Wednesday(19th)

* Daily objs
  + Figure out toast issue
  + Any incoming problems
* Jenny
  + Bigcommerce
    - Jenny asked if bigcommerce connection is possible
    - It is possible but it is in private preview which means there cannot be a connector token generated for it
    - We also checked and we don’t have access to it so we requested access to bigcommerce connector through fivetran
* Toast
  + Lambda
    - Direct connection through lambda did not succeed
      * We ran into a configuration problem
        + Aws configure command requires user input
      * Found a solution using user profiles instead
        + This eliminates the need for user input
        + The profiles on the ec2 worked
        + But when trying to call the profiles from lambda it didn’t recognize the profiles
    - Decided to utilize local python functions
      * Thought this would avoid all the issues of profiles because the functions would be ran locally so lambda compute would not run into a profile conflict
      * Used several libraries
        + Argparse

Used to pass command line arguments to a python script

* + - * + Os

Used to interact with the computer infrastructure itself

Such as command line

* + - * + Boto3

Used to interact with aws api

* + - Wrote the local functions with being able to pass the variables in line
      * This is important for the automation process
    - Example function
* import argparse
* import boto3
* import os
* parser = argparse.ArgumentParser()
* parser.add\_argument("directory\_name")
* #parser.add\_argument("customer")
* #parser.add\_argument()
* #parser.add\_argument()
* #os.system('cmd /c' "mkdir practice")
* #os.system('cmd /c' "cd practice")
* args = parser.parse\_args()
* #print(type(args.directory\_name))
* os.system(f'cmd /c "aws s3api create-bucket --bucket {args.directory\_name.lower()} --region us-east-1 --profile bearcog"')
* #os.system(f'cmd /k "cd {args.directory\_name} && aws s3 sync s3://restaurant-exports/{args.customer}/ . --profile {args.customer}"')
* #os.system('cmd /k' "aws s3 sync s3://restaurant-exports/TheSmokehouseExportUser2/ . --profile toast")
* #os.system('cmd /c' "aws s3 sync s3://restaurant-exports/TheSmokehouseExportUser2/ . --profile toast")
  + - * This function above is used to create the s3 bucket that we will use to store the data
    - Connected to the server via rdp
      * Used the extra options to access my local c drive
      * Put all of the python functions into a local folder
      * Access lambda
      * Attempted to run the functions and received the error message in the logs
        + python : The term 'python' is not recognized as the name of a cmdlet, function, script file, or operable program.
      * Upon research I learned that sometimes powershell scripts use py instead of python for python engine
        + Tried it with the py instead of python
        + This is the error message that was thrown

Can't find a default Python.

* + - * Then tried it through ssm directly and it did not work either
    - Looked further into the problem and it seemed to be a problem with the shebang in one of the python setup scripts
    - It has a linux path instead of a windows one
    - Going to switch over to ubuntu server and try again