

系统开发工具基础

Git & LaTeX

王志巍 22020007161

2024 - 08 - 23

目录

1	Git 版本控制	1
1.1	上传 main 分支	1
1.2	克隆课程仓库	1
1.3	对克隆的仓库进行操作	2
1.3.1	将版本历史可视化	2
1.3.2	git log	2
1.3.3	查看提交信息	3
1.3.4	添加文件	3
1.3.5	删除历史	3
1.3.6	git stash	4
1.3.7	git log -all -oneline	4
1.3.8	撤销操作	4
1.4	上传更改	4
2	LaTeX	5
2.1	生成目录	5
2.2	插入代码块	5
2.2.1	代码	5
2.2.2	仍未解决的问题	5
2.3	插入图片	5
2.4	插入公式	6
2.4.1	上标和下标	6
2.4.2	分数和根号	6
2.4.3	积分	6
2.4.4	累加和累乘	6
2.5	制表	7
2.6	纸张设置	7
3	实验心得	7

1 Git 版本控制

1.1 上传 main 分支

```
Starry_sky@StarrySkyLAPTOP MINGW64 ~/Desktop/开发工具/0823 (main)
$ git push origin main
fatal: unable to access 'https://github.com/Starry-Sky-OUC/gitTest/': OpenSSL SSL_read: SSL_ERROR_SYSCALL, errno 0
```

上传失败，经过查询，问题在于之前建立连接时，使用的是 http，但是我的 GitHub 仓库使用的是 SSH。将连接重定向，并重新上传 main。

```
Starry_sky@StarrySkyLAPTOP MINGW64 ~/Desktop/开发工具/0823 (main)
$ git remote set-url origin git@github.com:Starry-Sky-OUC/gitTest.git
```

再次上传，仍然发生问题，该问题是因为创建 GitHub 仓库生成了 readme 文件，而本地仓库没有这个文件。将本地仓库上传前，先通过 rebase 选项将 GitHub 仓库同步下来。

```
Starry_sky@StarrySkyLAPTOP MINGW64 ~/Desktop/开发工具/0823 (main)
$ git push origin main
To github.com:Starry-Sky-OUC/gitTest.git
 ! [rejected]          main -> main (fetch first)
error: failed to push some refs to 'github.com:Starry-Sky-OUC/gitTest.git'
hint: Updates were rejected because the remote contains work that you do not
hint: have locally. This is usually caused by another repository pushing to
hint: the same ref. If you want to integrate the remote changes, use
hint: 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

Starry_sky@StarrySkyLAPTOP MINGW64 ~/Desktop/开发工具/0823 (main)
$ git pull origin main --rebase
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 868 bytes | 62.00 KiB/s, done.
From github.com:Starry-Sky-OUC/gitTest
 * branch          main      -> FETCH_HEAD
 * [new branch]     main      -> origin/main
warning: unable to rmdir 'missing-semester-cn.github.io': Directory not empty
Successfully rebased and updated refs/heads/main.

Starry_sky@StarrySkyLAPTOP MINGW64 ~/Desktop/开发工具/0823 (main)
$ git push origin main
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 8 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (10/10), 1.81 KiB | 927.00 KiB/s, done.
Total 10 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
To github.com:Starry-Sky-OUC/gitTest.git
 4af0843..ebf0e0a  main -> main
```

再次上传，成功。

1.2 克隆课程仓库

```
git clone
git@github.com:missing-semester-cn/missing-semester-cn.github.io.git
```

江具 > 0823 > missing-semester-cn.github.io >

名称	修改日期	类型	大小
.git	2024/8/23 11:51	文件夹	
.github	2024/8/23 11:19	文件夹	
_2019	2024/8/23 11:19	文件夹	
_2020	2024/8/23 11:19	文件夹	
_includes	2024/8/23 11:19	文件夹	
_layouts	2024/8/23 11:19	文件夹	
static	2024/8/23 11:19	文件夹	
.editorconfig	2024/8/23 11:19	Editor Config 源...	1 KB
.gitignore	2024/8/23 11:19	文本文档	1 KB
_config.yml	2024/8/23 11:19	Yaml 源文件	1 KB
404.html	2024/8/23 11:19	Microsoft Edge ...	1 KB
about.md	2024/8/23 11:19	Markdown 源文件	6 KB
apple-touch-icon.png	2024/8/23 11:19	PNG 文件	7 KB
CNAME	2024/8/23 11:19	文件	1 KB
favicon.ico	2024/8/23 11:19	图标	8 KB
favicon-16x16.png	2024/8/23 11:19	PNG 文件	4 KB
favicon-32x32.png	2024/8/23 11:19	PNG 文件	4 KB
Gemfile	2024/8/23 11:19	文件	1 KB
Gemfile.lock	2024/8/23 11:19	LOCK 文件	7 KB
index.md	2024/8/23 11:19	Markdown 源文件	5 KB
lectures.html	2024/8/23 11:19	Microsoft Edge ...	1 KB
license.md	2024/8/23 11:19	Markdown 源文件	2 KB
README.md	2024/8/23 11:19	Markdown 源文件	3 KB
robots.txt	2024/8/23 11:19	文本文档	1 KB

1.3 对克隆的仓库进行操作

1.3.1 将版本历史可视化

```
git log --all --graph --decorate
```

```
* commit ebf0e0a240dccdc78c9d82191e709ae8d6f009ce (HEAD -> main, origin/main)
| Author: Starry-Sky-OUC <1539006465@qq.com>
| Date: Fri Aug 23 11:51:52 2024 +0800
|
| 0823
|
* commit 438fa67e691e00e3dee88eef274489c3e8d5da3e
| Author: Starry-Sky-OUC <1539006465@qq.com>
| Date: Fri Aug 23 11:43:07 2024 +0800
|
| 0823
|
* commit e9b4e8420b48d017ed829696306e70b5ae5bc43a
| Author: Starry-Sky-OUC <1539006465@qq.com>
| Date: Fri Aug 23 11:29:37 2024 +0800
|
| 0823
|
* commit 4af084322b6c2a9d7f507c4969c2fb390f7072b3
| Author: Starry-Sky-OUC <115649450+Starry-Sky-OUC@users.noreply.github.com>
| Date: Fri Aug 23 11:36:56 2024 +0800
|
| Initial commit
```

1.3.2 git log

用 git log 查看是谁最后修改了 README.md 文件

```
git log -1 README.md
```

```
commit de98852ef0604cf918bab7f39c63a53932c845d8
Author: yuzq <yuzq@sunwayworld.com>
Date: Thu Jun 6 14:43:07 2024 +0800

将readme文件中的url的绝对路径改为相对路径，不用重复访问github，利于分享传播
```

图中四行分别为该次提交的哈希值、修改者、修改日期、提交信息

1.3.3 查看提交信息

```
git blame __config.yml | grep collections
```

```
a88b4eac (Anish Athalye 2020-01-17 15:26:30 -0500 18) collections:
```

根据返回值，__config.yml 文件中的第 18 行 collections: 是由 Anish Athalye 在 2020 年 1 月 17 日 15:26:30（东部标准时间）提交的，提交的哈希值是 a88b4eac。

1.3.4 添加文件

```
echo "pw" > mypw.txt
git add .
git commit -m "add mypw"
```

```
$ git log head
commit 570a015fdfcad25b5d4c51018343c4b23e79be55 (HEAD -> master)
Author: Starry-Sky-OUC <1539006465@qq.com>
Date: Wed Aug 28 12:14:44 2024 +0800

add mypw
```

1.3.5 删除历史

```
git filter-branch --force --index-filter \
'git rm --cached --ignore-unmatch ./mypw.txt' \
--prune-empty --tag-name-filter cat -- --all
```

但是 git filter-branch 在面对大型数据库时显得十分的无力，在实验中效率较低：(228/833) (239 seconds passed, remaining 634 predicted)。后根据 git 的 warning，使用 git filter-repo。

```
git filter-repo --path mypw.txt --invert-paths --force
```

```
$ git log head
commit d50a857ac3ff8fa5d6da6ae101dd76bdb94e21ad (HEAD -> master)
Merge: 7534662 1eec59
Author: Lingfeng_Ai <hanxiaomax@gmail.com>
Date: Fri Aug 16 06:54:16 2024 +0800

Merge pull request #172 from pspdada/master

Thank you so much
```

```
$ ls
404.html  Gemfile  README.md  _2020/  _includes/  about.md  favicon-16x16.png  favicon.ico  lectures.html  robots.txt
CNAME    Gemfile.lock  _2019/  _config.yml  _layouts/  apple-touch-icon.png  favicon-32x32.png  index.md  license.md  static/
```

可见，mypw.txt 已经被删除

1.3.6 git stash

```
echo "111" > testStash.txt
git add .
git stash
```

运行 git stash 时，Git 会提交当前工作目录中所有未提交的更改，并保存到一个临时存储区，然后将工作目录恢复到与最新提交一致。

1.3.7 git log --all --oneline

```
git log --all --oneline
```

运行 git log --all --oneline 时，Git 会显示所有分支的提交历史记录。由于 git stash 只是将更改保存到临时存储区，并没有创建新的提交，不会显示刚刚的 git stash 操作。

```
$ git log --all --oneline
8348391 (refs/stash) WIP on master: d50a857 Merge pull request #172 from pspdada/master
544c072 index on master: d50a857 Merge pull request #172 from pspdada/master
d50a857 (HEAD -> master) Merge pull request #172 from pspdada/master
1eec59 remove irrelevant text
0f90ef1 fix wrong index
12802cd fix typo
7534662 Merge pull request #171 from HowieChih/for-better-understanding
26fce64 更新#课程概览与 shell##一个功能全面又强大的工具关于修改亮度文件报错的翻译
ad0fbe8 更新#课程概览与 shell##一个功能全面又强大的工具关于修改亮度文件报错的翻译
a965910 Merge pull request #170 from crosscap/typo-fix
```

1.3.8 撤销操作

```
git stash pop
```

运行 git stash pop 时，Git 会将最近一次保存的更改从临时存储区应用到当前工作目录，并从临时存储区中移除这些更改。










```
$ git stash pop
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   testStash.txt

Dropped refs/stash@{0} (8348391fd68049e9309f02984313ceab8184231b)
```

1.4 上传更改

```
git add .
git commit -m "*****"
git push origin main
```

```
$ git push origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 293 bytes | 293.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:Starry-Sky-OUC/gitTest.git
   ebf0e0a..7db25de  main -> main
```

 Starry-Sky-OUC 0828		7db25de · 1 minute ago	 5 Commits
 missing-semester-cn.github.io	0828	1 minute ago	
 .gitconfig - 快捷方式.lnk	0823	5 days ago	
 .ssh - 快捷方式.lnk	0823	5 days ago	
 0828.txt	0828	1 minute ago	
 1.txt	0823	5 days ago	
 README.md	Initial commit	5 days ago	
 test0823.txt	0823	5 days ago	

2 LaTeX

2.1 生成目录

Listing 1: LaTeX 生成目录

```
\pagenumbering{roman}
\tableofcontents
\newpage
\pagenumbering{arabic}
```

2.2 插入代码块

2.2.1 代码

Listing 2: 使用 listings 包的代码块插入

```
usepackage{listings}
usepackage{xcolor}
lstset{
    basicstyle=\ttfamily\footnotesize, 字体和大小
    keywordstyle=\color{blue}, 关键字颜色
    commentstyle=\color{gray}, 注释颜色
    stringstyle=\color{red}, 字符串颜色
    backgroundcolor=\color{white}, 背景颜色
    frame=single, % 代码块边框
}
begin{lstlisting}[language=语言, caption=标题]
***
end{lstlisting}
```

2.2.2 仍未解决的问题

当我想用代码块展示代码块的代码时，出现了标签匹配相关的问题。编译器识别到代码块中的 end 标签后，认为代码块标签闭合了。lstset 中的 escapeinside 选项可以自定义转义字符，理论上可以将标签转义为单纯的字符串，从而解决这个问题，但是实际操作后并没能达成预期的效果，因此我只能将所有“\”删除来排除标签匹配的问题。

2.3 插入图片

Listing 3: LaTeX 插入图片

```
\begin{figure}
\centering
```

```
\includegraphics [ width=1\linewidth ] {***.png}
\end{figure}
```

但这个代码存在问题，这样插入的图片会出现在当前页面的最下方



1

我们可以引用 float 包可以解决这个问题，float 包提供的 [H] 选项可以将图片插入在当前位置。

Listing 4: 使用 float 包的图片插入

```
\usepackage{float}
\begin{figure}[H]
  \centering
  \includegraphics [ width=1\linewidth ] {***.png}
\end{figure}
```

2.4 插入公式

2.4.1 上标和下标

```
\usepackage{amsmath}
\begin{equation}
a_i^j = b_{i+1}^{j-1}
\end{equation}
```

$$a_i^j = b_{i+1}^{j-1} \tag{1}$$

2.4.2 分数和根号

```
x=\sqrt{x} \cdot \sqrt{x}
```

$$x = \sqrt{x} \cdot \sqrt{x} \tag{2}$$

```
\frac{a}{b} = \frac{c}{d}
```

$$\frac{a}{b} = \frac{c}{d} \tag{3}$$

2.4.3 积分

```
\int_{a}^b f(x) \, dx
```

$$\int_a^b f(x) \, dx \tag{4}$$

2.4.4 累加和累乘

```
\sum_{i=1}^n i = \frac{n(n+1)}{2}
```

$$\sum_{i=1}^n i = \frac{n(n+1)}{2} \tag{5}$$

```
\prod_{i=1}^n i = n!
```

$$\prod_{i=1}^n i = n! \tag{6}$$

2.5 制表

Listing 5: LaTeX 制表

```
\begin{center}
  \begin{tabular}{|c|c|c|c|}
    \hline
    地点 & 8/27 & 8/28 & 8/29 \\
    \hline
    青岛 & 29°C & 31°C & 32°C \\
    \hline
  \end{tabular}
\end{center}
```

地点	8/27	8/28	8/29
青岛	29°C	31°C	32°C

2.6 纸张设置

```
\usepackage[a4paper, total={6in, 8in}]{geometry}
/*****/
a4paper      将纸张大小设置为A4，默认为A4
/*****/

\usepackage[legalpaper, margin=1in]{geometry}
/*****/
landscape    将纸张设置为横向，默认为纵向
margin       设置页边距，默认为2英寸
/*****/
```

3 实验心得

通过本次实验学习了 git 版本控制相关的操作，包括版本管理、分支及其合并、记录查看、操作撤销等有利于项目管理的强大功能。此外，学习了 LaTeX 相关语法，各等级标签、自动生成目录、数学公式、制表等功能，能帮助我们做出美观的报告、论文，这都对我们今后的学习生涯有很大的帮助。因为有过 Git 的使用经验，也学习过前端相关知识，本次实验进行较为顺利，帮我复习了很多遗忘的知识。