Git完整提交记录

包含提示信息和提交指令，提交指令采用//进行显式指出

01@DESKTOP-I31CNV2 MINGW64 ~/Documents/Arduino/TAN (master)

1.//$ git init

Initialized empty Git repository in C:/Users/01/Documents/Arduino/TAN/.git/

01@DESKTOP-I31CNV2 MINGW64 ~/Documents/Arduino/TAN (master)

2.//$ git add TAN.ino

01@DESKTOP-I31CNV2 MINGW64 ~/Documents/Arduino/TAN (master)

3.//$ git commit -m "final demo"

[master (root-commit) 8b308c4] final demo

1 file changed, 1137 insertions(+)

create mode 100644 TAN.ino

01@DESKTOP-I31CNV2 MINGW64 ~/Documents/Arduino/TAN (master)

4.//$ git remote add origin https://github.com/Perceptive-Intelligence/numcat.git

01@DESKTOP-I31CNV2 MINGW64 ~/Documents/Arduino/TAN (master)

5.//$ git pull origin master --allow-unrelated-histories

remote: Enumerating objects: 34, done.

remote: Counting objects: 100% (34/34), done.

remote: Compressing objects: 100% (24/24), done.

remote: Total 34 (delta 6), reused 28 (delta 3), pack-reused 0 (from 0)

Unpacking objects: 100% (34/34), 24.71 KiB | 232.00 KiB/s, done.

From https://github.com/Perceptive-Intelligence/numcat

\* branch master -> FETCH\_HEAD

\* [new branch] master -> origin/master

Merge made by the 'ort' strategy.

Gituser.docx | Bin 0 -> 13793 bytes

README.md | 4 +

"USB\357\274\210http\357\274\211/USB/OLED.cpp" | 22 +++

"USB\357\274\210http\357\274\211/USB/OLED.h" | 24 +++

"USB\357\274\210http\357\274\211/USB/USB.ino" | 197 ++++++++++++++++++++

change\_update.js | 137 ++++++++++++++

red\_line\_update.ino | 239 +++++++++++++++++++++++++

7 files changed, 623 insertions(+)

create mode 100644 Gituser.docx

create mode 100644 README.md

create mode 100644 "USB\357\274\210http\357\274\211/USB/OLED.cpp"

create mode 100644 "USB\357\274\210http\357\274\211/USB/OLED.h"

create mode 100644 "USB\357\274\210http\357\274\211/USB/USB.ino"

create mode 100644 change\_update.js

create mode 100644 red\_line\_update.ino

01@DESKTOP-I31CNV2 MINGW64 ~/Documents/Arduino/TAN (master)

6.//$ git push origin master

Enumerating objects: 6, done.

Counting objects: 100% (6/6), done.

Delta compression using up to 12 threads

Compressing objects: 100% (4/4), done.

Writing objects: 100% (5/5), 10.33 KiB | 10.33 MiB/s, done.

Total 5 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/Perceptive-Intelligence/numcat.git

1a70cf9..f9a90fa master -> master

提示可以使用（参数已经设置好，自主设置文件名称即可，其他不用动），复制使用ctrl+c，在bash界面使用shift+insert进行粘贴

下面是修改之后再次进行提交上传的记录：

01@DESKTOP-I31CNV2 MINGW64 /e/Git (master)

1.//$ git pull origin master

remote: Enumerating objects: 6, done.

remote: Counting objects: 100% (6/6), done.

remote: Compressing objects: 100% (3/3), done.

remote: Total 5 (delta 1), reused 5 (delta 1), pack-reused 0 (from 0)

Unpacking objects: 100% (5/5), 10.31 KiB | 78.00 KiB/s, done.

From https://github.com/Perceptive-Intelligence/numcat

\* branch master -> FETCH\_HEAD

1a70cf9..f9a90fa master -> origin/master

Updating 1a70cf9..f9a90fa

Fast-forward

TAN.ino | 1137 +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

1 file changed, 1137 insertions(+)

create mode 100644 TAN.ino

01@DESKTOP-I31CNV2 MINGW64 /e/Git (master)

2.//$ git add Gituser.docx(文件名)

01@DESKTOP-I31CNV2 MINGW64 /e/Git (master)

$ git commit -m "Updated"

[master 98100ca] Updated

1 file changed, 0 insertions(+), 0 deletions(-)

01@DESKTOP-I31CNV2 MINGW64 /e/Git (master)

3.//$ git push origin master

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Delta compression using up to 12 threads

Compressing objects: 100% (3/3), done.

Writing objects: 100% (3/3), 11.76 KiB | 11.76 MiB/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 https://github.com/Perceptive-Intelligence/numcat.git

f9a90fa..98100ca master -> master

下面是添加新文件夹下的文件进行上传的实例：

01@DESKTOP-I31CNV2 MINGW64 ~/WPSDrive/891486363/WPS云盘/numcat-master (git上传)/numcat-master (master)

$ git pull origin master

From https://github.com/Perceptive-Intelligence/numcat

\* branch master -> FETCH\_HEAD

Already up to date.

01@DESKTOP-I31CNV2 MINGW64 ~/WPSDrive/891486363/WPS云盘/numcat-master (git上传)/numcat-master (master)

$ git add /TAN/Document

fatal: Invalid path 'E:/Git/Git/TAN': No such file or directory

01@DESKTOP-I31CNV2 MINGW64 ~/WPSDrive/891486363/WPS云盘/numcat-master (git上传)/numcat-master (master)

$ git add TAN/Document

（注意添加多一个/就表示去到根目录了，当然找不到啦）

01@DESKTOP-I31CNV2 MINGW64 ~/WPSDrive/891486363/WPS云盘/numcat-master (git上传)/numcat-master (master)

$ git commit -m "Updated Document"

[master fc0800b] Updated Document

5 files changed, 0 insertions(+), 0 deletions(-)

create mode 100644 "TAN/Document/MES\346\272\220\347\240\201\344\272\214\346\254\241\345\274\200\345\217\221\345\256\232\345\210\266\346\211\213\345\206\214.docx"

create mode 100644 "TAN/Document/MES\346\272\220\347\240\201\350\260\203\350\257\225\346\211\213\345\206\214.docx"

create mode 100644 "TAN/Document/Windows\345\270\270\347\224\250\345\277\253\346\215\267\351\224\256\346\211\213\345\206\214.docx"

create mode 100644 "TAN/Document/~$S\346\272\220\347\240\201\344\272\214\346\254\241\345\274\200\345\217\221\345\256\232\345\210\266\346\211\213\345\206\214.docx"

create mode 100644 "TAN/Document/~$S\346\272\220\347\240\201\350\260\203\350\257\225\346\211\213\345\206\214.docx"

01@DESKTOP-I31CNV2 MINGW64 ~/WPSDrive/891486363/WPS云盘/numcat-master (git上传)/numcat-master (master)

$ git push origin master

Enumerating objects: 10, done.

Counting objects: 100% (10/10), done.

Delta compression using up to 12 threads

Compressing objects: 100% (7/7), done.

Writing objects: 100% (7/7), 1.63 MiB | 802.00 KiB/s, done.

Total 7 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/Perceptive-Intelligence/numcat.git

25d6726..fc0800b master -> master

这里有时候（看自己设置的令牌自动更新时间）会要求更新token，大致操作如下：

总结一下其实就是生成一个登录的身份令牌，值得注意的是一定要勾选一个repo用于确认访问权限

以下是生成个人访问令牌的步骤：

1. **登录 GitHub** 并点击页面右上角的头像，选择 **Settings**（设置）。
2. 在设置页面左侧的底部找到 **Developer settings**。
3. 进入 **Personal access tokens**，然后选择 **Tokens (classic)**。
4. 点击 **Generate new token**。
5. 输入一个描述（例如：“Git 操作”），选择 **过期时间**，并在权限中勾选 repo（用于访问您的仓库）。
6. 点击 **Generate token** 生成令牌。
7. GitHub 将显示生成的令牌，**请立即复制并保存**，因为此页面关闭后无法再次查看。

然后使用git push会提示登录github，这个时候我们选择token登录将生成的token粘贴进行登录即可