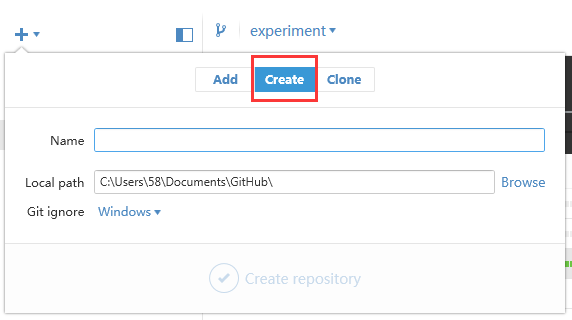
1. **创建一个仓库**

操作：打开 GitHub 图形界面，点击左上角 + ，选择 Create 。

**> git init**



在这里Git 会默认给我们创造2个文件



1. **修改文件，查看文件状态，再提交**

操作：打开对应文件夹，增减改文件。

1. 有变化的文件，都会出现在Changes 里面

Change 对应的是 Status: Untracked / Modify/Stadged

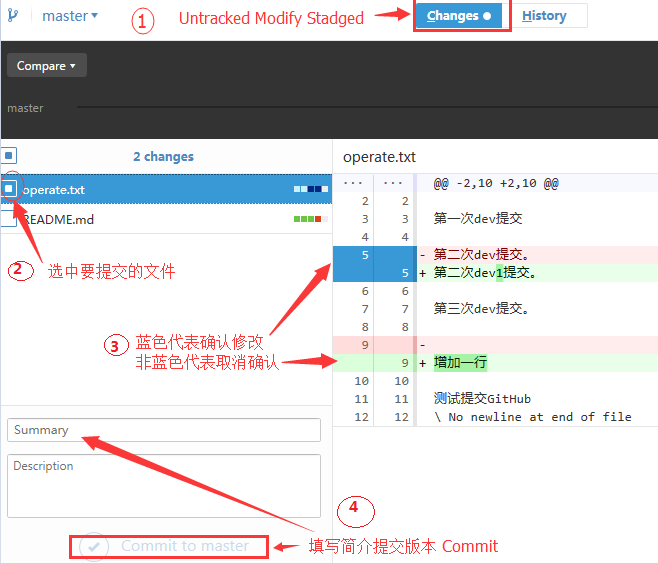
**> git status**

1. 选中你要确认提交的文件
2. 确认修改的地方
3. 提交

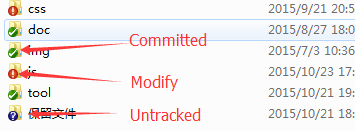
**> git commit –m “summary”**

提交后Changes 的文件都消失了。

这些我们也归为一类。叫做 **Committed** (已加入版本的文件)



附 SVN 状态对比图



1. **创建分支合并分支**

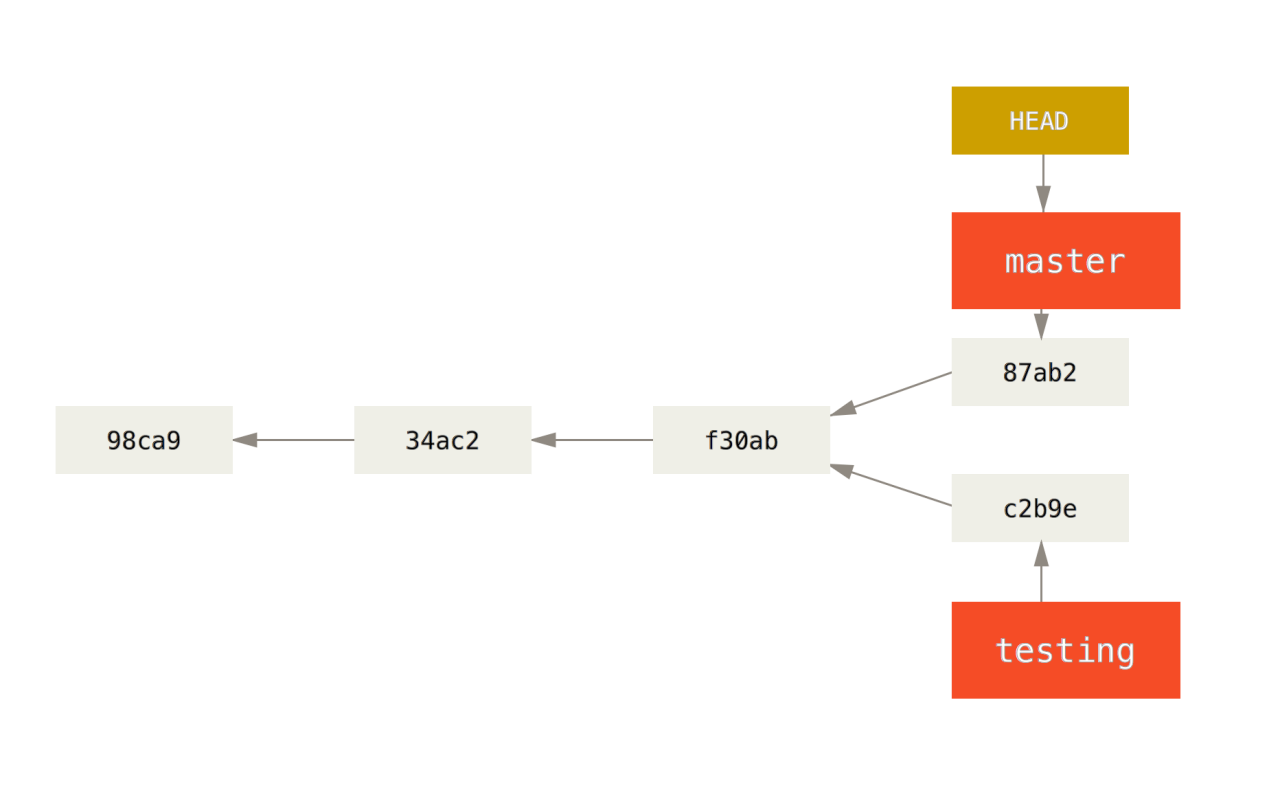
首先我们给 branch 分支下一个简单的定义便于理解。

File 是什么？ Js/css 文件

Commit 是什么？ File在一个时间点的集合

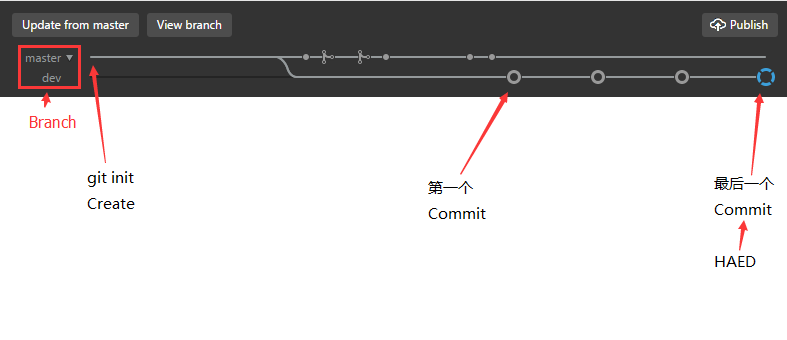
Branch 是什么？ Commit在时间轴上的有序集合

File 有两个版本的提交，就会有两个状态的 Commit ，就会导致出不同的 Branch。



但在 Git 上，Branch 实际上只是一个指针，详细说明的看文档。

<http://git-scm.com/book/zh/v2/Git-%E5%88%86%E6%94%AF-%E5%88%86%E6%94%AF%E7%AE%80%E4%BB%8B>



操作：

①新建分支

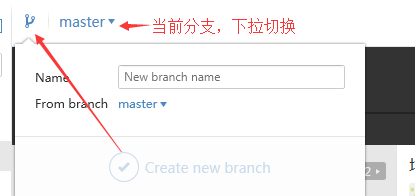
> git branch dev

②查看分支

> git branch

③切换分支

> git checkout dev



在这个 dev 分支上尽情 修改File，然后 commit 都会不影响 master 分支的内容。等开发完了，再把 file 合并到 master 分支上。

1. 合并分支