

数据分析平台手册

[V1.0.0]

[2021-07-04]

一、 目录.....	错误！未定义书签。
一、 环境准备.....	3
二、 平台代码获取、使用、开发、上传及版本管理.....	3
三、 当前平台库函数说明.....	5
四、 算法结果入库及部署.....	5

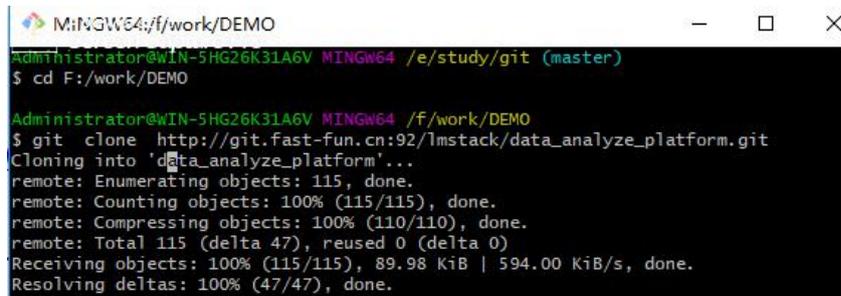
一、 环境准备

- python 3.8 (anaconda)
- git (版本管理工具)
- vscode (代码编辑开发工具, 集成了 git、 jupyter)

二、 平台代码获取、使用、开发、上传及版本管理

1. 平台代码获取

- 1) 运行 git bash
- 2) cd F:/work/DEMO, 进入个人工作的目录
- 3) git clone http://git.fast-fun.cn:92/lmstack/data_analyze_platform.git

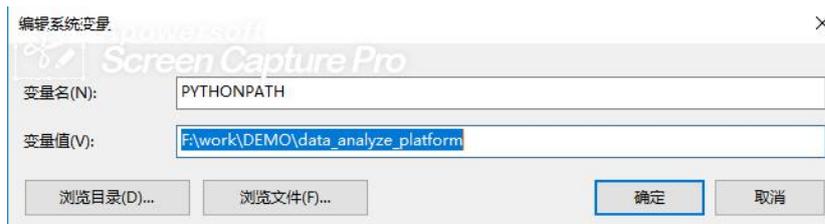


```
MINGW64:~/work/DEMO
Administrator@WIN-5HG26K31A6V MINGW64 /e/study/git (master)
$ cd F:/work/DEMO

Administrator@WIN-5HG26K31A6V MINGW64 /f/work/DEMO
$ git clone http://git.fast-fun.cn:92/lmstack/data_analyze_platform.git
Cloning into 'data_analyze_platform'...
remote: Enumerating objects: 115, done.
remote: Counting objects: 100% (115/115), done.
remote: Compressing objects: 100% (110/110), done.
remote: Total 115 (delta 47), reused 0 (delta 0)
Receiving objects: 100% (115/115), 89.98 KiB | 594.00 KiB/s, done.
Resolving deltas: 100% (47/47), done.
```

2. 平台代码使用

- 1) 新建环境变量 PYTHONPATH, 将平台路径添加到环境变量中, 此处为 F:/work/DEMO/ data_analyze_platfom



- 2) 运行 vscode, 打开工作文件夹, 后续即可在该目录下工作。(为使环境变量生效, 需要关闭所有 vscode 后重新打开)

文件 -> 打开文件夹 -> F:/work/DEMO/ data_analyze_platform

- 3) 运行 demo.ipynb 获取数据代码块进行测试。

- 需要配置 python 的环境, 以及通过 pip 安装必要的工具包 (pandas, numpy, numba 等);

如果使用 Anaconda, 可直接选择对应的 python.exe 即可)

- 执行获取数据代码, 如果获取成功, 则平台配置完成。

3. 平台代码开发

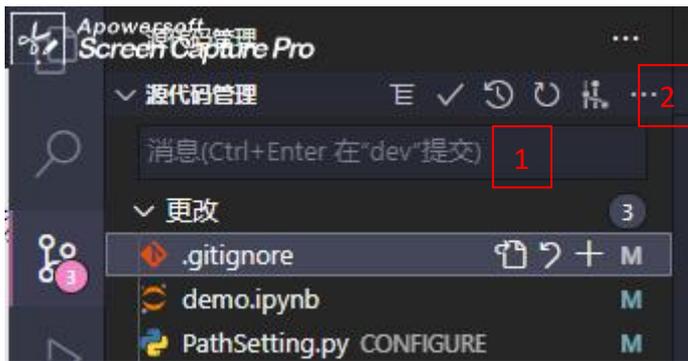
- 1) 建立新的文件夹 (此处为 TEST 文件夹), 进行代码开发工作。
- 2) 在该路径下进行功能的开发, 开发成熟的代码, 可以准备上传至库中。

4. 平台代码上传

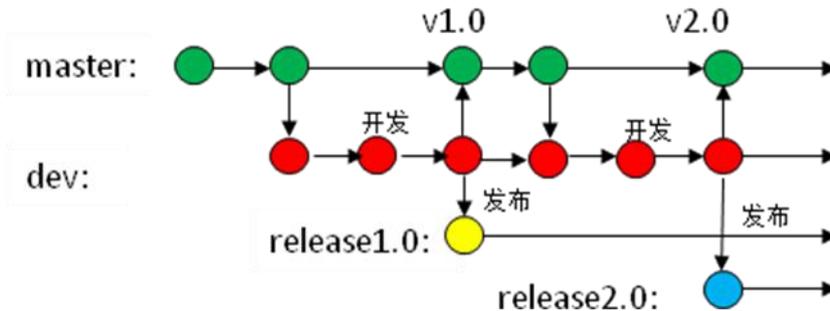
- 1) 将第 3 步中开发完成的代码,根据功能分类 移动至 LIB/BACKEND, LIB/MIDDLE, LIB/FRONTEND 中。
- 2) 修改.gitignore 文件, 加入/TEST/, 保证第三步中新建的个人开发文件夹不进行提交及版本管理
- 3) 确认当前处于 dev 分支



- 4) 在 <http://git.fast-fun.cn:92/> 注册账号, 并让管理员添加至项目协作中。
- 5) 使用 vscode 的 git 工具, 提交更改到本地并推送至远程仓库。
 - 在 1 位置填入本次提交的说明, 然后点击 ✓, 代码便可提交至本地仓库; (git commit)
 - 点击 2 处, 选择推送, 若无报错, 则代码已推送至远程仓库; (git push)



5. 版本管理

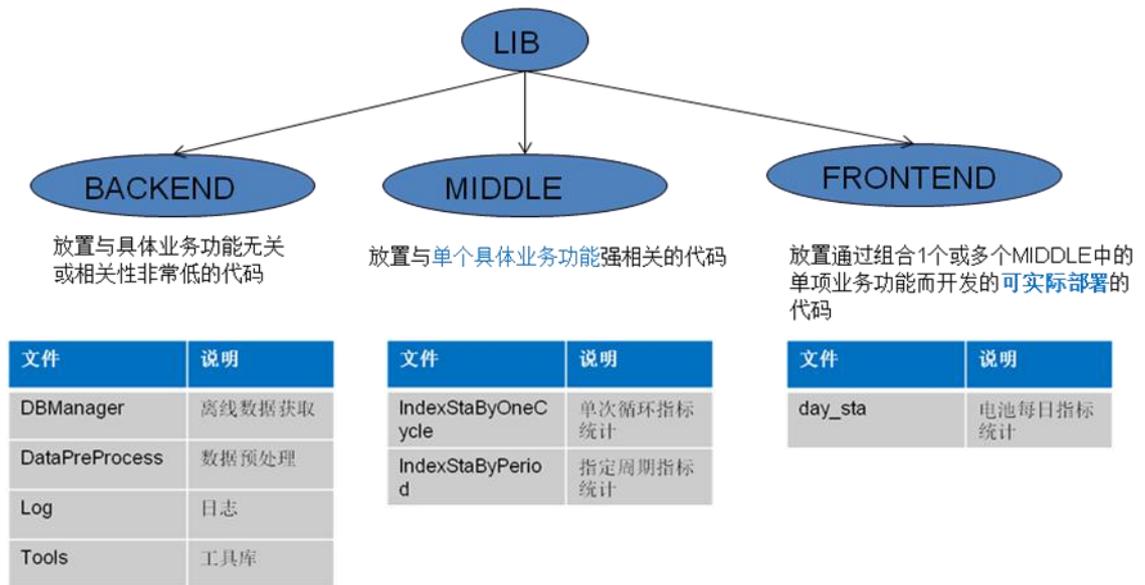


- 1) master:主分支, 稳定代码分支, 始终指向最新的已发布代码, 禁止直接在该

分支修改代码，通过 dev 分支合并的方式进行更新。

- 2) dev: 开发分支，在此分支上进行代码开发，及远程推送。
- 3) release*: 发布分支，每一个稳定版本的代码发布时通过 dev 分支创建一个发布分支

三、 当前平台库函数说明



具体的函数使用方式，可参照函数说明文件夹下的每个函数的 html 文档

四、 算法结果入库及部署

1. 算法结果入库流程
 - 1) 开发人员提供算法输出字段及数据类型
 - 2) 后台人员根据 1) 建表，并将表反馈给开发人员
 - 3) 开发人员在代码中嵌入写表语句，算法结果直接写入对应表中（后续可封装为工具函数）。
2. 算法部署流程以下：
 - 1) 开发人员将开发完成的代码上传至 git
 - 2) 开发人员提供算法运行周期（每天 6 点、持续）
 - 3) 后台人员在生产环境配置定时任务，启动对应的算法。
3. 确定输出字段及数据类型
4. 根据(1) 建表
5. 代码中嵌入写表语句，算法结果直接写入对应的表中