无法使用VBA来进行decode以及反序列化成数据结构，因此拆成两部分来做

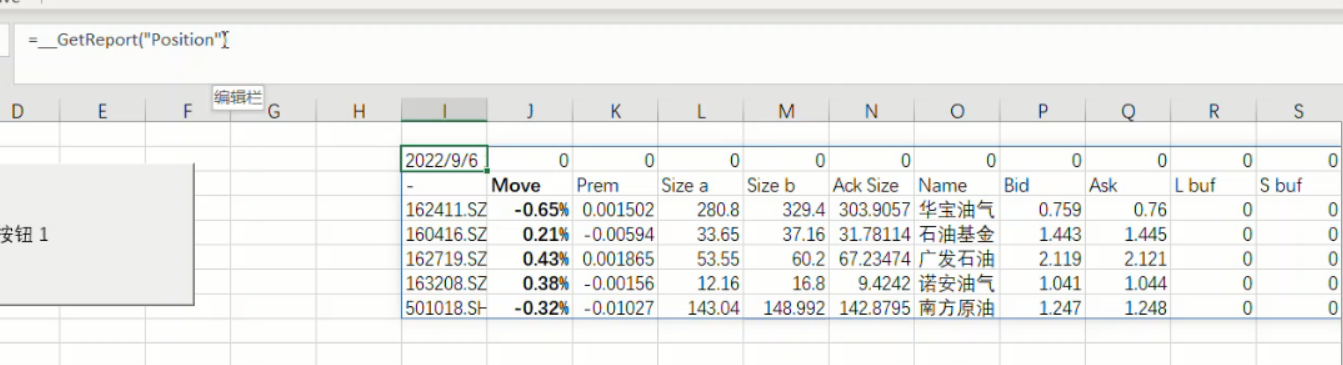
1. 在C#插件中添加Excel函数来读取数据库，将结果返回成基础类型 object [ , ] （二维表格）。该函数可以在Excel中直接使用，也可以通过VBA来调用；
2. 加载C#插件，然后在VBA中调用该C# Excel函数，通过Variant类型的变量来读取 object [ , ]类型的结果

* 在C#插件中创建一个Excel函数：object \_\_GetReport(string reportName, string columnsObj)

（<https://github.com/InterestingTinyProjects/ExcelPublisher_Public/blob/master/src/OpenAPI.Cms.TestTools/Publisher.ExcelAddin/Publisher.ExcelAddin/UI/Functions.cs> ）

* 该函数从数据库中读取某个report的最新值，然后根据columnsObj返回需要列，所有列名以逗号分割
* 该函数返回 Object[ , ]数据来描述表格，其中第一行第一列为DataTimeTag， 第二以下行为Pubilsh的Report的内容

例：需要从数据库获取Position report，显示所有列，则调用 \_\_GetReport(“Positions”)，返回如下

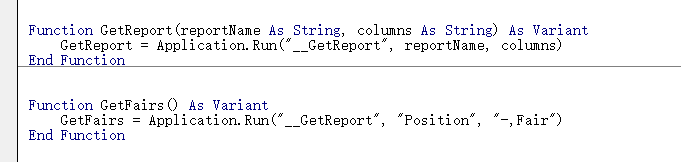


需要从数据库获取Position report, 显示Code，Move，则调用 \_\_GetReport(“Positions”， “-, Move, Size b”)

图形用户界面, 应用程序, 表格, Excel

描述已自动生成

* 在VBA中创建 Wrapper Function，方便 VBA对C#插件中的调用，具体可参考ReadFairsVBA.xlsb（<https://github.com/InterestingTinyProjects/ExcelPublisher_Public/tree/master/doc/20220912>）
* VBA中可采用 Application,Run(<函数名>，<参数1>, <参数2>, …) 的方式来调用C#插件中的函数
* 因此添加 VBA GetReport Function来调用C#中的\_\_GetReport函数
* 根据需要可以添加不带参数的GetFairs() Function来简化调用，函数体中写好Fairs对应的Report Name以及具体需要返回的column



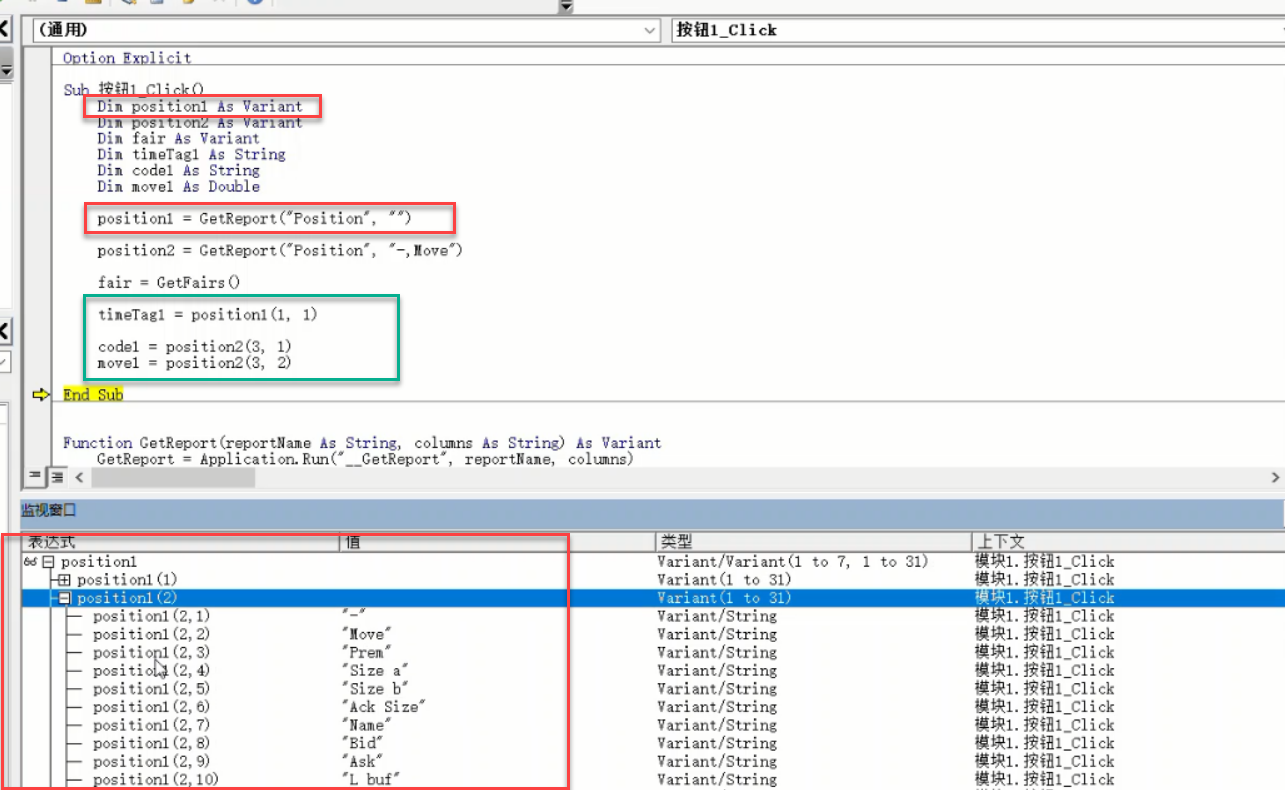
* 在VBA中调用 Wrapper Function，具体可参考ReadFairsVBA.xlsb

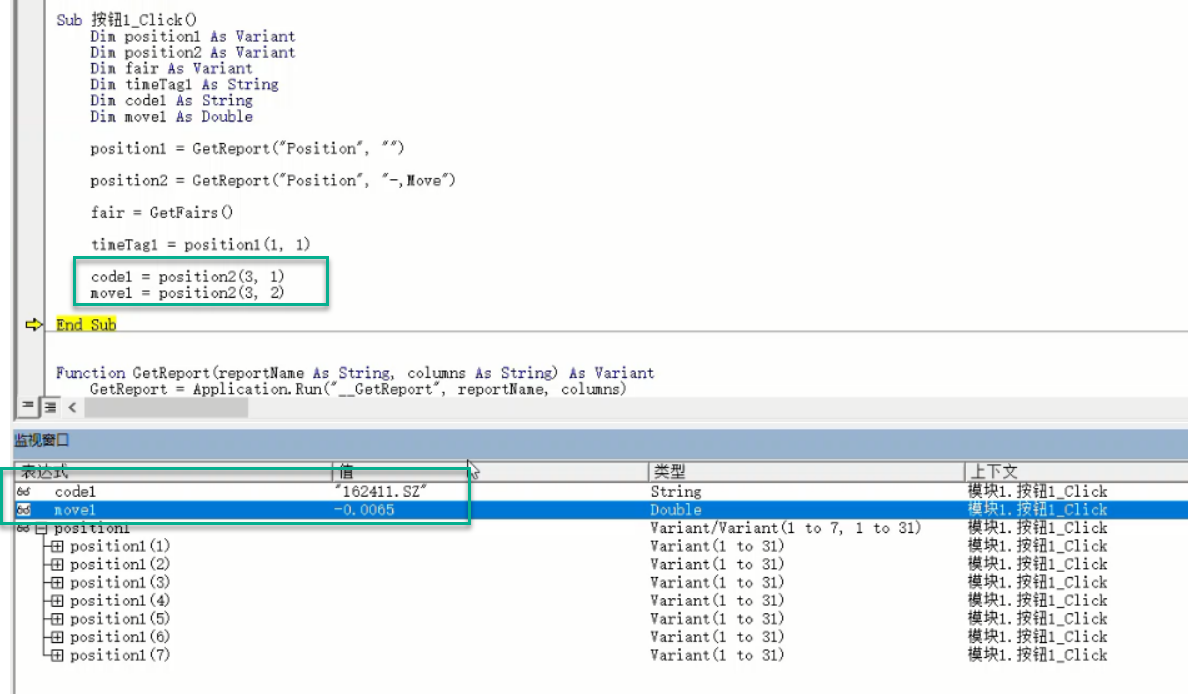
（<https://github.com/InterestingTinyProjects/ExcelPublisher_Public/tree/master/doc/20220912>）

可以在VBA中定义一个Variant类型的变量来存储GetReport返回的 Object[ , ]，然后按数组操作读取返回值即可；

例如 timeTag的值在 positions1(1, 1)；

以下Position Report的列名从行(2)开始，因此可以从 positions1(2,1)开始获得列名；从posiitons1(3,1)开始获得第一个持仓的代码，





* 异常处理

Excel函数的异常处理与Publish线程的处理不同没有写入Log，而是会将异常信息直接作为string返回。因此在Excel里直接调用并发生异常时，Excel的Cell会直接显示异常信息；在VBA调用函数并发生异常时，Variant变量获得的是异常信息的String而不是 Object[ , ]

