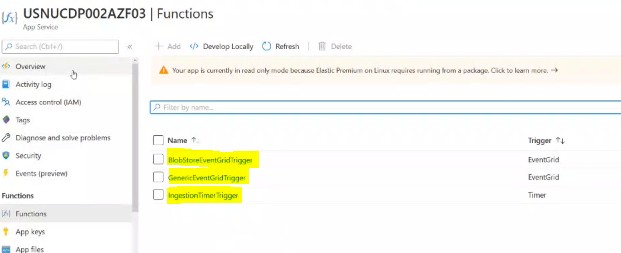
Entire orchestration layer that we have for Fusion data Fabric goes through basically via azure functions .We started dealing with three kinds of Azure Functions



Ingestion Timer Trigger :Triggering function for data ingestions and it is scheduled to run for every 1 minute ( 5min as of now it is not confirmed as per the video)

Blob Store event Grid Trigger – Triggering function for Blob Store

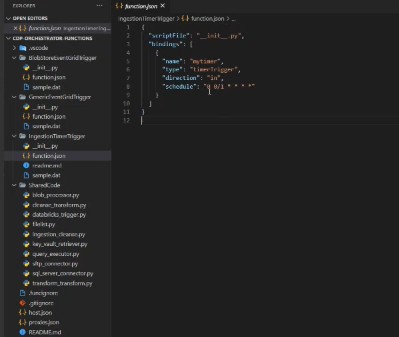
Generic Event Grid Trigger – Triggering function for ADLS

Whenever a file comes in to the ADLS location or when we write any data in to the ADLS location the event gets captured which is similar to the S3event trigger -AWS. There is another trigger function Event hub trigger which is not showed here

Event hub trigger – this trigger is used to essentially read the messages from eventhub as they come out to the event hub

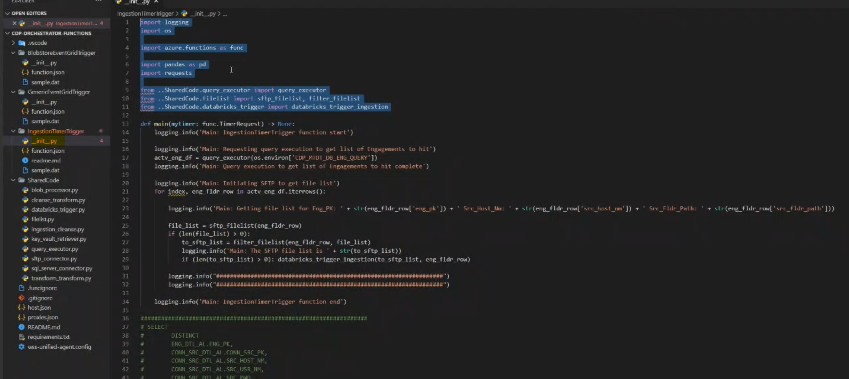
**Ingestion Time trigger**

When the above function is opened. We can see at one part there is a json code (pasted below screenshots- not clear) which shows that its been scheduled to run for every minute as mentioned in the code for a regular cron expression we will be having 5 parameters in it but here we have six parameters which needs to be taken care

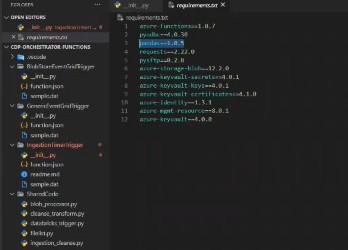


The trigger execution depends on how quickly the SFTP connection gets in and probably it takes 5 seconds to somewhere around 15-20.the code is written in python

**The actual code -\_\_init\_\_py**

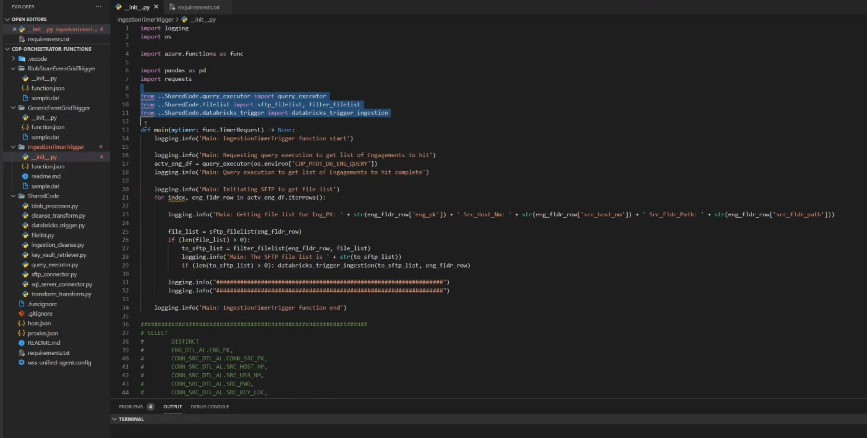


The required libraries are added in a requirements.txt and called in the code

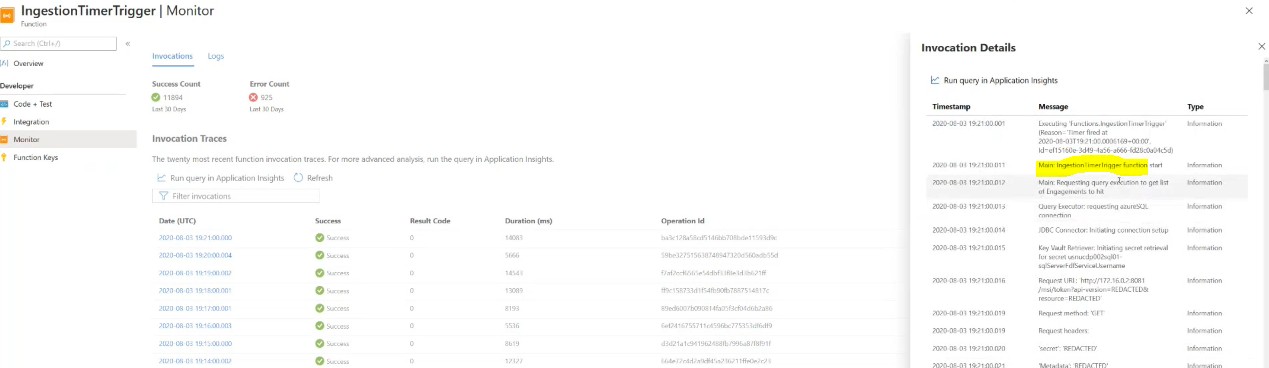


Any kind of new libraries for azure functions are added in this text only .

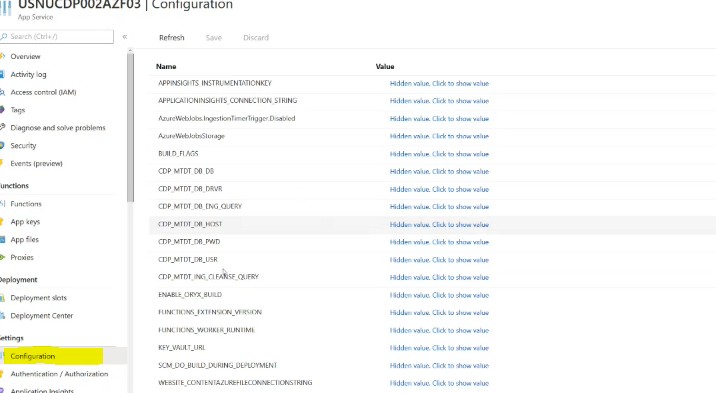
In the code after line 13 always gets poplulated by default line 9,10,11 are custom codes being written



Logs customized on the above code can be monitored on the Azure window below



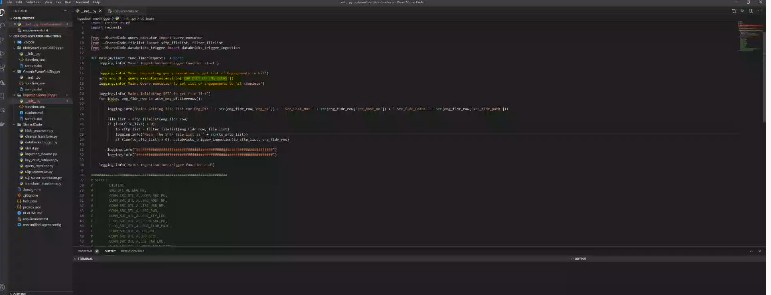
IN the configuration tab you can see that the configutaion partrametrs used in the code



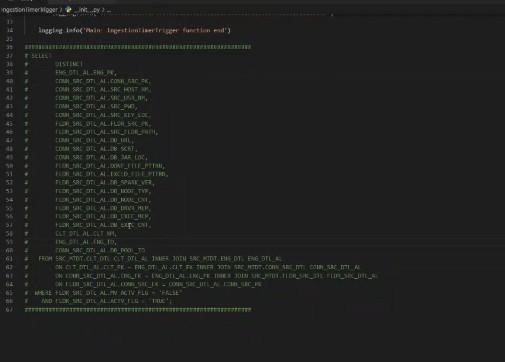
In this window you can find the sql codes which we used in the code



Here we have hardcoded the sql codes in python and called the same in the configuration parameters



In future if the query needs to be changed we can change in the configuration parameter without touching the code. Here is the query



Flag concept

Active Flag(False) - out of 10 clients if 2 clients opted out then the active flag is set to true for those clients and files wont e processed from their corresponding location

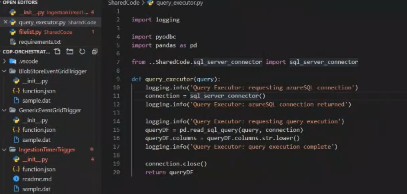
which is set to false for indicating that file is not processed anymore and moreover We are not deleting the data due to data retention period . Hence when the value is set to false that particulat SFTP site will never get picked up in the ingestion time trigger

Move Active Flag(False) - it is going to pull .if the value is false it knows that there is no existing job running and for example if four files are there it is going tospin a database cluster and makes the flag as true .So we are having this flag as true for avoiding the already processed files

Query is running via DB Visualizer

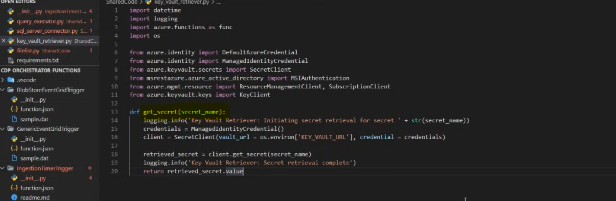
* Multiple Engagements for a single client and mutliple SFTP locations for multiple engagements with corresponding multiple folders from where files are pulled
* Azure Key vault is used for storing the secret prametrs like userid and password

There is a query executor function used in this code

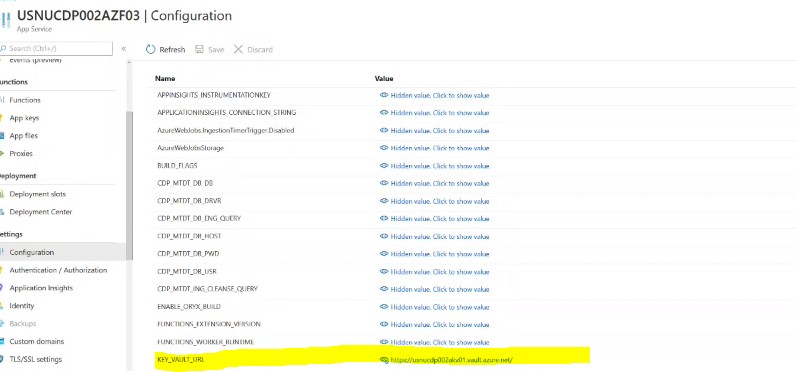


Which is connected via PYODBC where as system configurations are maintained in the code via OS Module .

A function called get\_secret(secret\_name) isused for getting the information . here using OS module we are calling the Keyvault and the DNS name as well

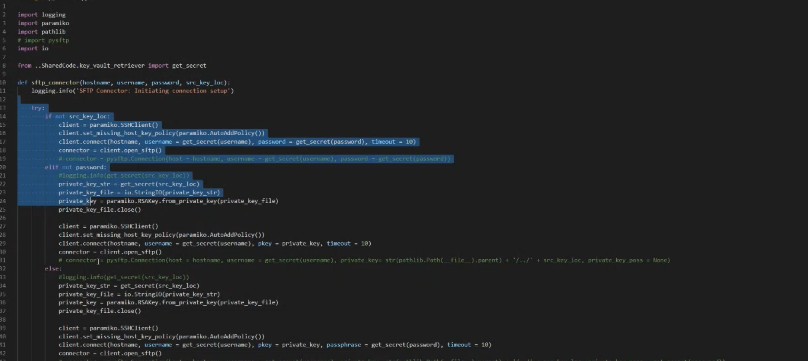


And inside the key vault we are hardcoding

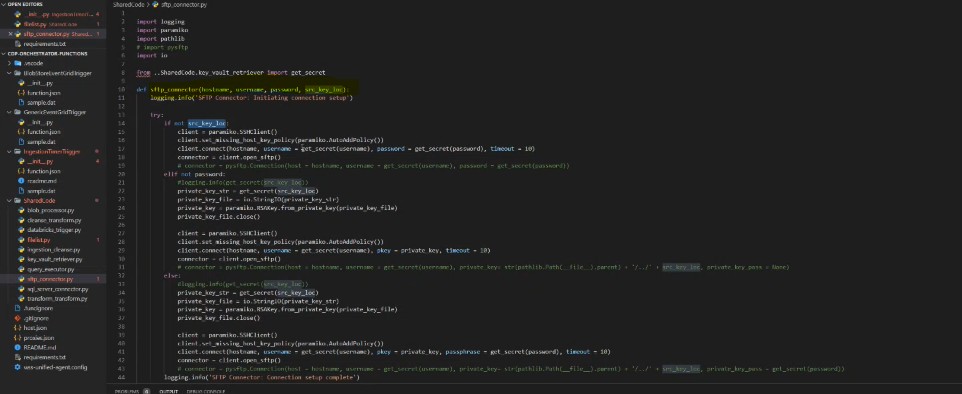


It will be different for different resources

A function named as SFTP\_CONNECTOR is used

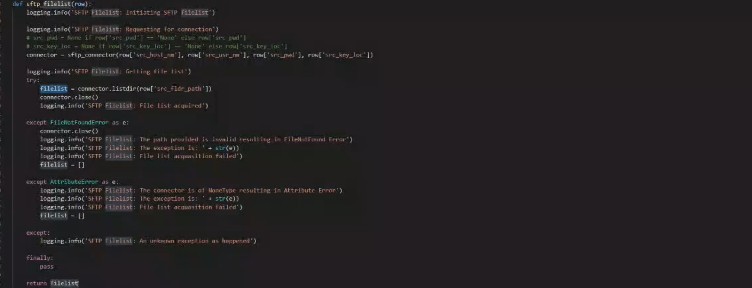


In this Function SFTP connection is set with the FTP connection we are passing and basically for paremeters which is the hostname ,username , password and the key location and as these all are coming from the metadata aside from the keys .Here we are using a paramiko library for establishing SFTP connection



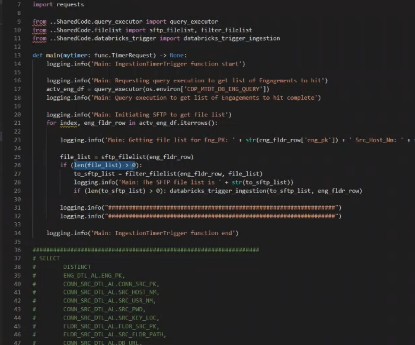
In the aabove code we are writing some conditions like if you don’t have a key use a username and password and if not use a username and private key based . those level conditions are written on the above code. And in the end we have some exception handling if the connection is not successful

Another function named as SFTP\_FILELIST



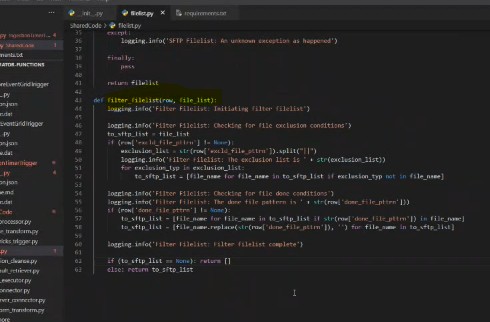
The above function gets triggered when SFTP connection is successful for pulling the files which we need to

Line no-23



if condition - if the list is greater than zero there are files and if it is not there are no files

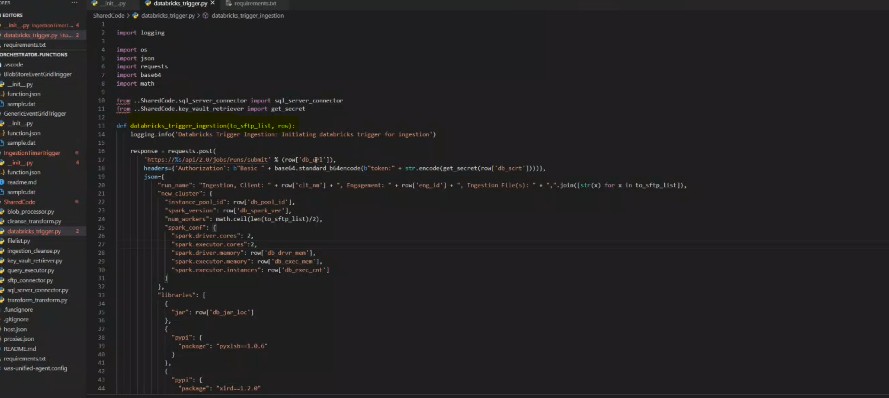
and the next function we have to see is **filter\_filelist**



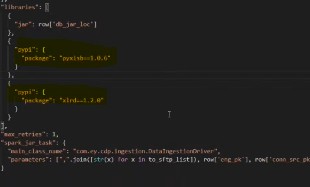
It basically filters files , if it is finding particular string in that filename it is omiiting that file . if the file pattern is null then it is good to go . It waits for a dotfile for any actual file . if it is received it is passed to the main function. As soon as the entry is happen to the metadata process starts .So once we are done with this the next thing we need to do is the databricks trigger

Database Time Trigger

Function to create a cluster and cluster configurations like no of vms and those informations are mentioned in this function.it includes spark version and other related information



Ideally standard packages like XLSP and XLRD packages are used which handles basically your Excel files and the packages are hardcoded



And finally the respective parameters are passed to get the metadata from databricks sideand this part is called runsubmit and status codes were assigned for successful connections.