### 2019.07.12学习笔记(a.m)

**Git相关：**

1. **新建一个文件夹，进入该文件夹；**

1）$ cd c:（进入c盘）

2）$ cd \aa（进入aa目录）

3）$ pwd

/c/aa（显示当前工作目录）

1. **git init：初始化一个仓库，会生成一个该目录下生成一个.git隐藏目录，此时该文件夹就变成了一个仓库，其中.git隐藏目录里面的内容很重要，不能随意修改；**

1）$ git init（初始化仓库，当前工作在哪个目录下，就把哪个目录初始化为一个仓库）

Reinitialized existing Git repository in C:/aa/.git/

2）$ ls -ah（查看隐藏文件夹）

./ ../ .git/

1. **在该文件夹里新建一个readme.txt文件；**

$ touch readme.txt（或者使用$ vi readme.txt命令直接新建并进入编辑模式）

1. **使用git add readme.txt将改文件添加到缓冲区；**

$ git add readme.txt

warning: LF will be replaced by CRLF in readme.txt.

The file will have its original line endings in your working directory.

（正常情况下没有warning，不过并不影响正常工作，此问题待解决）。

1. **使用git commit -m “注释内容”将改文件提交到git仓库；**

$ git commit -m "wrote a read.txt file"

[master (root-commit) 5e7297f] wrote a read.txt file

1 file changed, 2 insertions(+)

create mode 100644 readme.txt

（1 file changed, 2 insertions(+)：表示1个文件被修改，2个插入内容）

1. **使用git status查看工作区的状态；**

$ git status

On branch master

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git checkout -- <file>..." to discard changes in working directory)

modified: readme.txt

（表示readme.txt被修改了）

1. **使用git diff查看修改的内容（diff就是difference的缩写，即不同）**
2. **git中，—（红色）表示原来的内容，+（绿色）表示修改过后的内容，没有—、+（白色）则表示没有被修改过；**

$ git diff readme.txt

warning: LF will be replaced by CRLF in readme.txt.

The file will have its original line endings in your working directory.

**diff --git a/readme.txt b/readme.txt**

**index 46d49bf..9247db6 100644**

**--- a/readme.txt**

**+++ b/readme.txt**

@@ -1,2 +1,2 @@

-Git is a version control system.

+Git is a distributed version control system.

Git is free software.

1. **使用git log来看我们都做了哪些修改，即历史记录；**

$ git log

commit 41feac5dbd5ccce7c5fc13d6583548c89b4d8c78 (HEAD -> master)

Author: Helay <1562823234@qq.com>

Date: Fri Jul 12 09:42:39 2019 +0800

append GPL

commit 13f92ad816df04963757410ca3b18e94194a9353

Author: Helay <1562823234@qq.com>

Date: Fri Jul 12 09:30:07 2019 +0800

add distributed

commit 5e7297f6a62a1c070470c676ea2380ab3956b8b9

Author: Helay <1562823234@qq.com>

Date: Fri Jul 12 08:51:10 2019 +0800

wrote a read.txt file

1. **使用命令git log --pretty=oneline可以使修改记录简洁的显示，即每修改一次在显示一行；**

$ git log --pretty=oneline

41feac5dbd5ccce7c5fc13d6583548c89b4d8c78 (HEAD -> master) append GPL

13f92ad816df04963757410ca3b18e94194a9353 add distributed

5e7297f6a62a1c070470c676ea2380ab3956b8b9 wrote a read.txt file

**友情提示**：**你看到的一大串类似1094adb...的是commit id（版本号），用HEAD表示当前版本，也就是最新的提交1094adb...（注意我的提交ID和你的肯定不一样），上一个版本就是HEAD^，上上一个版本就是HEAD^^，当然往上100个版本写100个^比较容易数不过来，所以写成HEAD~100。**

1. **使用cat命令查看文件内容**

$ cat readme.txt

Git is a distributed version control system.

Git is free software distributed under the GPL.

1. **使用git reset命令来进行版本回退**

1）$ git reset --hard HEAD^

HEAD is now at 13f92ad add distributed

2）$ git reset --hard 41feac

HEAD is now at 41feac5 append GPL

**（想要回到哪个版本，就用git reset --hard 版本号）**

1. **使用git reflog来记录每一次操作的命令**

$ git reflog

41feac5 (HEAD -> master) HEAD@{0}: reset: moving to 41feac

13f92ad HEAD@{1}: reset: moving to HEAD^

41feac5 (HEAD -> master) HEAD@{2}: commit: append GPL

13f92ad HEAD@{3}: commit: add distributed

5e7297f HEAD@{4}: commit (initial): wrote a read.txt file

**重点**：

1. **区分git add和git commit的区别**

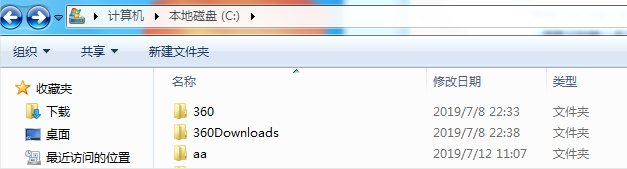
为什么Git添加文件需要add，commit一共两步呢？因为commit可以一次提交很多文件，所以你可以多次add不同的文件，比如：

$ git add file1.txt

$ git add file2.txt file3.txt

$ git commit -m "add 3 files."

1. **工作区和暂存区**

1）工作区（Working Directory）：电脑里能看到的目录，比如我的aa文件夹就是一个工作区：2）版本库（Repository）：aa工作区有一个隐藏目录.git，它不是工作区，而是Git的版本库。

Git的版本库里存了很多东西，其中最重要的就是称为stage的暂存区，还有Git为我们自动创建的第一个分支master，以及指向master的一个指针叫HEAD。

1. git add命令实际上就是把要提交的所有修改放到暂存区（Stage），然后，执行git commit就可以一次性把暂存区的所有修改提交到分支。
2. 执行完git add命令：



执行完git commit命令：

