Github

* 定义：

gitHub是一个面向[开源](https://baike.baidu.com/item/%E5%BC%80%E6%BA%90/20720669)及私有[软件](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6/12053)项目的托管平台，因为只支持git 作为唯一的版本库格式进行托管，故名gitHub。

* 使用github：

<http://blog.csdn.net/javaandroid730/article/details/53522872>

* 注册：

**. 点击绿色按钮Sign up for GitHub，填写用户名，邮件地址和密码。**

注意：

1.用户名只能包含字母和"-"字符

2.密码至少包含一个字母，一个数字和七个字符

**点击绿色按钮Create an account，创建账号。**

* 注册教程：

<http://blog.csdn.net/Take_Dream_as_Horse/article/details/64920991>

第1步。创建一个存储库

一个**库**通常用于举办单个项目。存储库可以包含文件夹和文件，图像，视频，电子表格和数据集 - 您的项目需要的任何东西。我们建议包括一个*自述*文件或包含关于您项目信息的文件。GitHub可以在创建新存储库的同时轻松添加一个。*它还提供其他常见选项，如许可证文件。*

您的hello-world存储库可以是您存储想法，资源甚至与其他人共享和讨论事物的地方。

创建一个新的存储库

* 在右上角，在你的头像或身份证旁边，点击 然后选择**新的存储库**。
* 命名您的存储库hello-world。
* 写一个简短的描述。
* 选择**使用自述文件初始化此存储库**。



点击**创建存储库**。

创建一个新的分支

* 转到您的新存储库hello-world。
* 单击文件列表顶部的下拉列表中的**分支：master**。
* readme-edits在新的分支文本框中输入分支名称。
* 选择蓝色的**创建分支**框或点击键盘上的“Enter”。
* 现在你有两个分支，master并且readme-edits。它们看起来完全一样，但时间不长！接下来，我们会将我们的更改添加到新分支。

第3步。进行并提交更改

好样的！现在，您在您的readme-edits分支的代码视图中，它是副本master。我们来做一些编辑。

在GitHub上，保存的更改称为*提交*。每个提交都有一个关联的*提交消息*，它是解释为什么进行特定更改的说明。提交消息捕获您所做更改的历史记录，所以其他贡献者可以了解您所做的工作以及原因。

**进行并提交更改**

* 点击该README.md文件。
* 点击  铅笔图标在文件视图的右上角进行编辑。
* 在编辑器中，写一些关于你自己的信息。
* 编写描述您的更改的提交消息。
* 单击**提交更改**按钮。

第4步。打开合并请求

好的编辑！现在您已关闭分支中的更改master，您可以打开*拉取请求*。

合并请求是GitHub协作的核心。当你打开一个*拉取请求时*，你正在提出你的改变，并要求有人审查并提交你的贡献，并将它们合并到他们的分支中。拉取请求显示来自两个分支的内容的*差异*或差异。更改，添加和减法以绿色和红色显示。

一旦你提交了，你甚至可以在代码完成之前打开一个pull请求并开始讨论。

通过在您的拉取请求消息中使用GitHub的[@mention HYPERLINK "https://help.github.com/articles/about-writing-and-formatting-on-github/"系统](https://help.github.com/articles/about-writing-and-formatting-on-github/)，您可以要求特定人员或团队提供反馈，无论他们是否在大厅或10个时区以外。

你甚至可以在自己的仓库中打开pull请求并自己合并。在开展大型项目之前，学习GitHub流程是一种很好的方式。

**打开合并请求以更改自述文件**

*点击图像查看大图*

|  |  |
| --- | --- |
| **步** | **截图** |
| 点击  **合并请求**标签，然后从合并请求页面点击绿色的**新建请求**按钮。 |  |
| 在“ **示例比较”**框中，选择您创建的分支readme-edits以与master（原始）进行比较。 |  |
| 查看比较页面上差异的更改，确保它们是您想要提交的内容。 |  |
| 如果您满意这些是您要提交的更改，请点击绿色的“ **创建合并请求”**按钮。 |  |
| 给你的拉取请求一个标题，并写一个你的改变的简要说明。 |  |

完成消息后，请点击**创建拉取请求**！

**提示**：您可以使用[的表情符号](https://help.github.com/articles/basic-writing-and-formatting-syntax/)和[拖放图片和 HYPERLINK "https://help.github.com/articles/file-attachments-on-issues-and-pull-requests/"GIF](https://help.github.com/articles/file-attachments-on-issues-and-pull-requests/)到的意见和引入请求。

第5步。合并您的请求

在这最后一步中，是时候把你的修改结合起来 - 将你的readme-edits分支合并到master分支中。

* 点击绿色**合并拉取请求**按钮，将更改合并到master。
* 点击**确认合并**。
* 继续操作，删除分支，因为它的更改已经合并，并且在紫色框中带有**删除分支**按钮。



**上传代码：**

<http://blog.csdn.net/llf369477769/article/details/51917557>

<https://www.cnblogs.com/shenchanghui/p/7184101.html>

自己的总结：

上传一个文件的步骤：

安装git客户端：

windows用户请下载 <http://msysgit.github.com/>

mac用户请下载 <http://code.google.com/p/tortoisegit/>

我们先在电脑硬盘里找一块地方存放本地仓库，比如我们把本地仓库建立在C:\MyRepository\1ke\_test文件夹下

进入1ke\_test文件夹 鼠标右键操作如下步骤：

1）在本地仓库里右键选择Git Init Here，会多出来一个.git文件夹，这就表示本地git创建成功。右键Git Bash进入git命令行，截图效果如下：

为了保险起见，我们先执行git init命令

注意：如果右键没有Git Init Here也没事，执行命令后.git文件出来就可以了。

2）为了把本地的仓库传到github，还需要配置ssh key。

$ ssh-keygen -t rsa -C "your\_email@youremail.com"

[your\_email@youremail.com HYPERLINK "mailto:your\_email@youremail.com是自己git"是自己 HYPERLINK "mailto:your\_email@youremail.com是自己git"git](mailto:your_email@youremail.com是自己git)账户的注册邮箱。

直接点回车，说明会在默认文件id\_rsa上生成ssh key。

然后系统要求输入密码，直接按回车表示不设密码

重复密码时也是直接回车，之后提示你shh key已经生成成功。

然后我们进入提示的地址下查看ssh key文件。 我的电脑的地址是C:\Users\lilu\.ssh ，其中lilu是我的电脑的名称，自己去找.ssh文件去。

打开id\_rsa.pub，复制里面的key。里面的key是一对看不懂的字符数字组合，不用管它，直接复制。

回到github网站，进入Account Settings，左边选择SSH Keys，Add SSH Key,

title随便填，粘贴key。

3）验证是否成功，在git bash下输入

$ ssh -T [git@github.com](mailto:git@github.com)

回车类似：You’ve successfully authenticated, but GitHub does not provide shell access 。这就表示已成功连上github。

4）接下来我们要做的就是把本地仓库传到github上去，在此之前还需要设置username和email，因为github每次commit都会记录他们

$ git config --global user.name "your name"

$ git config --global user.email "your\_email@youremail.com"

自己的git的账户名，和邮箱

5）进入要上传的仓库，右键git bash，添加远程地址

$ git remote add origin git@github.com:yourName/yourRepo.git

yourName：自己的git账户名

yourRepo：自己的仓库名

例如：$ git remote add origin git@github.com:yourName/yourRepo

**提交上传：**

* 接下来在本地仓库里添加一些文件，例如log.txt

然后输入命令：

$ git add log.txt

$ git commit (-m “提交说明，例如改bug的时候的说明”)

所以只要有commit就能把本地仓库的东西提交到git仓库上。



* push前先将远程repository修改pull下来

$ git pull origin master

* 上传到github仓库

$ git push origin master

更新代码

第一步：查看当前的git仓库状态，可以使用git status

git status

第二步：更新全部

git add \* 也就是说add \* 了就是把所有的文件都更新了

第三步：接着输入git commit -m "更新说明"

git commit -m "更新说明"

第四步：先git pull,拉取当前分支最新代码

git pull

第五步：push到远程master分支上

git push origin master

不出意外，打开GitHub已经同步了

EntityFrameWorker表基本创建

地址：<http://blog.csdn.net/quryktcs/article/details/71189721>

<https://www.cnblogs.com/CreateMyself/p/4739922.html>

<https://www.cnblogs.com/liuyuqing/p/6054774.html>

<http://www.cr173.com/html/20725_1.htm>

用first model画一下表结构

<https://www.cnblogs.com/DrHao/p/5367196.html>

[**ASP.NET WebAPI OData (V4)**](http://www.cnblogs.com/bluedoctor/p/4384659.html)

<https://www.cnblogs.com/bluedoctor/p/4384659.html>