

技术学习

sully

2025 年 2 月 19 日

目录

目录	1
1 引言	2
2 Java 代码示例	2
3 图片插入示例	2
4 引言	3
5 插入 C++ 代码	4
6 简历样本与我的简历	5
7 冒泡、插入、两数和	7
8 尼恩架构 2025	9
9 nien	10
10 账号	11

1 引言

这里是文章的引言部分。

2 Java 代码示例

以下是一个嵌入的 Java 代码示例：

```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello, world!");  
    }  
}
```

Listing 1: HelloWorld.java

3 图片插入示例

下面是一个插入图片的示例：



这是 Q-book L^AT_EX 书籍模板，当前版本为 v2.01。

这份模板主要基于上海交大的学位论文模板¹修改得到，结合少量个人审美喜好，重新定制了定义、定理等环境。由于这个模板本是用于一项书籍翻译计划，因此其中的一些环境，例如“观察”、“规则”、“关键点”等，读者可能用不到，可以根据自己的需求适当修改。

你也可以通过邮箱 jey74165@163.com 给我发邮件反映遇到的问题。不过作者水平有限，或许有些问题也无法解答，还请见谅。

1.1 文档说明

1.1.1 准备工作

要想灵活使用、魔改这个模板来撰写自己的书籍，需要对 TeX 系统有一定的了解，也需要掌握基本的 TeX 技能。

- ▶ TeX 系统：所使用的 TeX 系统要支持 Xe^{La}T_EX 引擎，且带有 ctex 2.x 宏包。一般来说，只要安装了的完整 TeXLive 或 MacTeX 发行版就不会出现问题。
- ▶ TeX 技能：尽管提供了对模板的必要说明，但这并不是一份 “L^AT_EX 入门文档”。用户应当有一定的 L^AT_EX 使用经验。

¹<https://github.com/sjtuug/SJTUThesis>

图 1: 示例图片

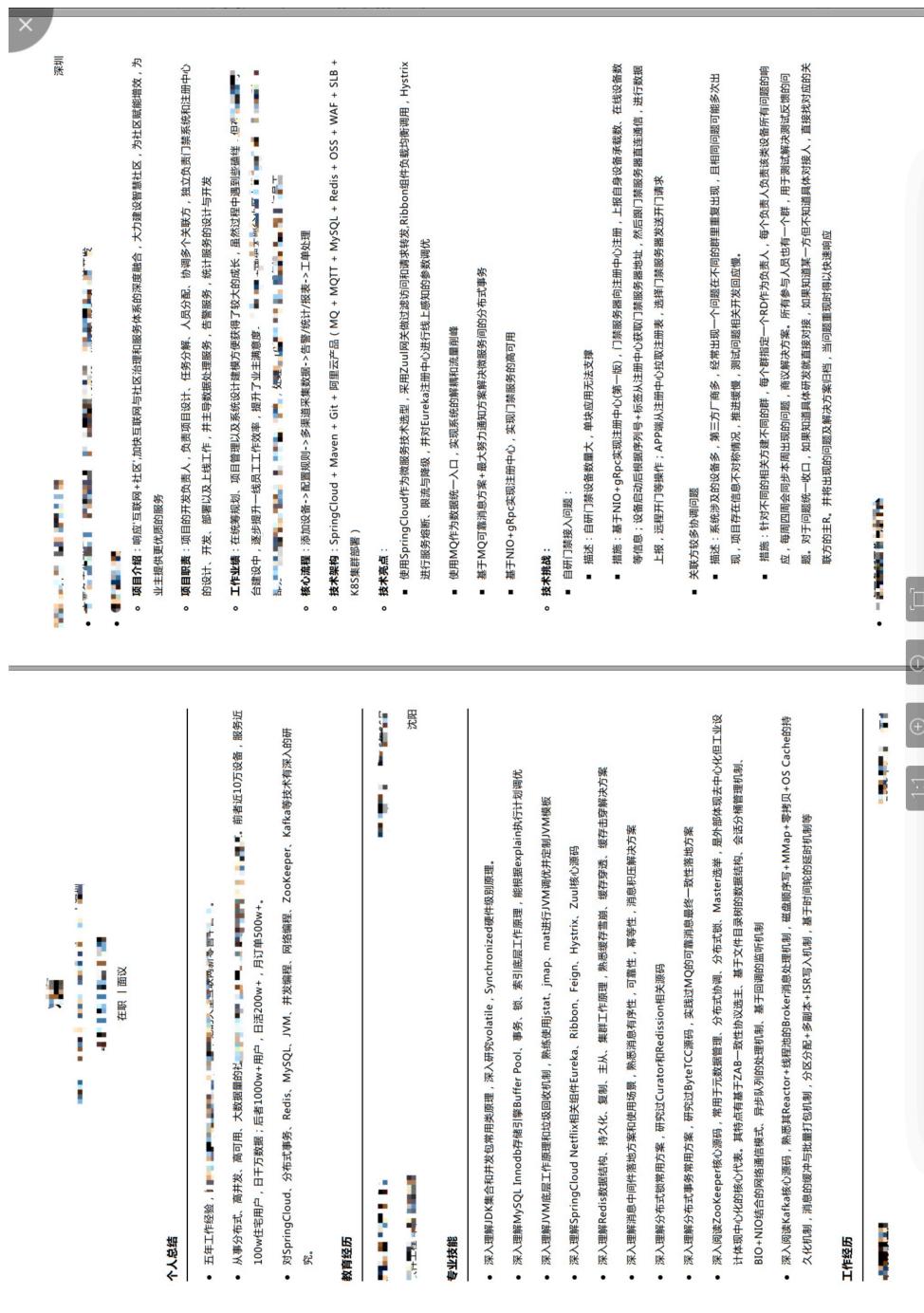
4 引言

这里是文章的引言部分。

5 插入 C++ 代码

本文展示了如何嵌入 Java 代码和图片，同时提供了目录、章节标题等结构。

6 简历样本与我的简历



于明

南京市, 鼓楼区和会街 131 号, 江苏, 中国 210000
(+86) 13912991965 | rateyu@gmail.com



个人总结

- 丰富经验: 10 年以上软件开发经验, 专注高并发、分布式系统, 精通 JVM 调优及性能优化, 擅长解决复杂系统瓶颈。
- 大数据平台: 主导 40GB 每日数据流处理, 优化 Kafka 和 Redis 数据传输, 确保系统高效稳定运行。
- 高并发系统设计: 设计并优化 RPC 调用组件, 支持百万级客户端接入, 特别在秒杀场景中, 提升 10 万并发连接性能, 保障高流量稳定性。
- 技术驱动团队: 熟练应用领域驱动设计 (DDD), 注重代码规范与单元测试, 推动团队协作与技术创新。

专业技能

- 深入理解 JVM 底层工作原理和垃圾回收机制, 熟练使用 jstat、jmap、arthas 等工具进行问题定位和性能调优。
- 负责整套系统的需求分析、信息模型设计、功能设计、数据库设计和功能测试等。
- 独立负责运动员管理与检录的开发工作, 基于 RFID 实现运动员自动化注册和检录。
- 承担 2016 年湖南省武术比赛、2017 年湖北省青少年武术锦标赛的竞赛管理。

工作经历

高级 java 软件工程师, 江苏开博科技有限公司, 南京, 江苏

05/2021 - 至今

- 主导设计并参与开发一套武术体育竞赛管理信息系统。
- 负责整套系统的需求分析、信息模型设计、功能设计、数据库设计和功能测试等。
- 独立负责运动员管理与检录的开发工作, 基于 RFID 实现运动员自动化注册和检录。
- 承担 2016 年湖南省武术比赛、2017 年湖北省青少年武术锦标赛的竞赛管理。

主任秘书, 襄阳百洋房地产开发有限公司, 襄阳, 湖北

05/2015 - 08/2015

- 协助公司开发部主任, 进行与房产开发有关地方行政单位的联络、日程和会议等工作。
- 学习主任与地方单位负责人的沟通技巧, 提炼语言能力。

教育背景

南京大学

南京, 江苏

计算机科学与技术 (本科)

09/2002 - 06/2004

徐州师范大学

徐州, 江苏

机电设备维修与管理 (大专)

09/1999 - 06/2002

7 冒泡、插入、两数和

```

class Solution {
    public int[] twoSum(int[] nums, int target) {
        int[] aaa = { 7, 5, 3, 2, 8, 0, 1 };

        System.out.println("冒泡排序");
        for (int i = 0; i < aaa.length; i++) {
            for (int j = 0; j < aaa.length - i - 1; j++) {
                int tmp = aaa[j + 1];
                if (aaa[j] > aaa[j + 1]) {
                    aaa[j + 1] = aaa[j];
                    aaa[j] = tmp;
                }
            }
        }

        for (int i = 0; i < aaa.length; i++) {
            System.out.print(aaa[i] + ",");
        }
        System.out.println("");
        System.out.println("插入排序");
        int[] bbb = { 7, 5, 3, 2, 8, 0, 1 };
        for (int i = 1; i < bbb.length; i++) {
            int tmpV = bbb[i];
            int j = i - 1;
            /* 方法一，将比 key 大的元素向右移动 */
            /*
            while (j >= 0 && bbb[j] > tmpV) {
                bbb[j + 1] = bbb[j];
                j = j - 1;
            }
            bbb[j + 1] = tmpV;
            */
            //方法二
            for(;j>=0;j--) {
                if(bbb[j]>tmpV){
                    bbb[j+1] = bbb[j];
                } else {
                    break;
                }
            }
            bbb[j+1] = tmpV;
        }

        for (int i = 0; i < bbb.length; i++) {
            System.out.print(bbb[i] + ",");
        }
        Map <Integer, Integer> myMap = new HashMap<>();
        for(int i=0;i<nums.length;i++){
            if(myMap.containsKey(target-nums[i])){
                return new int []{myMap.get(target-nums[i]),i};
            }
            myMap.put(nums[i],i);
        }
        return new int [0];
    }
}

def twoSum(self, nums: List[int], target: int) -> List[int]:
    n = len(nums)
    for i in range(n - 1):
        for j in range(i + 1, n):
            if nums[i] + nums[j] == target:
                return [i, j]
    return [] # No solution found

class Solution:
    def twoSum(self, nums: List[int], target: int) -> List[int]:
        numMap = {}
        n = len(nums)

        for i in range(n):
            complement = target - nums[i]
            if complement in numMap:
                return [numMap[complement], i]
            numMap[nums[i]] = i

        return [] # No solution found
go

func twoSum(nums []int, target int) []int {
    // Hash table to store number->index mapping
    ht := make(map[int]int)

    // Iterate through the array
    for i, num := range nums {
        // Check if complement exists in hash table
        if j, exists := ht[target-num]; exists {
            // If found, return indices of both numbers
            return []int{j, i}
        }

        // Store current number and its index in hash table
        ht[num] = i
    }

    // Return empty slice if no solution found
    return []int{}
}

javascript

var twoSum = function(nums, target) {
    let map = new Map();

    for (let i=0; i<nums.length; i++){
        if(map.has(target-nums[i])){
            return [map.get(target-nums[i]), i];
        }else{
            map.set(nums[i], i)
        }
    }
};


```

```
//c++
class Solution {
public:
    vector<int> twoSum(vector<int>& nums, int target
    ) {
        unordered_map<int, int> numMap;
        int n = nums.size();
        for (int i = 0; i < n; i++) {
            int complement = target - nums[i];
            if (numMap.count(complement)) {
                return {numMap[complement], i};
            }
            numMap[nums[i]] = i;
        }
    }
}

return {};// No solution found
};

N.283 移动零问题
public void moveZeroes(int[] nums) {
    int j = 0;
    for (int i = 0; i < nums.length; i++) {
        if (nums[i] != 0) {
            nums[j] = nums[i];
            if (i != j){
                nums[i] = 0;
            }
            j++;
        }
    }
}
}
```

8 尼恩架构 2025

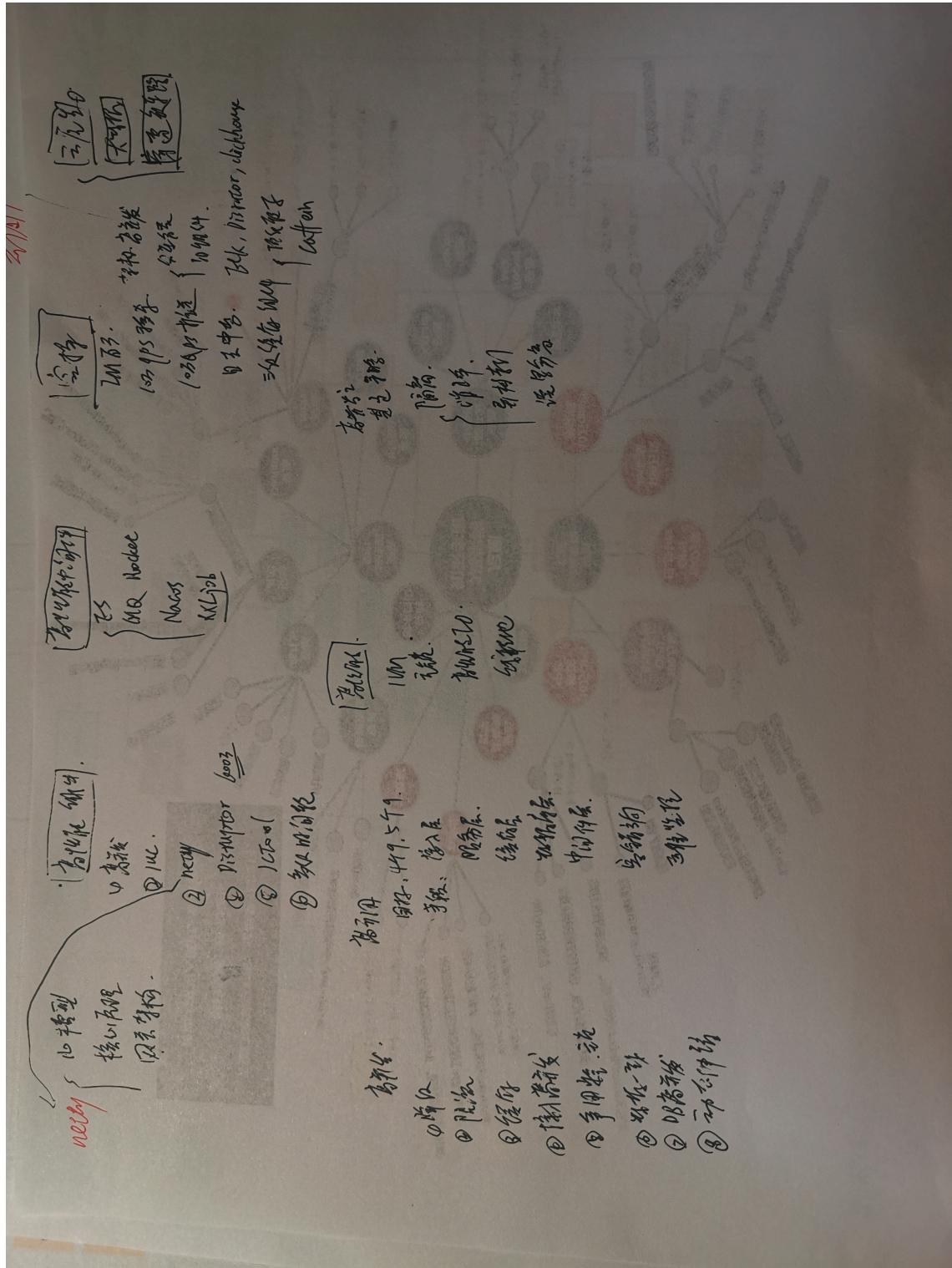


图 3: nien 架构

9 nien

- 第 1 章: 9 史上最强线程池学习盛宴
- 第 2 章: 6 Netty 核心原理与底层知识学习盛宴
- 第 3 章: 7 NettyByteBuf 学习盛宴
- 第 4 章: 3 百万级 IM 实战——CrazyIM 会话管理
- 第 5 章: 1 Java 必备——Netty 高并发灵魂编程
- 第 6 章: 11 九阳真经: 彻底揭秘 NIO、Selector 底层原理
- 第 7 章: 2 底层解读: 解密核心难题, 秒杀外国权威
- 第 8 章: 27 Netty 大实战: 从 0 到 1 开始亿级流量 CrazyIM 开发
- 第 10 章: 11-10W QPS 真刀实操以及基于 ZK+Netty 手写分布式测试工具
- 第 11 章. 4-5 分钟把简历变得闪闪发光, 人见人爱, 回头率 100
- 第 12 章.22-吊打面试官: 彻底明白分布式事务原理, 以及 seata 的 AT、TCC 原理与实操
- 第 13 章.21-史上最强: 从 0 开始 Netty IM 实战, 40 岁老架构师细致解读, 实战之中处处透着原理和精髓
- 第 14 章.40-横扫全网, elasticsearch 底层原理与高可用架构实操, 40 岁老架构师细致解读, 处处透着原理和精髓
- 第 15 章: 5-《springcloud nginx 高并发核心编程》配套视频
- 第 16 章: 73-葵花宝典 (高性能秘籍)
- 第 17 章: 45-横扫全网系列: 工业级 rocketmq 高可用底层原理和实操
- 第 18 章: 80-架构师超级内功篇: rocketmq 源码学习以及 3 高架构模式解读
- 第 19.1 章 vep: 61-10Wqps 推送中台从架构到实操
- 第 19 章: 35-10Wqps 推送中台实操 (下)
- 第 20 章: 156-横扫全网系列: 彻底穿透 Netty 原理与实操
- 第 21 章: 19-6 个面试必备 Redis cluster 的核心实操
- 第 22 章: 65-彻底穿透 Netty 架构与源码, 一卷到底, 卷透底层
- 第 23 章 sz: 70-100W 级别 QPS 日志平台实操
- 第 24 章 sz: 46-资深架构必备, 彻底穿透 Skywalking 链路跟踪源码、JavaAgent 探针技术
- 第 25 章 sz: 42-彻底穿透 400W 级 OPS 缓存之王 Caffeine 底层架构、核心源码
- 第 26 章 sz: 52-超高并发场景 100Wqps 三级缓存组件原理和实操
- 第 27 章 sz: 16-全链路异步超底层原理和实操: 手写 hystrix 熔断 + webflux + Lettuce + Dubbo
- 第 28 章 sz: 30-穿透云原生 K8S+Jenkins+SpringCloud 底层原理和实操
- 第 29 章 sz: 61-Golang 学习圣经, 高并发 Go+Java 混合微服务架构与实操
- 第 30 章 sz: 6-Java 调优圣经——顶尖高手的调优绝技
- 第 31 章 sz: 17-超高并发 1000W 级 golang 推送中台架构与实操
- 第 32 章 sz: 19-超高并发、超高可用 1000W 级 ID 组件架构与实操
- 第 33 章 sz: 41-10Wqps 高并发 Netty 网关架构与实操 (1)
- 第 34 章, 30-DDD 学习圣经, 从未开始 DDD 架构和落地
- 第 35 章: 46-中间件塔尖实战—穿透 Netty 对象池、内存池
- 第 36 章: 22-进大厂、做架构顶奢核心面试题 (第一期)
- 第 37 章: 25-百亿级数据存储架构: MYSQL 双写 + HABSE + Flink + ES 综合大实操
- 第 38 章: 12-ElasticSearch 学习圣经: 从 0 到 1, 精通 ElasticSearch 工业级实操
- 第 39 章: 12-XXL-JOB 学习圣经: 从 0 到 1, 精通 xxl-job 工业级实操
- 第 40 章: 11-HBASE 学习圣经: 从 0 到 1, 精通 HBASE 的底层原理和实操
- 第 41 章: 7-Flink 学习圣经: 从 0 到 1, 精通 Flink 的底层原理和实操
- 第 42-46 章: 尼恩 Java 硬核架构班 4 微服务底座-讲义与源码
- 42-token 25
- 43-dubbo 20
- 44-nacos 17
- 45-sentinel 27
- 46-elk, felk 22

10 账号

192.168.1.7

8.2 账户 root/root123

用户:seeyon

密码统一: Jsyh@123

mysqldump -uroot -p seeyon > seeyon.sql

mysql -uroot -p seeyon82b < seeyon.sql

8.0 账户 mysql

root/root123

ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'root123';

system/system12

group-admin/groupadmin12

s1agent/s1agent12

seeyon/user123456

admin1/user123456

01011001/user123456

new-normal-user/user123456