术和 json

1. json 表示数据
2. Ajax 技术原理_第一个 Ajax 程序
3. JQuery 实现 Ajax 交互_1
4. JQuery 实现 Ajax 交互_2
5. 使用 ajax 技术_实现省市县三级联动_省级数据加载
6. 使用 ajax 技术_实现省市县三级联动_市级数据加载
7. 使用 ajax 技术_实现省市县三级联动_县级数据加载
8. 使用 ajax 技术_简拼自动查找信息_1
9. 使用 ajax 技术_简拼自动查找信息_2
10. 使用 ajax 技术_简拼自动查找信息_3

5. 项目：财务管理系统

1. 演示界面原型和理解数据库设计

2. 搭建项目框架
3. 单表操作-登录功能的实现和测试
4. 单表操作-登录完善和注销功能
5. 单表操作-添加管理员
6. 用户管理模块（5 集）
7. 组织管理模块（5 集）
8. 支出模块（5 集）
9. 收入模块（5 集）
10. 工资核算模块（5 集）
11. 利润核算（5 集）
12. 手机验证码登录
13. 数据日志模块（5 集）
14. 报表(日报表、月报表、年报表)生成（8 集）
15. 图表和综合报表（5 集）
16. 借款申请和审批模块（8 集）
17. 项目总结

第六阶段：SSM 框架和项目开发（100 集）

1. Spring 框架

1. Spring 简介和官方压缩包介绍
2. Spring3_Spring4_Spring5 差异
3. Spring 环境搭建_演示使用 Spring 管理对象
4. Spring 核心_IOC 控制反转是什么
5. BeanFactory 和 ApplicationContext 区别
6. Bean 生成方式_构造方式
7. Bean 生成方式_实例工厂
8. Bean 生成方式_静态工厂
9. Spring 核心_DI 依赖注入
10. Bean Scopes_对象域属性几种可取值_单例_多例等
11. Spring 提供的注入方式_设值注入
12. Spring 提供的注入方式_构造注入
13. Spring 设值注入_注入简单数据类型_注入引用数据类型
14. Spring 设值注入_注入数组_Array
15. Spring 设值注入_注入集合_Map_Set_List
16. Spring 设值注入_注入 Properties
17. Spring 设值注入_注入空值_Null
18. Autowire 自动装配_局部配置_全局配置
19. Bean 标签其他常用属性
20. Spring 核心_AOP 面向切面编程是什么
21. 静态代理讲解
22. JDK 动态代理
23. CGLIB 动态代理讲解
24. 基于 AspectJ 方式实现_Before Advice 前置通知

25. 基于 AspectJ 方式实现_After Returning Advice 后置通知
26. 基于 AspectJ 方式实现_After Advice 后置通知
27. 基于 AspectJ 方式实现_Around Advice 环绕通知
28. 基于 AspectJ 方式实现_After Throwing Advice 异常通知
29. 基于 AspectJ 方式通知中获取切点参数的实现方案
30. 基于 AspectJ 方式通知中获取切点返回值的实现方案
31. 基于 Schema-based 方式_前置通知
32. 基于 Schema-based 方式_后置通知
33. 基于 Schema-based 方式_环绕通知
34. 基于 Schema-based 方式_异常通知
35. 基于注解方式实现 Spring AOP
36. Spring 声明式事务和编程式事务讲解
37. Spring 声明式事务_propagation 事务传播行为
38. Spring 声明式事务_isolation 事务隔离级别
39. Spring 声明式事务_rollback-for 事务回滚_read-only 只读
40. Spring 常用注解使用_Bean 关联的注解
41. Spring 常用注解使用_自动注入的注解

2. Spring MVC 框架

1. MVC 介绍和发展历史
2. SpringMVC 简介_为什么使用 SpringMVC 框架_SpringMVC 框架作用
3. Spring 容器和 SpringMVC 容器_父子容器详细解释
4. SpringMVC 运行原理详解_运行原理流程图
5. SpringMVC 源码跟踪_所有核心组件及含义
6. 基于纯配置文件方式搭建 SpringMVC
7. 基于注解方式快速搭建 SpringMVC
8. SpringMVC 中注解驱动 mvc:annotation-driven 详细解释
9. SpringMVC 中资源解析_resources 放行静态资源问题
10. SpringMVC 控制器方法 HandlerMethod_String 作为方法返回值
11. SpringMVC 控制器方法 HandlerMethod_ModelAndView 作为方法返回值
12. SpringMVC 控制器方法 HandlerMethod_Void 作为方法返回值
13. SpringMVC 参数接收_使用变量接收请求参数
14. SpringMVC 常用注解_@RequestParam 使用
15. SpringMVC 参数接收_使用对象接收请求参数
16. SpringMVC 参数接收_复杂类型请求参数接收
17. SpringMVC 参数接收_数组或集合类型参数接收
18. SpringMVC 参数接收_使用 RestFul 方式接收请求参数
19. 获取请求头中参数_@RequestHeader
20. 获取请求体中 JSON 流数据并转换为对象_@RequestBody
21. Spring 核心组件_视图解析器和自定义视图解析器
22. 紧耦 Servlet API_获取 Servlet 中常用的对象
23. SpringMVC 提供的解耦 ServletAPI 实现作用域传值_ModelAndView
24. SpringMVC 提供的解耦 ServletAPI 实现作用域传值_Model
25. SpringMVC 提供的解耦 ServletAPI 实现作用域传值_Map
26. SpringMVC 文件上传

27. SpringMVC 文件下载
28. SpringMVC 返回 JSON 字符串_@ResponseBody_Jackson
29. SpringMVC 中自定义拦截器_拦截全部控制器
30. SpringMVC 中自定义拦截器_拦截特定请求的实现方案
31. SpringMVC 中自定义拦截器_添加拦截器后执行流程
32. SpringMVC 中拦截器栈的执行流程

【注意】：myBatis 在第三阶段讲完 JDBC 后，已经详细讲解。需要的同学往前看。

3. SSM 框架整合

1. Spring JDBC 使用
2. ContextLoaderListener 源码解读.
3. DriverManagerDataSource 源码解读
4. SqlSessionFactoryBean 源码解读
5. MapperScannerConfigurer 源码解读
6. SSM 框架整合实现_不编写 MyBatis 配置文件方式
7. SSM 框架整合实现_保留 MyBatis 配置文件方式

4. Shiro 安全框架

1. Shiro 简介及_架构图介绍
2. Shiro 中名词汇总
3. INI 文件组成_[main]详细解释
4. INI 文件组成_[users]详细解释
5. INI 文件组成_[roles]详细解释
6. INI 文件组成_[urls]详细解释
7. Shiro 环境搭建实现简单认证
8. Shiro 中内置 Realm 讲解_JDBCRealm
9. Shiro 自定义 Realm 实现连接数据库获取数据库中数据
10. Shiro 提供密码加密实现方案_几种加密类型对比
11. Shiro 提供密码加密实现方案_MD5 加密_加盐_迭代
12. Shiro_凭证匹配器
13. Shiro 实现授权_注解触发_标签触发_代码触发
14. DelegatingFilterProxy 源码解读
15. Spring 整合 Shiro 实现登录认证
16. Spring 整合 Shiro_remember me 功能实现
17. 案例演示_实现注册_注册时密码加密
18. Spring 整合 Shiro 实现菜单授权
19. SessionManager 使用
20. 案例演示_实现必须登录可见_内置过滤器介绍

第七阶段：JAVA 高级技术（115 集）

1. Linux

1. Linux 简介_Linux 发展历史_Linux 市场地位
2. VMWare 安装及使用_快照功能_克隆功能
3. Linux 系统目录介绍_常用路径的几种写法
4. Linux 中常用命令(上)

5. Linux 中常用命令(下)
6. XShell5 的安装及使用_客户端操作 Linux 服务器系统
7. FileZilla 安装及使用_上传本地文件到服务器
8. 服务器环境配置_解压版 JDK7 的配置及测试
9. 服务器环境配置_解压版 Tomcat7 的配置及测试
10. 服务器环境配置_解压版 MySQL 的配置及测试
11. 项目部署_部署 WAR 项目到服务器

2. Maven

1. Maven 简介_Maven 和 Ant 对比_Maven 的优点
2. 单机版 Maven 使用_Maven 常用命令介绍
3. Eclipse 集成 Maven 步骤
4. Packaging 的几种类型_WAR_POM_JAR
5. 创建 JAR 类型 Maven Project
6. Maven 项目的生命周期
7. Maven 项目和项目之间的关系_依赖
8. Maven 项目和项目之间的关系_继承
9. Maven 项目和项目之间的关系_聚合
10. 创建 War 类型 Maven Project
11. Tomcat 插件使用_发布 Maven 项目到本地 Tomcat 对比
12. 资源拷贝插件 Resource 使用及作用
13. 案例演示_完整 SSM 项目搭建
14. 私服介绍_使用私服的场景
15. Nexus 搭建_离线索引库配置
16. Maven 连接 Nexus 实现 Jar 下载到本地仓库
17. 上传项目到 Nexus
18. 热部署

3. Dubbo

1. SOA 是什么_为什么使用 SOA
2. RPC 是什么
3. Dubbo 简介_Dubbo 使用前景_Dubbo 发展历史
4. Dubbo 架构图详细解释_Dubbo 执行流程
5. Dubbo 支持的几种 Registry 注册中心类型
6. Zookeeper 简介_Zookeeper 作用
7. Zookeeper 单机版安装及配置
8. Dubbo 支持的几种 Protocol 协议
9. Dubbo Provider 提供者项目架构分析
10. Dubbo Provider 提供者搭建实现发布服务.
11. Dubbo Consumer 消费者项目搭建及实现
12. Assembly 打包插件使用_打包 Provider_发布到服务器
13. Dubbo 自带 Admin 管理项目搭建流程
14. Monitor 监控中心搭建及使用.

4. VSFTPD

1. VSFTPD 简介_使用 VSFTPD 后流程图
2. VSFTPD 安装到 Linux 系统中

3. FTPClient 使用_发布本地项目到服务器
4. FTPClient 封装_FtpUtils 的实现过程
5. KindEditor 简介
6. KindEditor 文件上传组件使用
7. KindEditor 富文本编辑框使用
8. 案例演示_使用 KindEditor 和 FtpClient 上传资源到服务器

5. Nginx

1. Nginx 简介_发展历史_应用场景
2. 反向代理_正向代理
3. Nginx 安装及配置
4. 实现 Nginx 开机自启动
5. Nginx 代理本机目录_外部主机通过 HTTP 访问
6. 案例演示_实现上传图片回显功能
7. Nginx 代理 Tomcat 实现负载均衡

6. Redis

1. Redis 简介_Redis Hash 槽概念介绍
2. Redis 持久化策略_RDB
3. Redis 持久化策略_AOF
4. Redis 单机版安装
5. Redis 命令_KEY 键
6. Redis 命令_String 字符串
7. Redis 命令_Hash 表
8. Redis 命令_List 列表
9. Redis 命令_Set 集合
10. Redis 命令_SortedSet 有序集合
11. Jedis 实现过程_String 类型数据 CURD
12. Jedis 实现过程_HASH 类型数据 CURD
13. Jedis 实现过程_List 类型数据 CURD
14. Jedis 实现过程_Set 类型数据 CURD
15. Jedis 实现过程_SortedSet 类型数据 CURD
16. Redis 集群安装及配置
17. Spring 整合 Jedis
18. Redis Desktop 安装及使用

7. Solr

1. Solr 简介_Solr 实现原理_Solr 执行流程
2. Solr 数据检索方案_其他常用数据检索方案介绍
3. Solr 单机版项目安装
4. Solr 管理界面_主体功能介绍
5. IK Analyzer 中文拆词器介绍
6. 安装 IK Analyzer 中文拆词器到 Solr 项目中
7. Solr 配置文件 SchemaXML 文件内容介绍
8. Solr 管理界面_解析功能演示
9. Solr 管理界面_Document 功能_新增数据
10. Solr 管理界面_Document 功能_修改数据

- 11.Solr 管理界面_Document 功能_删除数据
- 12.Solr 管理界面_Query 功能演示
- 13.SolrJ 演示_新增数据
- 14.SolrJ 演示_删除数据
- 15.SolrJ 演示_修改数据
- 16.SolrJ 演示_查询数据
- 17.SolrCloud 配置及搭建
- 18.SolrJ 演示_操作 SolrCloud
- 19.案例演示_内容搜索

8.ActiveMQ

- 1. JMS 简介_什么是消息_什么是消息队列_常用消息服务应用
- 2. ActiveMQ 简介_ActiveMQ 特征
- 3.ActiveMQ 安装
- 4.ActiveMQ 术语
- 5.ActiveMQ 应用_Helloworld_队列服务监听
- 6.Spring 整合 ActiveMQ 开发

9.JsonP

- 1.跨域数据请求_有多少种跨域请求解决方案
- 2. JsonP 是什么_JsonP 优点_JsonP 缺点
- 3.JsonP 实现_控制器手动编写内容
- 4.JsonP 实现_使用 MappingJacksonValue

10.HttpClient

- 1.HttpClient 简介_使用场景
- 2.HttpClient_POST 方式访问
- 3.HttpClient_Get 方式访问
- 4.HttpClient_参数为 JSON 数据

11.MyCat

- 1.MySql 主从备份介绍
- 2.MySql 主从分离实现
- 3. MyCat 简介_使用 MyCat 的必要性
- 4. MyCat_配置逻辑库_配置逻辑表
- 5.MyCat_配置读写分离
- 6.MyCat_用户信息配置

第八阶段：电商高级项目_架构和编码实现（73 集）

- 1. 电商 Ego-基于 SOA 架构,使用 Dubbo,逆向工程,分页插件完成商品查询
 - 1. 电商行业介绍_人员分工介绍_总体结构介绍
 - 2. 数据库设计_数据库分析
 - 3. MyBatis 逆向工程讲解和使用
 - 4. 基于 SOA 架构创建 Dubbo Provider
 - 5. 基于 SOA 架构创建后台管理项目
 - 6. 分析页面_显示后台项目所有页面

7. 编写代码_商品查询_使用 MyBatis 分页插件完成
8. 编写代码_商品上架
9. 编写代码_商品下架
10. 编写代码_商品删除
2. 电商 Ego-使用 VSFTPD 和 Nginx 完成商品新增
 1. 总体需求分析_功能分析
 2. 编写代码_商品类目查询
 3. 编写代码_实现图片上传
 4. 编写代码_实现商品新增
 5. 编写代码_实现商品修改
3. 电商 Ego-基于 json 格式完成商品规格参数管理
 1. 功能需求分析
 2. 编写代码_实现商品规格参数模版查询
 3. 编写代码_实现选择类目后判断是否已经添加模版
 4. 编写代码_实现规格参数模版新增
 5. 编写代码_实现规格参数模版修改
 6. 编写代码_实现规格参数模版批量删除
 7. 编写代码_商品新增_具体规格参数新增
 8. 编写代码_商品修改_修改商品规格参数内容
4. 电商 Ego-使用 jsonp 完成前台首页导航菜单
 1. 功能需求分析_页面解读_接口文件解读
 2. 创建门户项目
 3. 创建商品管理项目
 4. 编写代码_实现商品类目查询_门户通过 JSONP 获取导航菜单信息
5. 电商 Ego-完成 CMS 系统
 1. 功能需求分析_页面解读
 2. 编写代码_商品类目查询
 3. 编写代码_商品类目新增
 4. 编写代码_商品类目重命名
 5. 编写代码_商品类目删除
 6. 编写代码_类目对应内容查询
 7. 编写代码_类目中内容新增
 8. 编写代码_类目中内容删除
 9. 编写代码_类目中内容修改
6. 电商 Ego-添加前台大广告位数据缓存
 1. 功能需求分析_页面解读
 2. 编写代码_前台大广告内容查询_未添加缓存方式
 3. 编写代码_前台大广告内容查询_添加 Redis 缓存
 4. 编写代码_后台商品管理新增_缓存同步
 5. 编写代码_后台商品管理修改_缓存同步
 6. 编写代码_后台商品管理删除_缓存同步
7. 电商 Ego-完成商品搜索功能
 1. 功能需求分析_页面解读
 2. 商品搜索项目创建

3. 编写代码_Solr 数据初始化
 4. 编写代码_搜索项目_商品条件查询
 5. 编写代码_商品项目通过 HttpClient 获取商品数据
 6. 编写代码_显示商品列表
8. 电商 Ego-显示商品详细详情
 1. 功能需求分析_页面解读
 2. 编写代码_点击商品列表显示商品详情主体信息
 3. 编写代码_显示商品描述信息
 4. 编写代码_延迟 1 秒显示商品规格参数
9. 电商 Ego-完成 SSO 单点登录系统
 1. 功能需求分析_页面解读
 2. 创建 SSO 单点登录系统
 3. 编写代码_实现登录功能
 4. 编写代码_实现注册功能
 5. 编写代码_获取登录用户状态
10. 电商 Ego-购物车
 1. 功能需求分析_页面解读
 2. 创建购物车项目
 3. 编写代码_用户未登录状态下添加商品到临时购物车
 4. 编写代码_用户登录后合并购物车
 5. 编写代码_购物车信息显示
 6. 编写代码_购物车商品数量修改
 7. 编写代码_购物车商品删除_商品清空
11. 电商 Ego-订单系统实现
 1. 功能需求分析_页面解读
 2. 创建订单项目
 3. 编写代码_显示订单预览信息
 4. 编写代码_确认订单信息_提交订单信息
12. 电商 Ego-ActiveMQ
 1. 功能需求分析
 2. 添加消息队列后总体分析
 3. 编写代码_在订单系统添加 ActiveMQ
13. 电商 Ego-MySQL 读写分离+MyCat 分库分表
 1. 总体功能介绍_目标实现方案
 2. 搭建电商项目总体数据库环境
 3. 添加 MyCat 配置逻辑库和逻辑表
 4. 实现读写分离
14. 电商 Ego-项目部署
 1. Dubbo Provider 打包发布到服务器
 2. 热部署所有 WAR 到 Linux 服务器
 3. 配置虚拟域名
 4. 总体测试
 5. 总体效果演示

第九阶段：设计模式和框架底层源码分析(190 集，选修)

1. GOF23 设计模式

1. 单例模式_应用场景_饿汉式_懒汉式
2. 单例模式_双重检查锁式_静态内部类式_枚举式_UML_类图
3. 单例模式_反射和反序列化漏洞和解决方案_多线程环境测试
4. 单例模式_双重检查锁式_静态内部类式_枚举式_UML_类图
5. 简单工厂模式_面向对象设计原则_开闭原则_依赖反转原则_迪米特法则
6. 工厂方法模式详解
18. 抽象工厂模式详解
19. 建造者模式详解_类图关系
20. 原型模式_prototype_浅复制_深复制_Cloneable 接口
21. 原型模式_反序列化实现深复制_效率对比_创建型模式总结
22. 适配器模式_对象适配器_类适配器_开发中场景
23. 代理模式_静态代理
24. 代理模式_动态代理_开发中常见的场景
25. 桥接模式_多层继承结构_银行日志管理_管理系统消息管理_人力资源的奖金计算
26. 组合模式_树状结构_杀毒软件架构_JUnit 底层架构_常见开发场景
27. 装饰模式_IO 流底层架构_装饰和桥接模式的区别
28. 外观模式_公司注册流程_迪米特法则
29. 享元模式_享元池_内部状态_外部状态_线程池_连接池
30. 责任链模式_公文审批_供应链系统的采购审批_异常链_过滤器和拦截器调用过程
31. 迭代器模式_JDK 内置迭代器_内部类迭代器
32. 中介者模式_同事协作类_内部类实现
33. 命令模式_数据库事务机制底层架构实现_撤销和回复
34. 解释器模式_访问者模式_数学表达式动态解析库
35. 策略模式_CRM 中报价策略_GUI 编程中布局管理器底层架构
36. 模板方法模式_钩子函数_方法回调_好莱坞原则
37. 状态模式_UML 状态图_酒店系统房间状态_线程对象状态切换
38. 观察者模式_广播机制_消息订阅_网络游戏对战原理
39. 观察者模式_observeable 类和 observer 接口_应用场景总结
40. 备忘录模式_多点备忘_事务操作_回滚数据底层架构

2. Servlet 和 Tomcat 底层源码分析

1. 底层源码分析_Servlet 接口
2. 底层源码分析_ServletConfig 接口
3. 底层源码分析_AbstractGenericServlet
4. 底层源码分析_AbstractHttpServlet
5. 底层源码分析_HttpServletRequest
6. 底层源码分析_HttpServletResponse
7. 底层源码分析_HttpServlet
8. Tomcat 底层源码分析_Bootstrap Loader
9. Tomcat 底层源码分析_Extended Loader

10. Tomcat 底层源码分析_WebappClassLoader
3. Mybatis 底层源码分析
 1. 底层源码分析_Resources
 2. 底层源码分析_XMLConfigBuilder
 3. 底层源码分析_Configuration
 4. 底层源码分析_DefaultSqlSessionFactory
 5. 底层源码分析_SqlSessionFactory
 6. 底层源码分析_Transaction
 7. 底层源码分析_Executor
 8. 底层源码分析_DefaultSqlSession
4. Spring 框架底层源码分析
 1. 底层源码分析_ApplicationContext
 2. 底层源码分析_BeanFactory
 3. 底层源码分析_ListableBeanFactory
 4. 底层源码分析_WebApplicationContext
 5. 底层源码分析_ConfigurableApplicationContext
 6. 底层源码分析_ResourceLoader
 7. 底层源码分析_XmlBeanFactory
 8. 底层源码分析_ClassPathResource
 9. 底层源码分析_AbstractAutowireCapableBeanFactory
 10. 底层源码分析_XmlBeanDefinitionReader
 11. 底层源码分析_EncodedResource
 12. 底层源码分析_AbstractBeanFactory
5. Struts2 框架使用和底层源码分析
 1. Struts2 框架简介_Struts2 发展历史
 2. Struts2 和 Struts1 对比
 3. Struts2 运行原理_运行流程图
 4. Struts2 源码跟踪_StrutsPrepareAndExecuteFilter 核心过滤器
 5. Struts2 实现创建 Action 类的三种方式_POJO 方式
 6. Struts2 实现创建 Action 类的三种方式_实现 Action 接口方式
 7. Struts2 实现创建 Action 类的三种方式_继承 ActionSupport 方式
 8. 底层源码分析_内置常量解读
 9. Struts2 配置文件_常量的值的配置
 10. Struts2 配置文件_package 标签及属性详细解释
 11. Struts2 配置文件_action 标签及属性详细解释
 12. Struts2 配置文件_result 标签及属性详细解释
 13. Struts2 中支持的五种内置逻辑视图名及含义
 14. Struts2 核心功能_属性驱动方式接收请求参数
 15. Struts2 核心功能_模型驱动接收请求参数
 16. Struts2 中 Action 类是否是线程安全的
 17. Struts2 中全局配置文件路径及名称的实现方案
 18. Struts2 中拦截器简介_为什么使用拦截器_和 Filter 的区别
 19. 底层源码分析_Struts2 内置拦截器解读_alias
 20. 底层源码分析 Struts2 内置拦截器解读_chain

21. 底层源码分析_Struts2 内置拦截器解读_conversionError
22. 底层源码分析_Struts2 内置拦截器解读_cookie
23. 底层源码分析_Struts2 内置拦截器解读_debugging
24. 底层源码分析_Struts2 内置拦截器解读_exception
25. 底层源码分析_Struts2 内置拦截器解读_fileUpload
26. 底层源码分析_Struts2 内置拦截器解读_i18n
27. 底层源码分析_Struts2 内置拦截器解读_timer
28. 底层源码分析_Struts2 内置拦截器解读_modelDriven
29. 底层源码分析_Struts2 内置拦截器解读_scopedModelDriven
30. 底层源码分析_Struts2 内置拦截器解读_params
31. 底层源码分析_Struts2 内置拦截器解读_servletConfig
32. 底层源码分析_Struts2 内置拦截器解读_token
33. 底层源码分析_Struts2 内置拦截器解读_validation
34. 使用 AbstractInterceptor 完成自定义拦截器_拦截特定 Action
35. 使用 AbstractInterceptor 完成自定义拦截器_拦截全部 Action
36. Struts2 创建拦截器栈的实现方案
37. Struts2 文件上传_单文件上传
38. Struts2 文件上传_多文件上传
39. Struts2 文件上传_控制上传文件大小_类型
40. Struts2 中 DMI 实现
41. Struts2 配置文件 method 结合通配符*
42. Struts2 配置文件_默认 Action 引用
43. Struts2 配置文件_团队开发_include 标签使用
44. Struts2 配置文件_全局结果 global-results
45. Struts2 配置文件_全局结果异常映射_global-exception-mappings
46. 底层源码分析_ActionContext_Action 容器
47. 底层源码分析_ServletActionContext
48. Struts2 作用域传值_紧耦方式获取
49. Struts2 作用域传值_解耦方式获取
50. Struts2 文件下载
51. Struts2 返回 JSON 结果类型
52. 底层源码分析_JsonResult 转换对象为 json 字符串
53. 服务器端数据验证_局部验证
54. 服务器端数据验证_全局验证
55. 服务器端数据验证_内置校验框架
56. Struts2 核心功能_国际化
57. Struts2 核心功能_OGNL
58. Struts2 核心功能_常用的标签解释
59. Struts2 核心功能_值栈
60. 底层源码分析_ActionProxy
61. 底层源码分析_Action Invocation
62. 底层源码分析_Action Mapper
63. 底层源码分析_ActionChainResult
64. 底层源码分析_ServletDispatcherResult

65. 底层源码分析_ServletRedirectResult
66. 底层源码分析_ServletActionRedirectResult
67. 底层源码分析_StreamResult
68. Struts2 核心功能_防止表单重复提交
6. Hibernate 框架使用和底层源码分析
 1. ORM 概念介绍_为什么使用 ORM
 2. Hibernate 简介_Hibernate 发展历史_Hibernate 的优点_Hibernate 的缺点
 3. Hibernate 环境搭建_快速创建一个 Hibernate 项目
 4. Hibernate 全局配置文件 hibernate.cfg.xml 详解解读
 5. Hibernate 映射文件 XXX.hbm.xml 详细解读
 6. Hibernate 正向工程的实现_hbm2ddl
 7. Hibernate 支持的主键生成策略_如果选择使用哪种生成策略
 8. 底层源码分析_Configuration 总体配置类
 9. 底层源码分析_SessionFactory Session 工厂类
 10. 底层源码分析_Session 封装 HibernateAPI 的核心类
 11. 底层源码分析_Transaction 事务管理类
 12. 底层源码分析_Hibernate4 支持的_StandardServiceRegistry 服务注册
 13. 底层源码分析_Hibernate4 支持的_MetadataSources
 14. Hibernate 功能实现_新增
 15. Hibernate 功能实现_修改
 16. Hibernate 功能实现_删除
 17. Hibernate 功能实现_根据 OID 主键查询
 18. Hibernate 核心功能_延迟加载详细介绍
 19. Hibernate 核心功能_对象声明周期
 20. Hibernate 核心功能_脏检查策略
 21. Hibernate 核心功能_Session 缓存
 22. Hibernate 核心功能_二级缓存_SessionFactory 缓存
 23. Hibernate 核心功能_查询缓存
 24. Hibernate 核心功能_缓存管理_Hibernate 提供缓存管理的 API
 25. Hibernate 核心功能_QBC
 26. Hibernate 核心功能_QBC 离线查询
 27. Hibernate 核心功能_QBE
 28. Hibernate 核心功能_QBE 离线查询
 29. Hibernate 核心功能_对 SQL 支持_投影查询
 30. Hibernate 核心功能_HQL 的使用
 31. Hibernate 核心功能_分页查询的实现方案
 32. Hibernate 核心功能_占位符_命名占位符
 33. Hibernate 核心功能_批量新增和批量删除
 34. Hibernate 核心功能_聚合查询的实现
 35. Hibernate 核心功能_抓取策略_Fetch
 36. Hibernate 核心功能_命名查询
 37. Hibernate 重难点_单向多对一的实现
 38. Hibernate 重难点_单向一对多的实现
 39. Hibernate 重难点_双向多对一的实现

40. 案例演示_基于多对一的自引用实现查询树状菜单内容
41. Hibernate 重难点_单向一对一的实现_双主键作为主外键
42. Hibernate 重难点_单向一对一的实现_非双主键关联
43. Hibernate 重难点_双向一对一的实现
44. Hibernate 重难点_双向多对多_省略第三方关联表的实现方案
45. Hibernate 重难点_双向多对多_创建关联表的实现方案
46. Hibernate 注解_基本实体类注解
47. Hibernate 注解_多对一注解使用
48. Hibernate 注解_一对一注解使用
49. Hibernate 注解_多对多注解使用
50. Hibernate 和 MyBatis 对比

第十阶段：面试和成功求职的秘技（30 集）

1. 简历

1. 为什么有人是“面霸”，有人是“面渣”？
2. 好简历的十要素
3. 好简历_个人基本要素合理写法
4. 好简历_JAVA 程序员的专业技能如何写？
5. 好简历_JAVA 程序员的项目经验如何写？
6. 简历投递的五大招聘网站

2. 面试前的准备

1. 接到面试电话，三要素让你完美应对 HR
2. 面试前，如何调查公司背景
3. 面试前，如何调查公司项目案例和相关技术使用
4. 面试前，准备完美的自我介绍

3. 笔试和面试

1. 常见面试非技术问题应对方案
2. 笔试题如何准备？
3. JAVA200 道面试题如何看？
4. 面试时，面对面试官的礼仪五要素
5. 面试时，表现自己的三大核心优势就成功了
6. 面试时，碰到不会的技术题如何应对？
7. 面试时，扬长避短如何表达自己想要表达的内容
8. 面试时，将自己对公司的了解表达出来
9. 面试时，如何谈薪水
10. 面试时，要不要问五险一金和加班问题？
11. 面试时，如何合理结束本次面试
12. 做好面试总结，让本次面试成为下一次更好发挥的演练

4. offer 和上班后，如何应对

1. 收到 offer，如何正确应对？
2. 体检报告如何去做？
3. 公司报道第一天，必须做的三大事
4. 上班的新菜鸟，一定不能做的八件蠢事
5. 对你的上级，做到 4 点职业准则，就一定能成为职场赢家

6. 你的同事，是菜鸟生存下去的关键
7. 遇到不会的问题，三步让你正确应对
8. 加班，对于菜鸟不是坏事

第十一阶段：入职后如何快速成长到 CTO（20 集）

1. 入职后三个月试用期要做的事
 1. 三法宝，处理同事关系
 2. 核心两点，处理好领导关系
 3. 每件事都是学习的机会
 4. 主动加班，试用期加班是学习的好机会
 5. 未通过试用期，如何应对？
2. 前三年需要学的技术
 1. 工作后，千万不要停止学习
 2. 项目经验如何累积？
 3. JAVA 高级技术还需要学习哪些？
 4. 架构师课程如何学习？
 5. 工作中，快速学习新技术的捷径（重要的是形成体系，而不是钻到某个技术点）
3. 前三年需要提高的软实力
 1. 软实力是真实力
 2. 软实力的七要素，7A 软实力理论
 3. 好的沟通有方法
 4. 快速说服别人的技巧
 5. 口才也能练出来
 6. 领导力如何自我修炼
 7. 会议发言的三要素
 8. 酒宴祝词、会议发言的完美公式
 9. 五点职场礼仪，让你秒变职场受欢迎人士
 10. 软实力是让你脱颖而出的关键

JAVA 的学习路线图

JAVA 语言是一门非常流行和重要的语言，目前仍是需求量很大的语言，应用范围很广的语言，在：企业级开发、移动开发、大数据云计算、人工智能等领域都有大量的应用。

怎么样学习好 JAVA 语言，达到可以轻松就业的地步，是每个初学者都关注的话题。我们一般将 JAVA 编程的学习分为这四个大阶段：基础阶段、WEB 开发阶段、框架阶段、高级技术阶段。各阶段具体学习内容和方法如下：

1. 基础阶段

基础阶段的学习说白了就是 JAVASE 的学习，这是基础的阶段。目标是让初学者进入编程的殿堂，具备基本的编程思维。在这个阶段我们可以在细分为如下内容：

1. JAVA 编程入门

入门阶段主要学习：变量、数据类型、控制语句这些基本的概念。这些技术是所有编程语言都有的内容。很基础也很重要。专心学完，大约需要 3 天时间。

2. 面向对象

面向对象是深入学习 JAVA 的核心，也是大家真正进入 JAVA 世界的难点和重点。这里会有各种各样的概念，非常重要。但是，大家在学习面向对象的时候，先注意“开始学习语法和概念，不要期望一口气学的很透彻。面向对象，需要大家在后面做项目时才能深入体会如何将面向对象思想应用于实践”。因此，我建议大家不要在此处过于纠结概念，能学多少学多少，遇到不懂的概念，先记结论开始用，快速往后学，不要停留过长的时间在此处。如果全天候学习，建议学习时间 3 天。

3. 异常

异常是 JAVA 安全机制的核心，也是让我们程序更易懂更稳定的核心。初学异常重在理解概念、学习如何解决异常的思路。如果全天候学习，建议学习时间 0.5 天。

4. 数组

数组是常用的概念。在学习本内容的时候，掌握核心“数组也是对象，数组的元素相当于对象的成员变量”，理解了这一句话，就掌握了数组的一大半了。数组学习时顺便学习一下一些基本的算法，比如：排序、二分法等。这样，既可以学习数组，也能复习前面的控制语句、面向对象等知识。因此，在此处可以多做练习，停留时间长一点。如果全天候学习，建议学习时间 2 天。

5. 常用类

我们重点掌握常用类的用法，学习 API 的查询和使用。如果你想更深入，学习时可以阅读这些常用类的底层源码，开始培养高手的格局。如果全天候学习，建议学习时间 2 天。

6. 容器

容器是开发中常用的内容。学习容器时，既要学习相关类和接口的使用，也要明白容器的底层原理。容器的底层实现和数据结构知识紧密挂钩。也是，我们在工作面试中经常会碰到的问题。因此，容器需要初学者花大力气学习，并且要深入底层，明白底层的核心机制。如果全天候学习，建议学习时间 2 天。

7. 多线程

多线程的 JAVA 语言的特点。我们在后续学习服务器编程时，服务器的底层核心机制就是基于“多线程”的。因此，我们需要掌握多线程的基本原理和应用。为了后续学习并发编程，也有必要学习一下“生产者消费者模式”、“锁机制”、“ThreadLocal”、“线程池”等高端一点的概念。如果全天候学习，建议学习时间 2 天。

8. 网络编程

更容器的网络编程是 JAVA 在互联网时代击败 C++ 的关键。我们需要掌握 TCP 编程、UDP 编程的经典模式，但是不需要太深入去学习。因为，大多数项目开发时底层都做好了封装。如果全天候学习，建议学习时间 1.5 天。

2. WEB 服务器开发阶段

学习 JAVA，大多数人最终会进入服务器编程、企业级开发领域。因此，为了能够完整的编写项目。这个阶段我们也要学习一些 JAVA 之外的技术：网页开发和设计、数据库编程。JAVA 之内的技术有：Servlet、JSP、标签库、服务器等。这都是大家进入开发领域必须要学习的内容，光会 JAVASE 是不能实现就业、不能独立开发项目的。但是，学完本阶段的技能，大家就可以独立开发一些项目、具备初步的就业能力了。

1. 网页开发和设计

JAVA 程序员主要完成服务器端的开发，但是对于浏览器端如何展现数据也是很有必要掌握的。网页开发的基本知识可以说是必学的技能。集中包括：HTML、CSS、JavaScript、jQuery 等。网页开发也是专门的专业，JAVA 程序员去学习，掌握基本概念即可，不需要深入太多细节。大家在学习的时候，把握好“度”。如果全天候学习，建议学习时间 6 天。

2. 数据库编程和设计

数据库是绝大多数项目的存储的核心，因此，作为 JAVA 程序员也必须掌握。在学习数据库技术时，重点掌握 SQL 查询、表的设计范式、JDBC 技术、ORM 思想和 Mybatis 框架。其他关于数据库管理的知识不需要掌握太多。如果全天候学习，建议学习时间 9 天。

3. Servlet 和 JSP 技术

Servlet 和 JSP 是 JAVA 服务器开发的核心技术，也是必备技能。重点掌握如下知识：HTTP 协议、Tomcat 服务器、Servlet、Jsp、EL 表达式和标签库。如果全天候学习，建议学习时间 5 天。

4. 做一个简单的项目

学到现在，大家已经具备初步的项目开发能力。非常有必要通过一个项目，将所学的知识串联一遍，强化所学成果。同时，也通过项目开发的学习，真正进入 JAVA 企业级开发的行列。如果全天候学习，建议学习时间 5 天。

3. 框架阶段

框架做为企业使用的重要技术，其本质是“写更少的代码,做更多的事情”，从而大大减少代码量，让程序员更加关注业务的实现。自己写的少了，都是用别人写好的东西就要明白别人写了什么，学习框架重难点不仅仅是如果使用框架而是要明白别人到底写了什么。但是也要掌握好“度”，不要一味追求底层，要“择其重”，否则可能出现东西太多，永远也学习不完的情况。

1. MyBatis 框架

学习 MyBatis 框架要知道它是一个持久层框架。它的主要职责就是完成数据库操作，因此也要求我们有一定的 SQL 基础。MyBatis 灵活就体现在所有 SQL 都是程序员编写，想要什么样的结果就写什么样的 SQL 命令，剩下大部分的事情都交给 MyBatis 完成了。在我们这次课程设计上，我们把 Mybatis 放到了学完 JDBC 之后，这样既可以巩固 JDBC 和数据库知识，也能在即将要做的项目中开始使用 Mybatis，这样更贴合企业实战。

如果全天候学习，建议学习时间 3 天。

2. Spring 框架

学习 Spring 框架的目标很明确，就是学习好 IOC/DI、AOP、声明式事务。单独使用 Spring 很简单，重点是学习了核心功能后如何把这些功能运行。目前大部分技术都需要被 Spring 整合，所以 Spring 是很重要的框架。

Spring 框架不是专注于某一层，哪里都有 Spring 的影子。所以基础语法学习好后要多做几个项目去，才能真正的感受到 Spring 的魅力。

如果全天候学习，建议学习时间 3 天。

3. SpringMVC 框架

SpringMVC 做为 MVC 框架，其关注点就是控制器的编写。如何接收参

数, 如何传递内容到页面, 还有大量注解这些都是核心。学习时把握好主脉, 要从运行原理入手, 清楚的知道里面的核心组件, 学习起来就事半功倍了。

如果全天候学习, 建议学习时间 3 天

4. Shiro 安全框架

Shiro 应用场景都是在权限、安全上, 应用场景很明确, 所以就导致 Shiro 里面有着大量专业术语。学习 Shiro 从架构图出发, 清晰的记忆所有专业术语。我们分两大块学习 shiro, 分别是: 认证和授权。

如果全天候学习, 建议学习时间 2 天。

4. 高级技术阶段

高级技术阶段是随着电商火热后被大量使用的, 这些技术大部分都是为了解决”高负载”, ”高并发”等使用。

1. Linux

Linux 是操作系统, 学习 Linux 就是为了让自己的项目发布到 Linux 中, 所以学习 Linux 的主要要求是如何把项目运行的环境都配置出来。作为 JAVA 程序员不要求掌握特别深的 Linux 知识, 掌握基本操作即可。本课程就是针对 JAVA 程序员进行设计, 快速入门, 简单直接。

如果全天候学习, 建议学习时间 1 天。

2. Maven

Maven 是项目构建工具, 必须先要弄清楚分布式的概念。从学习了 Maven 后每个项目可能不是一个 Project 而是很多个, 所以重难点是项目和项目之间的关系以及 Maven 中 POM 的配置。

如果全天候学习, 建议学习时间 1 天

3. Dubbo

使用 Dubbo 先要弄清楚 RPC 概念。从架构图入手, 清晰明白 Dubbo 的架构和运行过程, 要多花一定时间在 Dubbo 的架构图上。另外, Dubbo 使用过程中要尽量使用配置文件配置而不要使用注解。

如果全天候学习, 建议学习时间 1 天

4. VSFTPD

VSFTPD 可以搭建图片服务器, 先要知道 FTP 协议, 然后弄清楚为什么使用 VSFTPD 以及熟练使用 FtpClient。

如果全天候学习, 建议学习时间 1 天

5. Nginx

先要弄清楚反向代理的概念, 代理本机目录和代理其他主机。熟练安装 Nginx 和配置 Nginx 是学习 Nginx 的重点。

如果全天候学习, 建议学习时间 1 天

6. Redis

NoSQL 数据库在企业中使用越来越多, 主要的优点就是效率高。每个 NoSQL 数据库有着自己独有的操作 API, 所以在能够熟练安装 Redis 基础上, 更要熟练使用 Jedis 操作 Redis。

如果全天候学习, 建议学习时间 1 天

7. Solr

Solr 搜索, 先要能够搭建 SolrCloud。学会使用 SolrJ 之后, 还需要明白”索引”, 搜索算法, Solr 实现原理才能掌握好 Solr。

如果全天候学习，建议学习时间 1 天

8. ActiveMQ

基于队列实现，明白队列概念后，把 ActiveMQ 安装上可以很快的上手。

如果全天候学习，建议学习时间 1 天

9. HttpClient

基于 HTTP 协议使用 Java 代码完成请求/响应的 API,所以请求及响应概念清楚，HttpClient 可以很快的上手。

如果全天候学习，建议学习时间 1 天

10. JsonP

浏览器异步请求静态资源，这个可以通过浏览器的调试工具清晰看到。明白这点 JsonP 也是可以很快的学习明白的。

如果全天候学习，建议学习时间 1 天

11. MyCat

为了保留关系型数据的优点，MyCat 作为一个解决 MySQL 不能分库分表的解决方案。学习 MyCat 先从配置 MySQL 主从备份开始，然后熟练配置 MyCat 配置文件。

如果全天候学习，建议学习时间 1 天

5. 设计模式和框架底层深化阶段

设计模式是解决特定问题的较优代码编写方案.在程序开发中时时刻刻都有着设计模式。学习设计模式必须要明白这么写代码是为了解决什么事情的。

可以先把所有设计模式都学习完，对于 GOF23 设计模式有一个基本的认识。然后在学习成熟框架的底层代码，这样技能将框架学习更深入也能让自己的底功更扎实。

学习设计模式可以拆分成多个小块学习，不建议一次性占用大量时间。

框架学习分为两种：一种是使用一种是明白实现原理。如果学习框架只是为了知道如果搭建框架的环境，如果使用框架，框架的学习还是比较容易的。但是如果想要非常清楚的明白每个功能是如何实现的就必须清楚的知道框架底层。

学习框架底层可以帮助我们在出现异常时快速的想到是什么原因出现这个错误。因为我们知道自己写的代码，也知道别人写了什么，出现了错误肯定会更好的解决。

MyBatis 框架底层可以从 SqlSessionFactory 入手，一个类一个方法的往下跟踪，一直跟踪到 SqlSession 关闭。

Spring 的底层源码要从 BeanFactory 和 ApplicationContext 入手，知道 Spring 容器作用。

SpringMVC 底层源码可以从 DispatcherServlet 入手，观察所有核心组件的运行过程。

Struts2 的底层代码要从 struts-default.xml 入手，里面配置了很多内置的内容。

Hibernate 的底层代码查看思路和 MyBatis 类似，从 Configuration 开始。

每个框架可以自己绘制一份流程图，这样更利于自己把整个框架的运行原理弄清楚

框架底层内容可以在边学习框架边学习。

【注】框架底层的学习适合学有余力的同学。如果掌握框架的使用都比较吃力，可以先放一下。熟练使用框架以后再深入底层。

6. 综合项目

技术技能学习完毕后，必须通过一个综合的项目将所学知识串一遍。既是复习，也是升华。我们采用“电商项目”作为综合项目。电商项目可以很简单，也可以很复杂。有的所谓电商项目就是一个商品管理和购物车，但是一个京东商城也是电商项目。所以，电商项目是一个非常好的可以串联知识点的项目，以及解决高并发问题好的场景。我们这里设计的电商项目，既包含了项目本身的各个模块实现逻辑，也包含了高并发场景如何处理的技术。吃透这个项目完全可以胜任绝大部分的开发工作。

7. 面试和求职

技术学习完毕了，我们也必须掌握一些基本的面试知识，让自己在找工作和求职过程中最少做到“不吃亏”。如果这部分知识掌握到位，可以起到“画龙点睛”的作用，让自己的临门一脚非常漂亮。为自己的职场发展开好头。