1. **Extraction of ADMET data from RD Kit.**

**DEFINE FUNCTION calculate\_properties(smiles):**

SET mol TO Chem.MolFromSmiles(smiles)

IF mol is None:

RETURN None

**# calculate ADME/T properties**

SET mw TO Descriptors.MolWt(mol)

SET logp TO Descriptors.MolLogP(mol)

SET tpsa TO Descriptors.TPSA(mol)

SET hbd TO Descriptors.NumHDonors(mol)

SET hba TO Descriptors.NumHAcceptors(mol)

SET rotb TO Descriptors.NumRotatableBonds(mol)

SET arom TO Descriptors.NumAromaticRings(mol)

SET # ring TO Descriptors.NumRings(mol)

**# additional ADME/T properties**

SET gi\_absorption TO 'Unknown'

IF logp >= 0.4 and logp <= 5.6 and tpsa <= 140:

SET gi\_absorption TO 'High'

ELSEIF logp >= -2 and logp <= 6 and tpsa <= 180:

SET gi\_absorption TO 'Moderate'

ELSE:

SET gi\_absorption TO 'Low'

SET bbb\_permeant TO 'Unknown'

IF logp >= -0.4 and logp <= 2.5 and hbd <= 2:

SET bbb\_permeant TO 'Yes'

ELSE:

SET bbb\_permeant TO 'No'

SET p\_gp\_substrate TO 'Unknown'

IF logp <= 3 and hbd <= 5 and hba <= 10:

SET p\_gp\_substrate TO 'Yes'

ELSE:

SET p\_gp\_substrate TO 'No'

SET cyp1a2\_inhibitor TO 'No'

SET cyp2c19\_inhibitor TO 'No'

SET cyp2c9\_inhibitor TO 'No'

SET cyp2d6\_inhibitor TO 'No'

SET cyp3a4\_inhibitor TO 'No'

IF logp <= 5 and tpsa <= 120:

SET cyp1a2\_inhibitor TO 'Yes'

SET cyp2c19\_inhibitor TO 'Yes'

SET cyp2c9\_inhibitor TO 'Yes'

SET cyp2d6\_inhibitor TO 'Yes'

SET cyp3a4\_inhibitor TO 'Yes'

SET lipinski TO 0

IF mw <= 500 and logp <= 5 and hba <= 10 and hbd <= 5:

SET lipinski TO float(1)

SET ghose TO 0

IF mw >= 160 and mw <= 480 and logp >= -0.4 and logp <= 5.6 and hba >= 0 and hba <= 10 and hbd >= 0 and hbd <= 5:

SET ghose TO float(1)

SET veber TO 0

IF rotb <= 10 and tpsa <= 140 and logp <= 5:

SET veber TO float(1)

SET result TO {

'Molecular weight': mw,

'LogP': logp,

'TPSA': tpsa,

'H-bond donors': hbd,

'H-bond acceptors': hba,

'Rotatable bonds': rotb,

'Aromatic rings': arom,

# 'Rings': ring,

'GI absorption': gi\_absorption,

'BBB permeant': bbb\_permeant,

'P-gp substrate': p\_gp\_substrate,

'CYP1A2 inhibitor': cyp1a2\_inhibitor,

'CYP2C19 inhibitor': cyp2c19\_inhibitor,

'CYP2C9 inhibitor': cyp2c9\_inhibitor,

'CYP2D6 inhibitor': cyp2d6\_inhibitor,

'CYP3A4 inhibitor': cyp3a4\_inhibitor,

'Lipinski': lipinski,

'Ghose': ghose,

'Veber': veber

}

RETURN result

**Appending the properties to a dataframe**

SET admet\_prop TO []

SET nan\_smiles TO []

FOR smile IN df['SMILES']:

SET res TO calculate\_properties(smile)

IF res :

SET res['SMILES'] TO smile

admet\_prop.append(res)

ELSE:

nan\_smiles.append(smile)

1. **Extraction of compound information by Pubchem.Py**

**DEFINE FUNCTION smile\_to\_iupac(smile):**

TRY:

compounds =pubchempy.get\_compounds(smile, namespace='smiles')

SET match TO compounds[0].to\_dict(properties=['cid','canonical\_smiles','exact\_mass',

'h\_bond\_acceptor\_count','h\_bond\_donor\_count','iupac\_name','molecular\_formula','inchi'])

RETURN match

except PubChemHTTPError as e:

RETURN None

# Appending the properties to a dataframe

SET pub\_prop TO {}

FOR val IN df['SMILES']:

IF val NOT IN pub\_prop:

SET prop TO smile\_to\_iupac(val)

IF prop:

pub\_prop[val]= prop

with open('result.json', 'w') as fp:

json.dump(pub\_prop, fp)

**Save the new dataframe with all properties as a .csv file**

**Python code for extraction of compound information by RDKit and Pubchempy**