

GitHub 入门实践

课程目标：

1. 代码审查不到位，审查效率低下
2. 只有编程者本人能看懂的代码直接被部署到正式环境中
3. 因低级代码错误导致 BUG 频繁出现
4. 没有机会和其他人互相交流代码，共享知识

没有一个简单高效、能在一天之内添加多个功能的开发流程

GitHub 为我们提供了解决这些问题的机会和功能。

本课程旨在指导读者使用 GitHub 进行高效开发。

1. 内容全面，系统讲解 GitHub 的功能和实用技巧
2. 图文直观，一步步演示 GitHub 的使用方法
3. 实战导向，专门搭建实践仓库，邀请读者进行 Pull Request 并共同维护
4. 包含 GitHub 的开发流程和辅助工具，适合需要引入 GitHub 的公司或团队直接使用

适用学员：

- 程序员
- 系统工程师
- 开发工程师

课程设计：

第一章 Git 概述

- 1 诞生背景
- 2 什么是版本管理
 - ✓ 集中型与分散型
 - ✓ 集中型
 - ✓ 分散型
 - ✓ 集中型与分散型哪个更好
- 3 安装

- ✓ Mac 与 Linux
- ✓ Windows
- ✓ 组件的选择
- ✓ 设置环境变量
- ✓ 换行符的处理
- ✓ Git Bash

4 初始设置

- ✓ 设置姓名和邮箱地址
- ✓ 提高命令输出的可读性

第二章 Git 命令

1. 设置用户签名
2. 初始化本地库
3. 查看本地库状态
4. 添加暂存区
5. 提交本地库
6. 修改文件
7. 版本穿梭

第三章 GitHub

1. 创建远程库
2. 推荐本地库到远程库
3. 拉取远程库到本地库
4. 克隆远程库到本地
5. 团队内协作
6. 跨团队协作
7. SSH 免密登录

第四章 Git-IDEA 集成

1. Git 环境准备
2. 初始化
3. 切换版本
4. 创建分支
5. 合并分支
6. GitHub 设置账号
7. 分享项目到 GitHub
8. 推送代码到远程库
9. 拉取远程库代码合并本地库
10. 克隆代码到本地

第四章 Git-码云

1. 账号创建登录
2. 初始化
3. Gitee 集成码云
4. 导入 GitHub 项目

第五章 GitLab

1. 简介
2. 安装和环境准备
3. 初始化
4. 启动服务
5. 登录 GitLab 创建
6. GitLab 和 IDEA 集成