

1. 练习 - 学生管理

• 1.1 创建版本库，在版本库中提交一个文件：Student.java

1. 创建student目录，切换到student目录

```
etoak@Damon MINGW64 ~/Desktop
$ mkdir student ① 创建student目录
```

```
etoak@Damon MINGW64 ~/Desktop
$ cd student ② 切换到student目录
```

```
etoak@Damon MINGW64 ~/Desktop/student
$
```

2. 在student目录中初始化本地版本库

```
etoak@Damon MINGW64 ~/Desktop/student
$ git init ③ 初始化本地版本库
Initialized empty Git repository in C:/Users/etoa
it/
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$
```

3. 配置用户名和邮箱

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git config --local user.name "zs"
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git config --local user.email "zs@et.com"
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git config --local -l
```

```
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true
user.name=zs
user.email=zs@et.com
```

配置级别	配置文件	配置命令	优先级
本地版本库	.git/config	git config --local	最高
全局（用户级）	~/.gitconfig	git config --global	其次
系统级	Git安装目录/etc/git config	git config --system	最低

查看配置内容

```
git config [--local] --list
git config [--global] -l
```

修改: git config [--local] --replace-all user.name "lisi"

删除: git config [--local] --unset user.name

4. 创建Student.java，提交到本地版本库

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ touch Student.java ① 创建Student.java
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git add Student.java ② 添加到暂存区
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git commit -m "添加Student.java" ③ 提交到本地版本库
[master (root-commit) a0e10e2] 添加Student.java
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Student.java
```

- 1.2 开发添加学生功能

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ echo "// 添加学生" >> Student.java ①
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git add . ②
warning: in the working copy of 'Student.java'
CRLF the next time Git touches it
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git commit -m "完成添加功能" ③
[master 2296aa4] 完成添加功能
1 file changed, 1 insertion(+)
```

- 1.3 开发查询功能 - (向Student.java中写一行注释“//查询学生”) , 提交到本地版本库

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ echo "// 查询学生" >> Student.java ①
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git add . ②
warning: in the working copy of 'Student.java'
LF the next time Git touches it
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git commit -m "完成查询功能" ③
[master ef6bf97] 完成查询功能
1 file changed, 1 insertion(+)
```

- 1.4 开发聊天功能 - (向Student.java中写一行注释“// 聊天”) , 提交到本地版本库

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ echo "// 聊天" >> Student.java

etoak@Damon MINGW64 ~/Desktop/student (master)
$ git add .
warning: in the working copy of 'Student.java'
LF the next time Git touches it

etoak@Damon MINGW64 ~/Desktop/student (master)
$ git commit -m "完成聊天功能"
[master 61bc85a] 完成聊天功能
1 file changed, 1 insertion(+)
```

2. 重置版本

- 前提：要重置的版本不应该推送到远程版本库

1. 重置版本库、暂存区和工作区： `git reset --hard 版本记录`

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git log --oneline
61bc85a (HEAD -> master) 完成聊天功能
ef6bf97 完成查询功能
2296aa4 完成添加功能
a0e10e2 添加Student.java
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git reset --hard ef6bf97 重置三大区域
HEAD is now at ef6bf97 完成查询功能
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git log --oneline
ef6bf97 (HEAD -> master) 完成查询功能
2296aa4 完成添加功能
a0e10e2 添加Student.java
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ cat Student.java
// 添加学生
// 查询学生
```

2. 重置版本库，不重置暂存区和工作区： `git reset --soft 版本记录`
3. 重置版本库和暂存区，不重置工作区： `git reset --mixed 版本记录`

3. 撤销暂存区修改

- 实际就是撤销刚执行的 `git add` 步骤

1. `git reset HEAD 文件...`
2. `git restore --staged 文件...` (git 2.20 +)

4. 撤销工作区修改

- 前提：要撤销的文件一定已经生成版本了

1. `git checkout -- 文件...`
2. `git restore 文件...`

5. 分支

• 5.1 创建dev分支，在dev分支上开发成绩查询(Score.java)，合并回master分支

1. 创建dev分支，切换到dev分支

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git branch dev ① 创建dev分支
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git branch -l 查看本地分支列表
dev
* master
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git checkout dev ② 切换到dev分支
Switched to branch 'dev'
```

2. 在dev分支上开发成绩查询功能，提交到本地版本库

```
etoak@Damon MINGW64 ~/Desktop/student (dev)
$ touch Score.java
```

```
etoak@Damon MINGW64 ~/Desktop/student (dev)
$ git add .
```

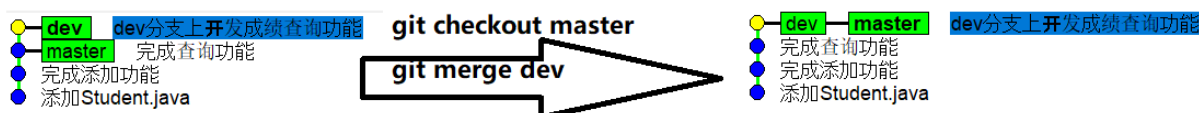
```
etoak@Damon MINGW64 ~/Desktop/student (dev)
$ git commit -m "dev分支上开发成绩查询功能"
[dev 9687bf9] dev分支上开发成绩查询功能
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Score.java
```

3. 切换到master分支，合并dev分支的版本

```
etoak@Damon MINGW64 ~/Desktop/student (dev)
$ git checkout master ① 切换到master分支
Switched to branch 'master'
```

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git merge dev ② master分支上合并dev分支的版本
Updating ef6bf97..
Fast-forward
 Score.java | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Score.java
```

4. 合并结果展示



5. 删除dev分支

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git branch -d dev 删除dev分支
Deleted branch dev (was 9687b19).

etoak@Damon MINGW64 ~/Desktop/student (master)
$ git branch -l 查看本地分支, dev分支已经被删除
* master

etoak@Damon MINGW64 ~/Desktop/student (master)
$
```

• 5.2 创建dev分支并切换到dev分支, 在dev和master上分别提交一个版本, 将dev分支的版本合并到master上

1. 创建dev分支, 并直接切换到dev分支, 在dev上开发课程查询 (Course.java), 提交到本地版本库

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git checkout -b dev ① 创建并切换到dev分支
Switched to a new branch 'dev'

etoak@Damon MINGW64 ~/Desktop/student (dev)
$ touch Course.java ② 开发课程查询功能

etoak@Damon MINGW64 ~/Desktop/student (dev)
$ git add . ③ 添加到暂存区

etoak@Damon MINGW64 ~/Desktop/student (dev)
$ git commit -m "dev分支开发课程查询" ④
[dev 3ffb738] dev分支开发课程查询
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Course.java
```

2. 切换到master分支, 在master上开发作业功能 (Homework.java), 提交到本地版本库

```
etoak@Damon MINGW64 ~/Desktop/student (dev)
$ git checkout master ①
Switched to branch 'master'

etoak@Damon MINGW64 ~/Desktop/student (master)
$ touch Homework.java ②

etoak@Damon MINGW64 ~/Desktop/student (master)
$ git add . ③

etoak@Damon MINGW64 ~/Desktop/student (master)
$ git commit -m "master开发作业功能" ④
[master ece21af] master开发作业功能
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Homework.java
```

3. 在master上合并dev分支的版本

```
git merge dev
```

4. 合并结果



• 5.3 分支命令

1. 创建分支: `git branch 分支名称`
2. 切换分支: `git checkout 分支名称`
3. 创建并直接切换分支: `git checkout -b 分支名称`
4. 查看本地分支列表: `git branch -l`
5. 查看所有分支列表: `git branch -va`
6. 合并分支: `git merge 分支名称`
7. 删除分支: `git branch -d 分支名称`

6. 在gitee.com上注册账号，并配置SSH KEY

• 6.1 配置SSH Key

1. 执行命令

```
ssh-keygen ed25519 xxx@xxx.com
```

2. 拷贝用户目录下的 `id_算法名.pub` 文件的公钥到Gitee

• 6.2 创建远程版本库

新建仓库

在其他网站已经有仓库了吗? [点击导入](#)

仓库名称 * ✓

stu

归属

x

路径 * ✓

/ stu

仓库地址: <https://gitee.com/xilelu/stu>

仓库介绍

0/100

用简短的语言来描述一下吧

- ☐ 开源 (所有人可见) ②
- ☒ 私有 (仅仓库成员可见)
- ☐ 企业内部开源 (仅企业成员可见) ②

- ☐ 初始化仓库 (设置语言、.gitignore、开源许可证)
- ☐ 设置模板 (添加 Readme、Issue、Pull Request 模板文件)
- ☐ 选择分支模型 (仓库创建后将根据所选模型创建分支)

创建

• 6.3. 为本地版本库关联刚创建远程版本库，并将本地版本库的版本推送到远程版本库

1. 关联远程版本库

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git remote add origin git@gitee.com:xilelu/stu.git

etoak@Damon MINGW64 ~/Desktop/student (master)
$ cat .git/config
[core]
    repositoryformatversion = 0
    filemode = false
    bare = false
    logallrefupdates = true
    symlinks = false
    ignorecase = true
[user]
    name = zs
    email = zs@et.com
[remote "origin"]
    url = git@gitee.com:xilelu/stu.git
    fetch = +refs/heads/*:refs/remotes/origin/*
```

2. 推送版本到远程版本库

```
git push origin master
```

7. 模拟两个用户协同开发

• 7.1 克隆一个远程版本库，并配置用户名为lisi和邮箱 lisi@etoak.com

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ cd .. 退出student仓库 不能在student目录中克隆

etoak@Damon MINGW64 ~/Desktop
$ git clone git@gitee.com:xilelu/stu.git
Cloning into 'stu'...
remote: Enumerating objects: 21, done.
remote: Counting objects: 100% (21/21), done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 21 (delta 5), reused 0 (delta 0)
Receiving objects: 100% (21/21), done.
Resolving deltas: 100% (5/5), done.

etoak@Damon MINGW64 ~/Desktop
$ 

etoak@Damon MINGW64 ~/Desktop
$ cd stu 切换到刚才克隆的仓库目录

etoak@Damon MINGW64 ~/Desktop/stu (master)
$ git config --local user.name "lisi"

etoak@Damon MINGW64 ~/Desktop/stu (master)
$ git config --local user.email "lisi@etoak.com"

etoak@Damon MINGW64 ~/Desktop/stu (master)
$ git config --local -l
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true
remote.origin.url=git@gitee.com:xilelu/stu.git
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
branch.master.remote=origin
branch.master.merge=refs/heads/master
user.name=lisi
user.email=lisi@etoak.com
```


• 7.2 lisi和zs分别开发一个新的功能，分别向远程版本库推送

1. lisi先在stu仓库中开发签到功能 (Sign.java)，推送到远程版本库

```
etoak@Damon MINGW64 ~/Desktop/stu (master)
$ touch Sign.java ①

etoak@Damon MINGW64 ~/Desktop/stu (master)
$ git add . ②

etoak@Damon MINGW64 ~/Desktop/stu (master)
$ git commit -m "lisi开发签到功能" ③
[master a529c74] lisi开发签到功能
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Sign.java

etoak@Damon MINGW64 ~/Desktop/stu (master)
$ git push origin master ④
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
```

2. zs在student仓库中开发签退功能 (Out.java)，推送到远程版本库，但是被远程版本库拒绝了

拒绝原因：本地没有远程版本库的所有版本

```
etoak@Damon MINGW64 ~/Desktop/stu (master)
$ cd ../student/ ① 切换到zs的student版本库

etoak@Damon MINGW64 ~/Desktop/student (master)
$ touch Out.java ②

etoak@Damon MINGW64 ~/Desktop/student (master)
$ git add . ③

etoak@Damon MINGW64 ~/Desktop/student (master)
$ git commit -m "zs开发签退功能" ④
[master 004f51f] zs开发签退功能
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Out.java

etoak@Damon MINGW64 ~/Desktop/student (master)
$ git push origin master ⑤
To gitee.com:xilelu/student.git
master
! [rejected]        master -> master
error: failed to push some refs to 'gitee.com:xilelu/student.git'
hint: Updates were rejected because the remote contains work that
hint: not have locally. This is usually caused by another repository
having made updates in the same time period.
```

3. 所以，zs应该先拉取远程版本库的版本，与本地的版本合并之后，再向远程版本库推送

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git pull origin master
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), fetch ratio 1.000
Retrieving objects: 100% (3/3), done.
Unpacking objects: 100% (3/3), done.
From gitee.com: xilelu/student
* branch            master       -> FETCH_HEAD
f766dcb..a529c74  master       -> origin/master
Merge made by the 'ort' strategy.
 sign.java | 0
1 file changed, 0 insertions(+), 0 deletions(-)

etoak@Damon MINGW64 ~/Desktop/student (master)
$ git push origin master
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
```

8. 版本冲突说明

- 多个用户同时修改了相同的文件

1. 修改相同文件的不同行：一般情况下，版本控制系统都可以自动合并
2. 修改相同文件的相同行：合并时出现冲突，需要手动解决冲突
3. 既修改了不同行，也修改了相同行：一般情况下不同行自动合并，相同行手动解决冲突

• 8.1 lisi和zs同时修改签到功能

1. lisi先更新最新版本，修改签到功能，推送到远程版本库

```
etoak@Damon MINGW64 ~/Desktop/stu (master)
$ git pull origin master
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 2), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (4/4), 449 bytes | 44.00 KiB/s, done.
etoak@Damon MINGW64 ~/Desktop/stu (master)
$ echo "//lisi" >> Sign.java
etoak@Damon MINGW64 ~/Desktop/stu (master)
$ git commit -am "lisi修改签到功能"
warning: in the working copy of 'Sign.java', LF will be replaced by CRLF
[master 8f27870] lisi修改签到功能
1 file changed, 1 insertion(+)
etoak@Damon MINGW64 ~/Desktop/stu (master)
$ git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
```

2. zs在没有更新版本的情况下，也修改签到功能，向远程版本库推送时被拒绝

```
etoak@Damon MINGW64 ~/Desktop/stu (master)
$ cd ../student/ ①
etoak@Damon MINGW64 ~/Desktop/student (master)
$ echo "//zs" >> Sign.java ②
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git commit -am "zs修改签到功能" ③
warning: in the working copy of 'Sign.java', LF will be replaced by CRLF
[master e70a1d7] zs修改签到功能
1 file changed, 1 insertion(+)
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git push origin master ④
To gitee.com:xilelu/stu.git
! [rejected]        master -> master (fetch first)
```

3. 所以，zs先拉取最新的版本，此时，出现版本冲突了

```
etoak@Damon MINGW64 ~/Desktop/student (master)
$ git pull origin master
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 260 bytes | 37.00 KiB/s, done.
From gitee.com:xilelu/stu
* branch                master      -> FETCH_HEAD
   cd6db65..8f27870      master      -> origin/master
Auto-merging Sign.java
CONFLICT (content): Merge conflict in Sign.java
Automatic merge failed; fix conflicts and then commit the result.
```

4. zs要先解决冲突
5. zs使用 `git add` 命令 标记解决了冲突（告诉Git冲突解决了）

```
etoak@Damon MINGW64 ~/Desktop/student (master|MERGING)
$ git add .

etoak@Damon MINGW64 ~/Desktop/student (master|MERGING)
$ git status
On branch master
All conflicts fixed but you are still merging.
(use "git commit" to conclude merge)
```

Changes to be committed:
modified: Sign.java

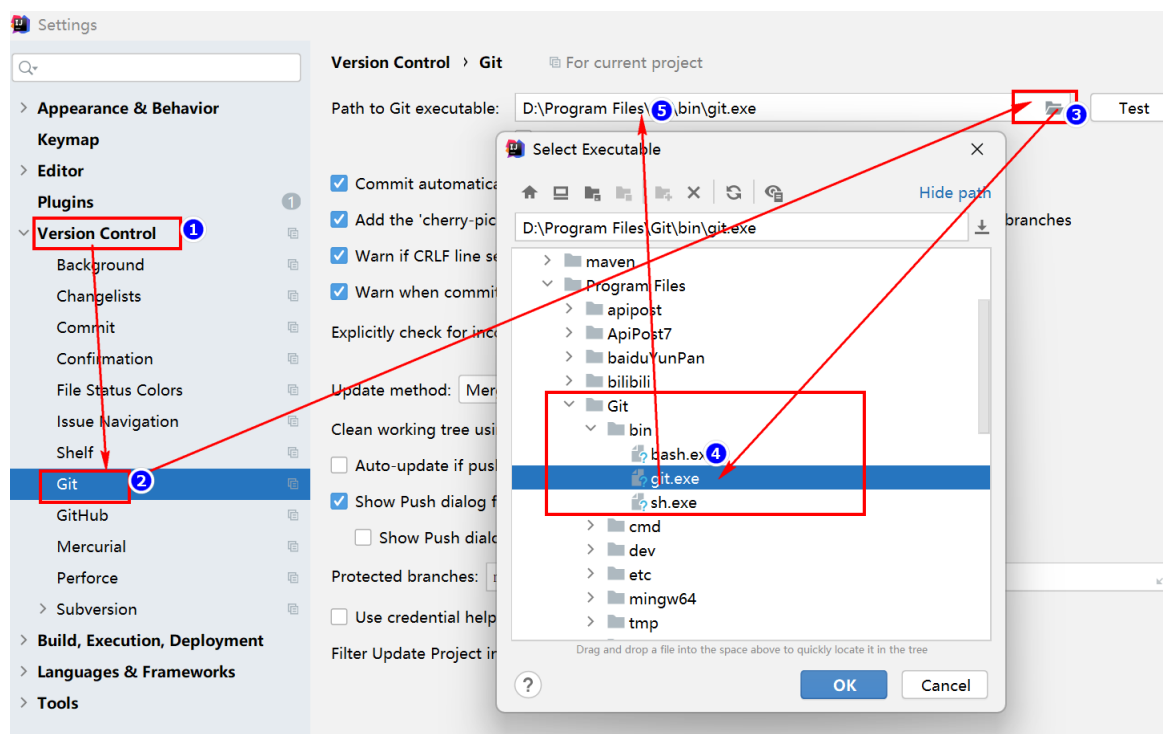
6. 使用 `git commit -m ""` 命令完成本次版本的合并

```
etoak@Damon MINGW64 ~/Desktop/student (master|MERGING)
$ git commit -m "解决了冲突"
[master f03eee4] 解决了冲突

etoak@Damon MINGW64 ~/Desktop/student (master)
$
```

7. 使用 `git push` 将合并结果推送到远程版本库

9. IDEA配置Git



10. 删除文件

- 删除版本库的文件也是对文件的修改，也需要现在本地版本库操作，然后推送到远程版本库

1. 手动删除：右击文件点击删除或者执行 `rm -rf 文件`

先执行 `git add .`

再执行 `git commit -m "xx"`

2. 使用 `git rm 文件...`

直接执行 `git commit -m "xx"`

11. 重命名文件

- 重命名版本库的文件也是对文件的修改，也需要现在本地版本库操作，然后推送到远程版本库

1. 手动重命名：右击文件点击重命名或者执行 `mv 旧文件名 新文件名`

先执行 `git add .`

再执行 `git commit -m "xx"`

2. 使用 `git mv 旧文件名 新文件名`

直接执行 `git commit -m "xx"`

今天练习

- 1、先克隆仓库：

```
git clone git@gitee.com:et2301/git-01.git
```

- 2、配置本地版本库用户名和邮箱（姓名全拼 | 姓名全拼@etoak.com）

```
cd git-01
```

```
git config --global user.name "zhangsan"
```

```
git config --global user.email "zhangsan@etoak.com"
```

- 3、使用IDEA打开工程

- 4、创建模块：姓名全拼-01

- 5、右击git-01父工程，选择Git -> Commit Directory...
提交

- 6、右击git-01父工程，选择Git-> repository -> push

假设推送失败

自动弹出对话框，选择Merge

如果有冲突，还会弹出对话框，再选择Merge

出现三列内容的对话框，解决冲突之后，点击Apply

重新push，