GITHUB

2018年10月18日 星期四   下午12:52

What is github?
- Publishing tool?collaberation?
- A web based hosted service for Git repositories. Git is a popular open-source version control system. Github allows you to host aremote Gite repositories, and has a wealth of community-based services that make it ideal for open source projects.
  gitHub是一个面向开源及私有项目的的托管平台，因为只支持git作为唯一的版本库格式进行托管，故名gitHub
  Cloud-based Git service

  Repositories=project
  Modernizr 是一个用来检测浏览器功能支持情况的 JavaScript库。通过这个库我们可以检测不同的浏览器对于HTML5特性的支持情况。
- Buttons watch   watch this repositories and make changes
                   Star love this thing
                   Fork :make a copy and work on yourself

Three advantages
- Versioning project Have version in control
- Publishing project
- Collaborating with teams

Set up Git -> create repositories -> fork repositories ->be social
http:// training.github.com/kit

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency
http://git-scm.com

Centralized version control systems VS. Distributed VCS
- Git is a distributed VCS
- Each client keeps a local mirror of the project repository

efficiency
http://git-scm.com

Centralized version control systems VS. Distributed VCS
- Git is a distributed VCS
- Each client keeps a local mirror of the project repository
- If the individual mirrors are lost, simply cloning another client can restore them

03-02 Using the command line
Use the terminal to navigate around
 -> pwd-- present working directory
 -> cd- change directory : cd~ (represent that whole path)"~"英文符号叫" Tilde"
 -> example: cd ~/desktop/exercise_files

　　　　　　　　ls (显示具体的文件)

　　　　　　　　Mkdir terminal(文件夹名字)
-> mkdir make directory
-> rm -r terminal (remove directory)

03_05 How Git works

- Working directory (root)
- Staging area (index)
- Git repo (.git)

3 statement of a file
- Unmodified
- Modified (available for next commit)
- Staged

Branching
Photo gallery branch->master branch

03-06 Creating a repo

```
zhangxintongs-MacBook-Pro:~ elsazhang$ cd ~/desktop
zhangxintongs-MacBook-Pro:desktop elsazhang$ mkdir repos
zhangxintongs-MacBook-Pro:desktop elsazhang$ mkdir repos/git-basics
zhangxintongs-MacBook-Pro:desktop elsazhang$ cd repos/git-basics
zhangxintongs-MacBook-Pro:git-basics elsazhang$ pwd
/Users/elsazhang/desktop/repos/git-basics
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git init
Initialized empty Git repository in /Users/elsazhang/Desktop/repos/git-basics/.git/
zhangxintongs-MacBook-Pro:git-basics elsazhang$
Checking status
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git status
On branch master
```

/Users/elsazhang/desktop/repos/git-basics
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git init
Initialized empty Git repository in /Users/elsazhang/Desktop/repos/git-

zhangxintongs-MacBook-Pro:git-basics elsazhang$
Checking status
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
Adding files
    -   Git add (+file name)

zhangxintongs-MacBook-Pro:git-basics elsazhang$ git add index.html
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git status (检查状态：是否成
功)
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

new file: index.html

Untracked files:
  (use "git add <file>..." to include in what will be committed)

.DS_Store
_css/
_scripts/

Add all our new files and tell git that we actually want to track these
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git add .
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

new file: .DS_Store
new file: _css/main.css
new file: _scripts/modernizr-custom.js
new file: _scripts/respond.min.js
new file: _scripts/scripts.js
new file: _scripts/vendor/jquery-1.10.1.min.js
new file: _scripts/vendor/modernizr-2.6.2-respond-1.1.0.min.js
new file: index.html

Making a commit
git commit 是将本地修改的内容提交的缓存中，一般这么使用 git

commit -m "XXXXX"
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git commit -m "added a
readme file"
[master (root-commit) 56a617e] added a readme file

new file: index.html

git commit 是将本地修改的内容 提父的缓存 中 一般这么使用 git commit -m "XXXXX"

```
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git commit -m "added a readme file"
[master (root-commit) 569617e] added a readme file
```

Modifying files

在文件中更改以后，保存好，在终端里进入文件夹，然后输入：git status

Checking differences

```
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git diff
diff --git a/index.html b/index.html
index 5829ef8..606de78 100755
--- a/index.html
+++ b/index.html
@@ -37,7 +37,7 @@ Modernizr.load([
       // test for media query support, if not load respond.js
     test : Modernizr.mq('only all'),
       // If not, load the respond.js file
-    nope : 'js/respond.min.js'
+    nope : '_scripts/respond.min.js'
 }
 ]);
 </script>
```

如果想将修正后的内容存储，输入：

```
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git add index.html
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)
```

```
modified:   index.html
```

```
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git diff
zhangxintongs-MacBook-Pro:git-basics elsazhang$ （不再出现原版和
```

修正版的对比）

若输入：git diff --staged

```
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git commit -m 'updated path
```

若输入：`git diff --staged`

还是会显示原来和现在的对比

zhangxintongs-MacBook-Pro:git-basics elsazhang$ `git commit -m 'updated path for respond.js'`（提交）
[master c953053] updated path for respond.js
 1 file changed, 1 insertion(+), 1 deletion(-)
zhangxintongs-MacBook-Pro:git-basics elsazhang$ git status
On branch master
nothing to commit, working tree clean
zhangxintongs-MacBook-Pro:git-basics elsazhang$