**Python Scripts**

Need to know how to write Error handlers in python.

How to look up for variables in a python scripts.

\n = Start a new line.

\\* or \# = This means python will write the special character(\* or #) in the script.

Name = ‘Adele’ => means it is assigning the value Adele to Name.

Concatenation = Pending a string to another one like Name + ‘Like Cake’.

Functions = Block of code that can do specific operation ( modification or get info).

Example Functions: abs(); pow(,); max(,); min(,); round(,); floor(); ceil();

Print(Name.upper()) => so the function upper turn the value ‘Adele’ into Upper Case.

Print(Name.upper().isupper()) => so turn The value into Upper Case and check if the value in Name is upper Case.

Print(len(name)) = > give us the number of characters in Name = 5.

Print(Name[0]) => Here we specify that we want the character of index = 0 in the string Name which is = A, so A= 0, D=1, E=2, L=3, E=4. Or we can do “ print(name.index(‘L’)) ” which will give us = 3.

Everytime you wanna print a number next to a string, you have to convert the number into a string.

Num= 5; print( str (Num) + “oranges sont sur la table”).

How to import function libraries: “ from math import \* ”.

Data Type: Int(all numbers);

How to train a model in python.

import rdkit  
from rdkit import Chem  
m = Chem.MolFromSmiles('CC(=0)NCCC1=CNc2c1cc(0C)cc2')  
print(m)

from rdkit import Chem  
from rdkit.Chem import Draw  
from rdkit.Chem.Draw import IPythonConsole  
from rdkit.Chem import Descriptors  
from rdkit.Chem import AllChem  
from rdkit import DataStructs  
import numpy as np  
  
  
  
m = Chem.MolFromSmiles('COc1cc(C=O)ccc1O')  
print(m)

import pandas as pd  
from rdkit import Chem  
from rdkit.Chem import Draw  
  
from rdkit.Chem import Descriptors  
from rdkit.Chem import AllChem  
from rdkit import DataStructs  
import numpy as np  
  
  
  
m = Chem.MolFromSmiles('COc1cc(C=O)ccc1O')  
print(m)  
  
  
  
# How to read a csv smile file  
# data = pd.read\_csv('./location of the file', index\_col=0) data.head()  
  
#we define a variable to take individual smile string 1 by 1; data defined at the top; Smiles\_std is the column in our table when smiles are stored; and index 0  
# my\_one\_smile\_string = data['SMILE\_STD'][0]  
  
#Transform a smile to a mol; our one smile that we save, we redifining it and transform it into a mol from a smile  
#my\_one\_smile\_string = chem.MolFromSmiles(my\_one\_smile\_string, sanitizer=True)  
  
#so we will see what the mol object looks like  
#print(my\_one\_mol\_object)  
  
#Transform mols to smiles; so we can tranform what we saved into our variable into smiles  
#chem.MolToSmiles(my\_one\_mol\_object)  
  
  
  
#With Panda DataFrame, you can just write it like this: so we copy the first 40 entries of the data table  
#data\_small = data.head(40).copy()  
#print('data\_small.head()')  
  
  
#Here we are creating a new table with the first column having smiles code and 2nd col with mol object  
#from rdkit.chem import PandasTools  
#pandasTools.AddMoleculeCOlumnToFrame(data\_small, smilesCol = 'Smile\_STD', molCol='RoMol')  
  
  
#so we can dra mol files to grid image to see them; here we will use the data from our mol object col RoMol  
#from rdkit.chem import Draw  
#Draw.MolsToGridImage(list(data\_small.RoMol), useSVG=True)