

- 1 介绍
 - 1.1 课程介绍
 - 1.2 自我介绍
- 2 项目开发流程
- 3 Git&Gitee&GitHub
- 4 Git常用命令
 - 4.1 基础命令
 - 4.2 分支命令
 - 4.3 远程仓库命令
 - 4.4 快捷操作命令
- 5 团队协作
 - 5.1 团队内协作
 - 5.2 跨团队协作
- 6 注意
- 6 工作中正确使用git

1 介绍

1.1 课程介绍

- 总时长：15个工作日
- 课程内容：SSM框架
 - Git、Maven
 - Spring框架、Spring MVC框架、MyBatis框架
 - Knife4j、Lombok、JsonResult、拦截器、AOP...

1.2 自我介绍

- 姓名：王伟超
- 邮箱：wangweichao@tedu.cn
- 微信：13603263409

2 项目开发流程

- 需求文档[产品经理]
- 原型图设计[设计]
- 数据库设计[数据架构师 - ER图]
- 前后端并行开发
 - 前端：Web前端工程师进行开发
 - 后端：Java开发工程师
 - 用户模块：注册、登录、第三方登录、忘记密码、修改密码... ..
 - 商品模块：首页展示、详情页、列表页... ..
 - 购物车模块
 - 订单模块
 - 支付模块
- 前后端联调

- 测试[测试工程师]
- 部署上线
- 运行维护,更新迭代

3 Git&Gitee&GitHub

Git是版本控制工具，Gitee和GitHub则是基于Git的代码托管仓库。

- Git
Git：免费的、开源的分布式版本控制系统。
- Gitee
开源中国推出的基于Git的代码托管服务[国内,稳定速度快]。
- GitHub
开源的代码托管平台[外网,不稳定]

4 Git常用命令

注意：首次安装使用git时，需要设置用户签名。

```
git config --user.name 用户名[自定义]
git config --user.email 用户邮箱[自定义]
```

4.1 基础命令

- 初始化本地仓库： `git init`
- 添加到暂存区： `git add 文件名`
- 提交到本地仓库： `git commit -m "提示信息" 文件名`
只有提交到本地仓库后，才会生成唯一的历史版本
- 查看本地仓库状态： `git status`
未添加到暂存区的文件，颜色为：**红色**
添加到暂存区但未提交到本地仓库的文件，颜色为：**绿色**
- 查看历史版本： `git reflog`
- 版本穿梭： `git reset --hard 历史版本号`

4.2 分支命令

- 创建分支： `git branch 分支名`
- 查看分支： `git branch -v`
- 切换分支： `git checkout 分支名`
- 合并分支： `git merge 分支名`
注意：在合并之前首先要切换到合并到的分支，然后再执行合并分支的命令
- 合并冲突问题解决方案
 - 第1步：手动编辑有冲突的文件；
 - 第2步：添加到暂存区，并提交到本地仓库[提交到本地仓库时commit命令不能加文件名]。

4.3 远程仓库命令

- 本地推送到远程: `git push 远程地址 分支`
- 克隆远程仓库: `git clone 远程地址`
- 本地拉取远程: `git pull 远程地址 分支`

操作本地代码前，一定要先拉取pull远程仓库的代码。

4.4 快捷操作命令

- 清屏: `Ctrl + l`
- 自动补全: `命令关键字 + Tab键` **能使用自动补全的坚决杜绝手写!**
- 字体放大缩小: `Ctrl + 鼠标滚轮`
- 粘贴: `Shift + Insert`
- 终止当前命令执行: `Ctrl + c`

5 团队协作

5.1 团队内协作

领导把开发人员添加为公司远程仓库的成员即可。

5.2 跨团队协作

- 第1步: 孙悟空把仓库地址发送给观音菩萨;
- 第2步: 观音菩萨首先 Fork 到自己的仓库;
- 第3步: 观音菩萨解决问题, 并提交到自己的仓库;
- 第4步: 观音菩萨创建 Pull Requests 请求, 发送给孙悟空;
- 第5步: 孙悟空在 Pull Requests 中, 进行审核、测试、代码合并;
- 第6步: 给观音菩萨酬劳。

6 注意

push 是将本地库代码推送到远程库, 如果本地库代码跟远程库代码版本不一致, push 的操作是会被拒绝的。也就是说, 要想 push 成功, 一定要保证本地库的版本要比远程库的版本高! 因此一个成熟的程序员在动手改本地代码之前, 一定会先检查下远程库跟本地代码的区别! 如果本地的代码版本已经落后, 切记要先 pull 拉取一下远程库的代码, 将本地代码更新到最新以后, 然后再修改, 提交, 推送!

6 工作中正确使用git

- 第1: 写代码之前先把项目从远程仓库pull下来, 先把master主分支上的代码与远程仓库同步到最新版本;
- 第2: 在本地创建并切换到工作分支, 创建完成后会自动切换到该分支, 此时master/work分支内容是相同;
- 第3: 编写代码;
- 第4: 编码完成后, add commit到work分支;
- 第5: 将主分支从远程仓库pull最新的版本下来;
- 第6: 将工作分支合并到主分支, 如果有合并冲突, 则需要手动解决冲突;
- 第7: 此时所有在work分支上做的修改就会合并到主分支上, 最后就可以push到远程仓库上了。

