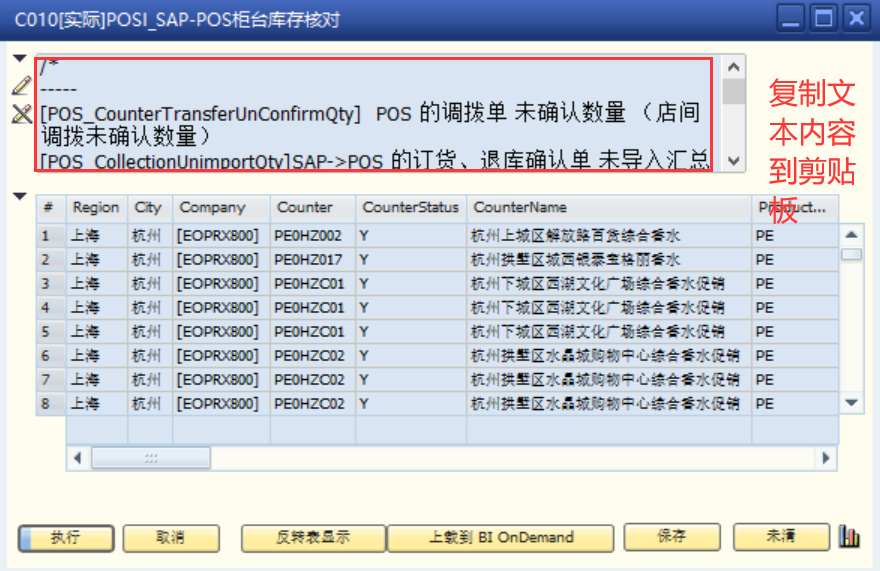
集团业务取数工具V1.0；

随着公司业务的增长，急需一个能够由非核心技术人员定义和参与的取数工具，在多个数据库中运行，将取出的数据合并，形成一个合并全集团的报表；

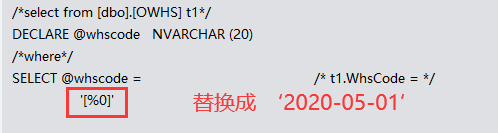
考虑到数据库、帐套、软件产品都是一样的，那取数逻辑只要来自事实表，理论上不会发生取数异常的情况；

1. 取数参考语句的取得方法；



找到报表的SQL语句；注意这里要把变量替换掉；

通常的变量包括 开始时间、结束时间、商品、组织维度等；



这里按需要调整就可以了。非常建议是带变量作为参数取数；否则因为取数的周期比较长，造成业务库锁死；

后期必要的时候，我们会引入SQL审阅流程；由业务部门提交SQL，技术团队确认没有异常方可进行取数；

数据库或者表锁死，表现为系统无法登入、数据不能刷新、卡顿、报表无法跑出来等异常，通常是由大规模的“慢SQL”运行造成的；

1. 定义目标表，通常建议用一个固定的名称，例如C010；根据业务习惯就可以了。但是建议统一管理，避免被其它数据取数覆盖；本工具会自动按照名称输出Excel文件；每次汇总后输出一个文件；
2. 任务；可以理解任务是若干个取数语句、输出对应表排序构成的作业集合；

Jobs = [

[SQL1,目标表1],

[SQL2,目标表2],

]

这里用到了List，每次运行按照不同的SQL排队取数，最后输出到Excel；

为了更容易理解，这里把例子再具体一点；

Jobs = [

[SQL\_081,A081],

[SQL\_031,A031],

]

也就是说，先运行SQL\_081这个语句，最后把结果输出到A081.xlsx；再按顺序执行SQL\_031这个语句，最后把结果输出到A031这个语句；

1. 错误处理机制；

按照此前的ETL开发，要求集团帐套全部在线才可以抽取；这有几个问题：

1. 远程数据库因为断电、维护等原因不可用；
2. 网络原因造成的不能访问；
3. 业务忙时造成表锁无法同时访问；

基于以上的要求，一旦发生这些问题，本工具会在日志写入“数据读取失败”等报错，然后继续面向下一个数据库执行；

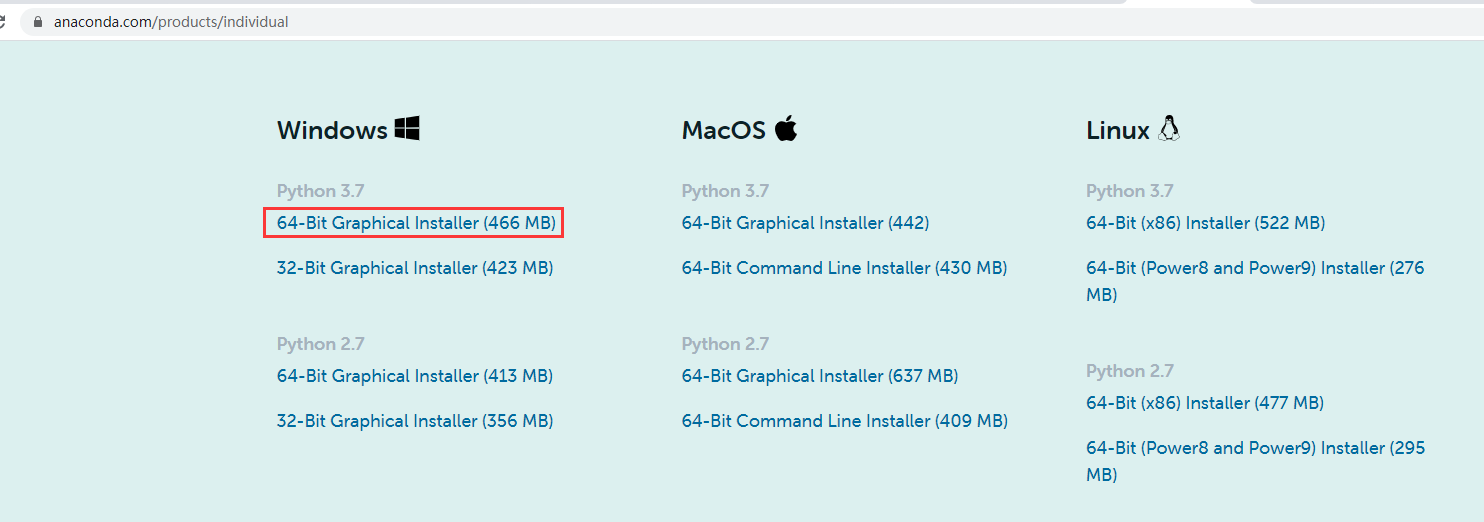
经过这样处理之后，在数据异常时能保证多数数据是按照要求取得的。不会造成全局取数失败；

环境安装；

代码运行在Python3.7+以下；

建议使用anaconda作为部署环境；Conda的包管理比较灵活，也可以继续使用pip作为包管理器使用；

<https://www.anaconda.com/products/individual>



<https://repo.anaconda.com/archive/Anaconda3-2020.02-Windows-x86_64.exe>

这个包里包括了Spider开发环境、Jupiter开发环境，比较适合数据挖掘使用；

安装使用默认就可以了。最好使用整体环境变量作为系统默认；（安装到环境变量时选择系统）；

代码发布：

1. 程序依赖包

pip install -r requirements.txt

或者自行安装sqlalchemy，pandas，pymssql等常用包；

命令使用方法pip install 包名；

1. 程序运行；

Python EtGroupETL.py

注意事项；

程序运行时请关闭excel文档；避免文档锁定，不能成功写入；