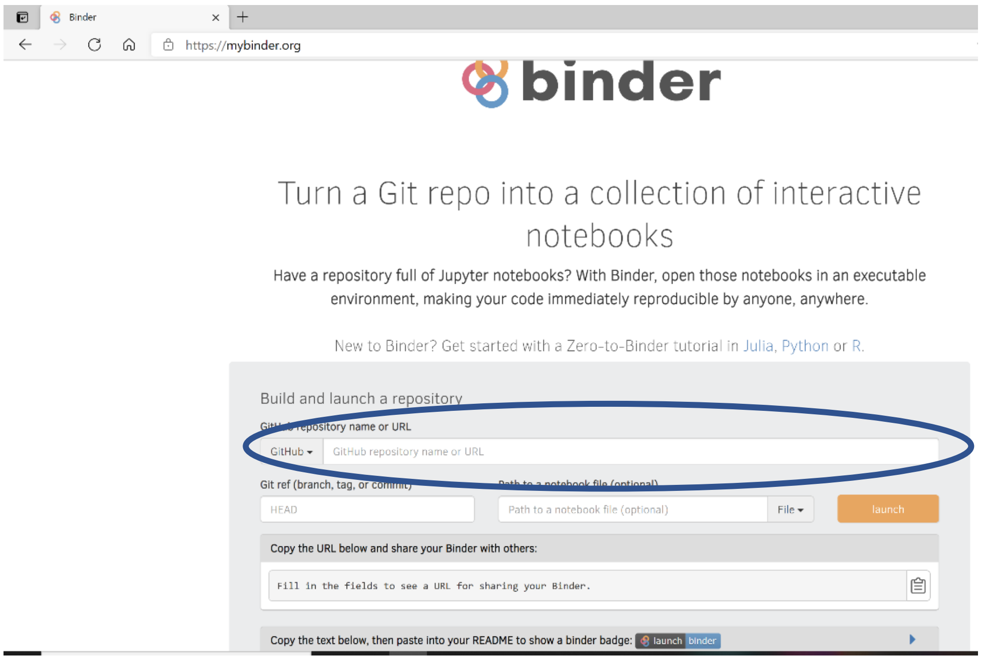
1) Open browser and type mybinder.org

SHOULD HAVE A SCREEN LIKE THE ONE BELOW



Turn a Git repo into a collection of interactive notebooks

Have a repository full of Jupyter notebooks? With Binder, open those notebooks in an executable environment, making your code immediately reproducible by anyone, anywhere.

New to Binder? Get started with a Zero-to-Binder tutorial in Julia, Python or R.

Build and launch a repository

GitHub repository name or URL: GitHub ▾ | GitHub repository name or URL

Git ref (branch, tag, or commit): HEAD

Path to a notebook file (optional): Path to a notebook file (optional) | File ▾ | launch

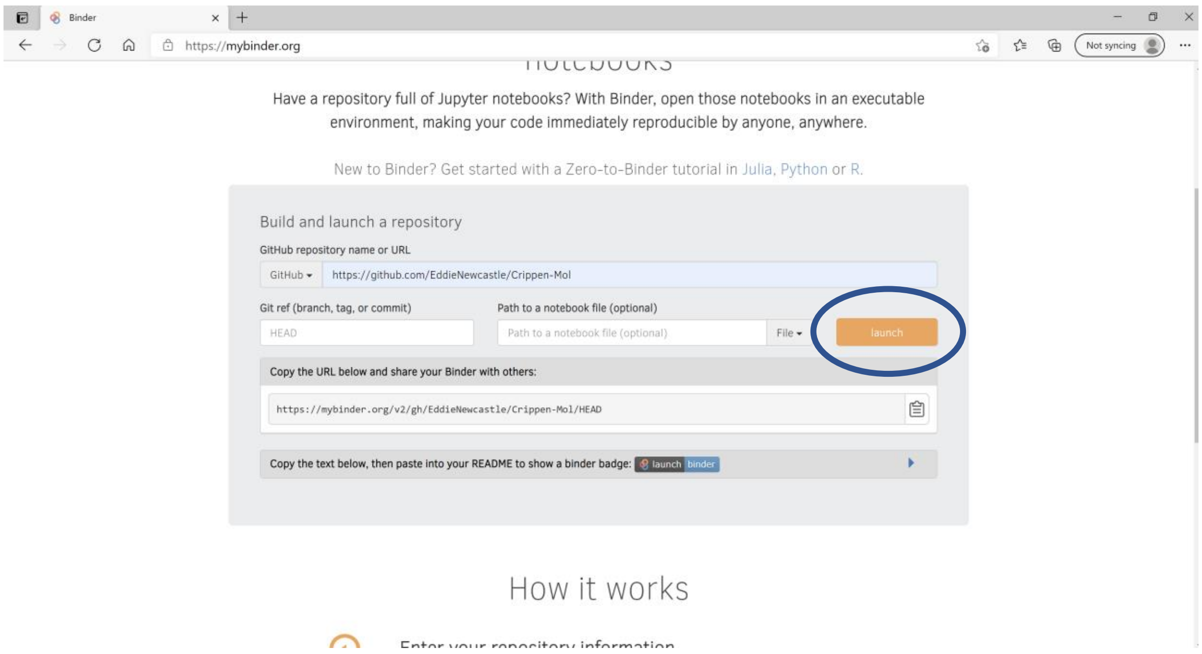
Copy the URL below and share your Binder with others:

```
Fill in the fields to see a URL for sharing your Binder.
```

Copy the text below, then paste into your README to show a binder badge: launch binder

2) In the "GitHub repository name or URL" input https://github.com/EddieNewcastle/Crippen-Mol

Then click launch



Have a repository full of Jupyter notebooks? With Binder, open those notebooks in an executable environment, making your code immediately reproducible by anyone, anywhere.

New to Binder? Get started with a Zero-to-Binder tutorial in Julia, Python or R.

Build and launch a repository

GitHub repository name or URL: GitHub ▾ | https://github.com/EddieNewcastle/Crippen-Mol

Git ref (branch, tag, or commit): HEAD

Path to a notebook file (optional): Path to a notebook file (optional) | File ▾ | launch

Copy the URL below and share your Binder with others:

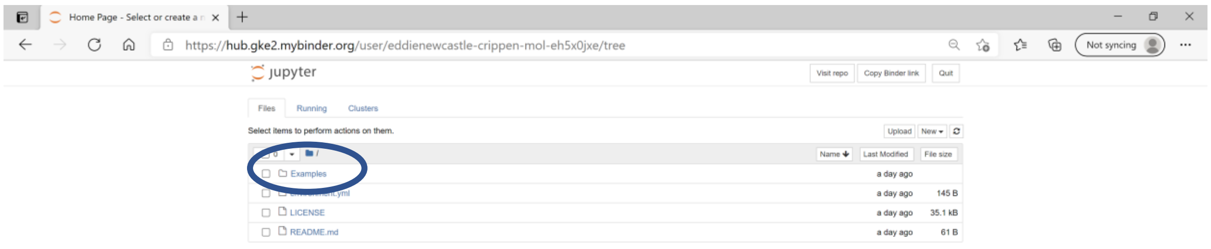
```
https://mybinder.org/v2/gh/EddieNewcastle/Crippen-Mol/HEAD
```

Copy the text below, then paste into your README to show a binder badge: launch binder

How it works

SHOULD TAKE SOME TIME TO LOAD BUT BE PATIENT

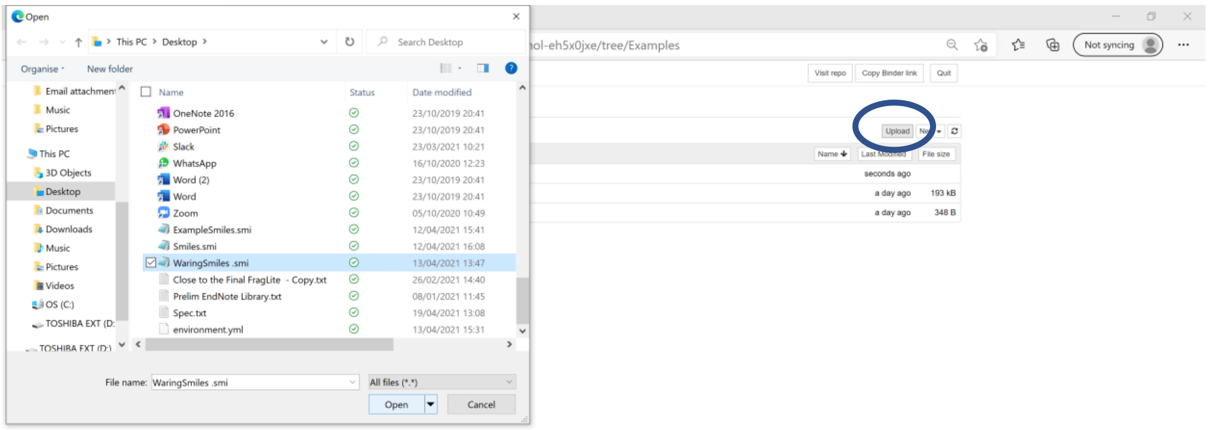
3) Click on Example file


Home Page - Select or create a
https://hub.gke2.mybinder.org/user/eddienewcastle-crippen-mol-eh5x0jxe/tree

Select items to perform actions on them.

	Name	Last Modified	File size
	Examples	a day ago	
	environment.yml	a day ago	145 B
	LICENSE	a day ago	35.1 kB
	README.md	a day ago	61 B

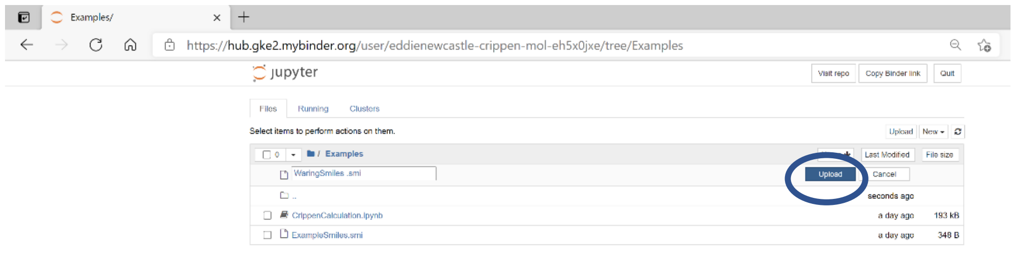
4) In the Example file – Click upload and then select the relevant SMILES file and click open.



Name	Status	Date modified
OneNote 2016	✓	23/10/2019 20:41
PowerPoint	✓	23/10/2019 20:41
Slack	✓	23/03/2021 10:21
WhatsApp	✓	16/10/2020 12:23
Word (2)	✓	23/10/2019 20:41
Word	✓	23/10/2019 20:41
Zoom	✓	05/10/2020 10:49
ExampleSmiles.smi	✓	12/04/2021 15:41
Smiles.smi	✓	12/04/2021 16:08
WaringSmiles .smi	✓	13/04/2021 13:47
Close to the Final FragLite - Copy.txt	✓	26/02/2021 14:40
Prelim EndNote Library.txt	✓	08/01/2021 11:45
Spec.txt	✓	19/04/2021 13:08
environment.yml	✓	13/04/2021 15:31

File name: WaringSmiles .smi   All files (*.*)
Open   Cancel

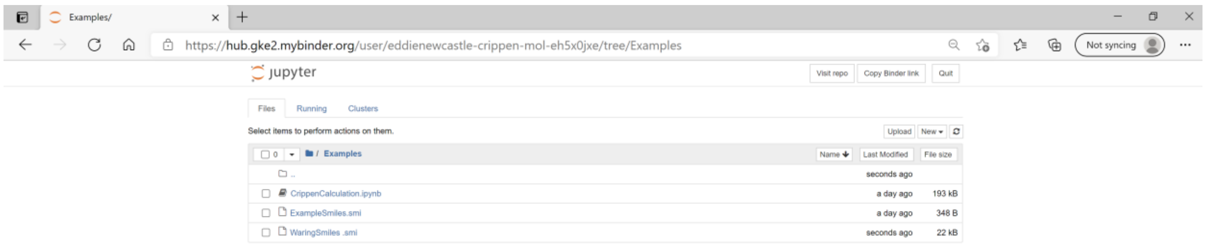
5) Click upload.


Examples/
https://hub.gke2.mybinder.org/user/eddienewcastle-crippen-mol-eh5x0jxe/tree/Examples

Select items to perform actions on them.

	Name	Last Modified	File size
	WaringSmiles .smi		
	..	seconds ago	
	CrippenCalculation.ipynb	a day ago	193 kB
	ExampleSmiles.smi	a day ago	348 B

SCREEN SHOULD LOOK LIKE THIS

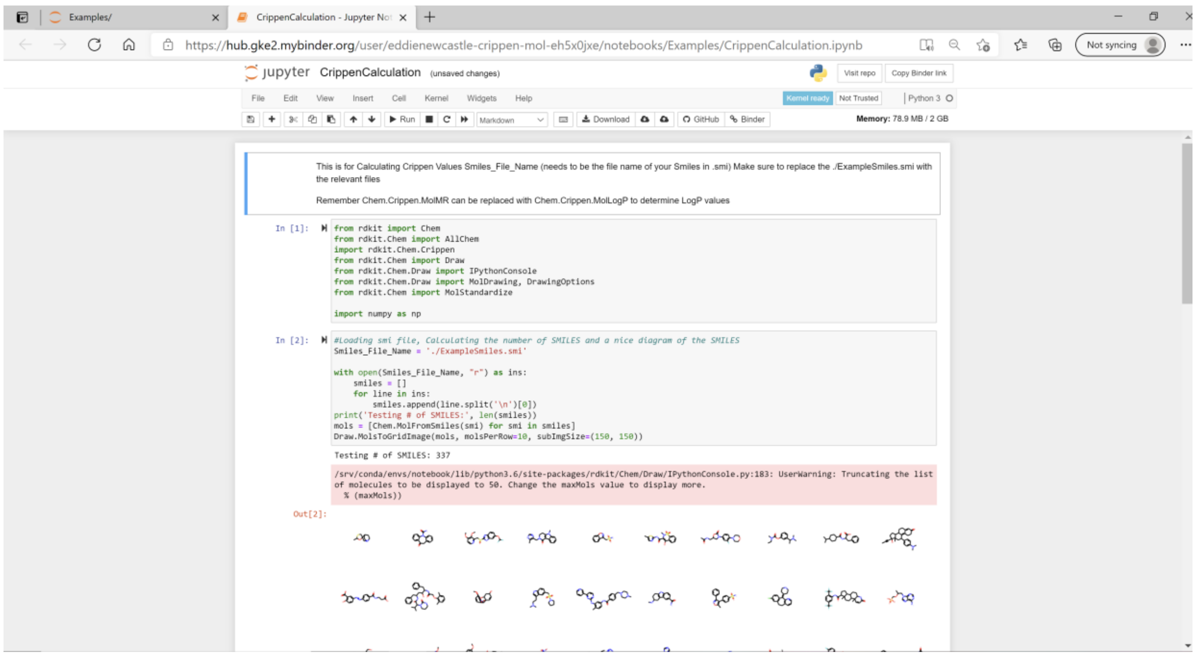

Examples/
https://hub.gke2.mybinder.org/user/eddienewcastle-crippen-mol-eh5x0jxe/tree/Examples

Select items to perform actions on them.

	Name	Last Modified	File size
	..	seconds ago	
	CrippenCalculation.ipynb	a day ago	193 kB
	ExampleSmiles.smi	a day ago	348 B
	WaringSmiles .smi	seconds ago	22 kB

6) Now click on CrippenCalculation.ipynb

Should look like this


Examples/
CrippenCalculation - Jupyter Not
https://hub.gke2.mybinder.org/user/eddienewcastle-crippen-mol-eh5x0jxe/notebooks/Examples/CrippenCalculation.ipynb
Not syncing
jupyter CrippenCalculation (unsaved changes)
Visit repo
Copy Binder link
File Edit View Insert Cell Kernel Widgets Help
Kernel ready
Not Trusted
Python 3
Run
Markdown
Download
GitHub
Binder
Memory: 78.9 MB / 2 GB

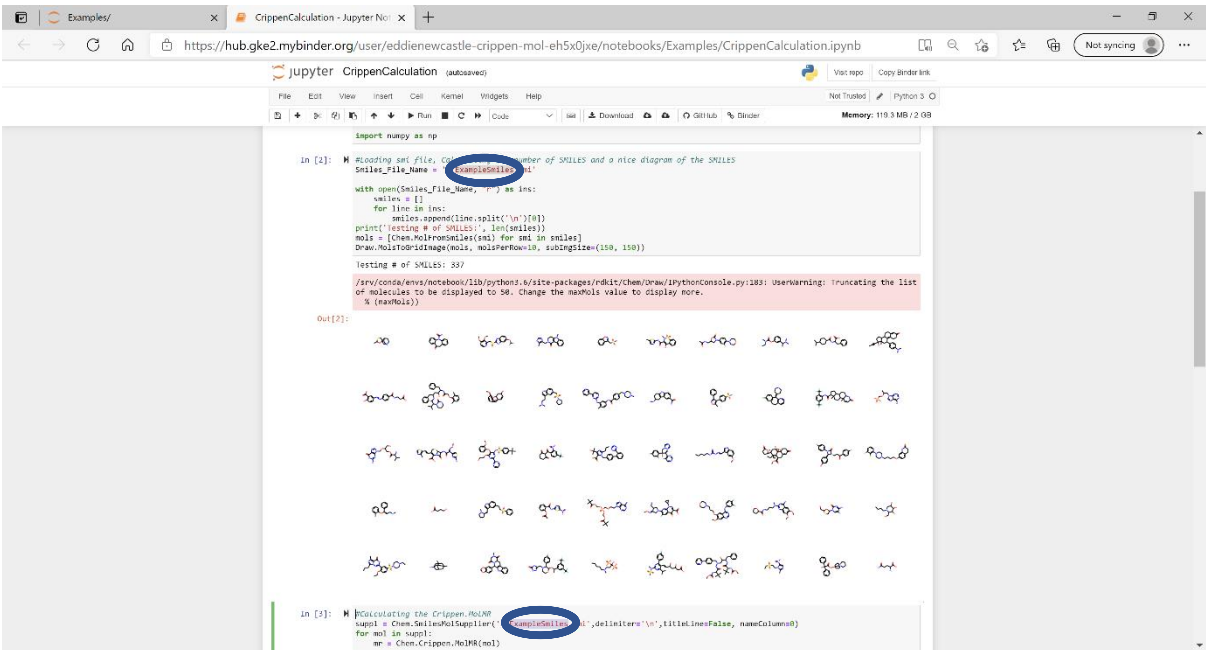
This is for Calculating Crippen Values Smiles_File_Name (needs to be the file name of your Smiles in .smi) Make sure to replace the ./ExampleSmiles.smi with the relevant files

Remember Chem.Crippen.MolMR can be replaced with Chem.Crippen.MolLogP to determine LogP values

```
In [1]:  from rdkit import Chem
         from rdkit.Chem import AllChem
         import rdkit.Chem.Crippen
         from rdkit.Chem import Draw
         from rdkit.Chem.Draw import IPythonConsole
         from rdkit.Chem.Draw import MolDrawing, DrawingOptions
         from rdkit.Chem import MolStandardize

         import numpy as np

In [2]:  #Loading smi file, Calculating the number of SMILES and a nice diagram of the SMILES
         Smiles_File_Name = './ExampleSmiles.smi'

         with open(Smiles_File_Name, "r") as ins:
             smiles = []
             for line in ins:
                 smiles.append(line.split('\n')[0])
         print('Testing # of SMILES:', len(smiles))
         mols = [Chem.MolFromSmiles(smi) for smi in smiles]
         Draw.MolsToGridImage(mols, molsPerRow=10, subImgSize=(150, 150))

         Testing # of SMILES: 337

         /srv/conda/envs/notebook/lib/python3.6/site-packages/rdkit/Chem/Draw/IPythonConsole.py:183: UserWarning: Truncating the list
         of molecules to be displayed to 50. Change the maxMols value to display more.
           % (maxMols))

Out[2]:
```

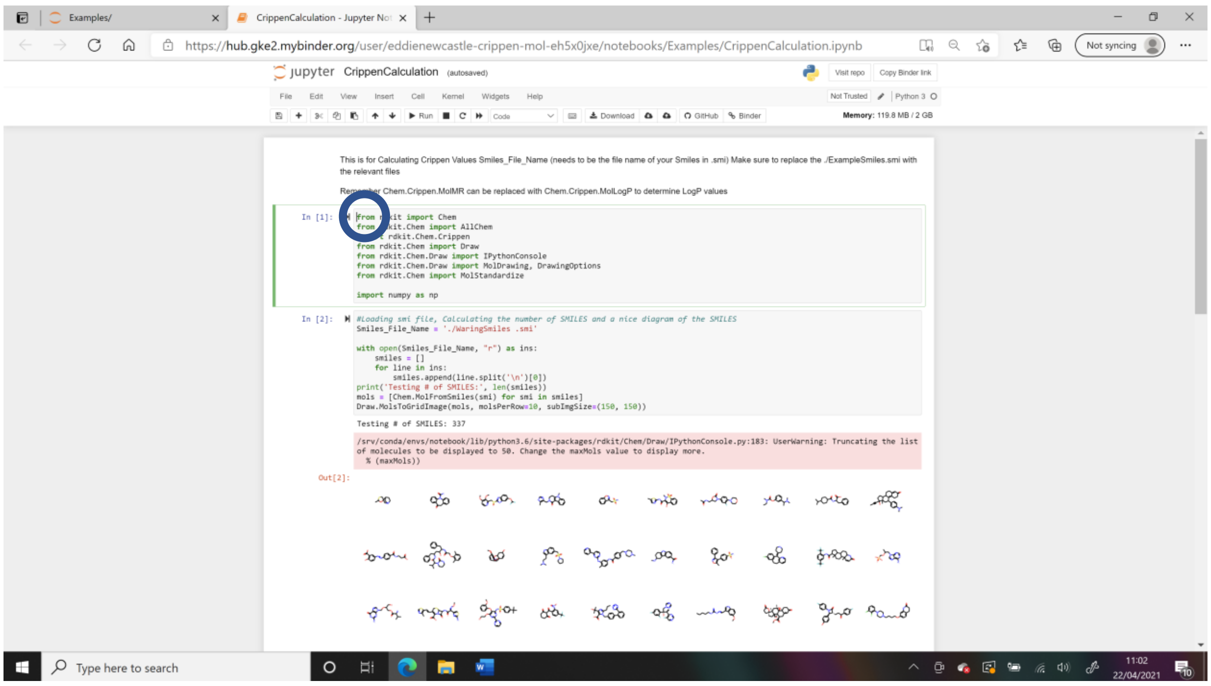
7) Replace ExampleSmiles.smi for the SMILES file uploaded.

The file I uploaded was called WaringSmiles .smi


Examples/
CrippenCalculation - Jupyter Not
https://hub.gke2.mybinder.org/user/eddienewcastle-crippen-mol-eh5x0jxe/notebooks/Examples/CrippenCalculation.ipynb
Not syncing
jupyter CrippenCalculation (autosaved)
Visit repo
Copy Binder link
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
Download
GitHub
Binder
Memory: 119.3 MB / 2 GB

```
         import numpy as np

In [2]:  #Loading smi file, Ca[illegible]number of SMILES and a nice diagram of the SMILES
         Smiles_File_Name = '[illegible]ExampleSmiles[illegible]mi'

         with open(Smiles_File_Name, [illegible]) as ins:
             smiles = []
             for line in ins:
                 smiles.append(line.split('\n')[0])
         print('Testing # of SMILES:', len(smiles))
         mols = [Chem.MolFromSmiles(smi) for smi in smiles]
         Draw.MolsToGridImage(mols, molsPerRow=10, subImgSize=(150, 150))

         Testing # of SMILES: 337

         /srv/conda/envs/notebook/lib/python3.6/site-packages/rdkit/Chem/Draw/IPythonConsole.py:183: UserWarning: Truncating the list
         of molecules to be displayed to 50. Change the maxMols value to display more.
           % (maxMols))

Out[2]:

In [3]:  #Calculating the Crippen.MolMR
         suppl = Chem.SmilesMolSupplier('[illegible]ExampleSmiles[illegible]i',delimiter='\n',titleLine=False, nameColumn=0)
         for mol in suppl:
             mr = Chem.Crippen.MolMR(mol)
```

8) Once the ExampleSmiles have been changed to the uploaded file name. We click on the top of the cell of the first line.



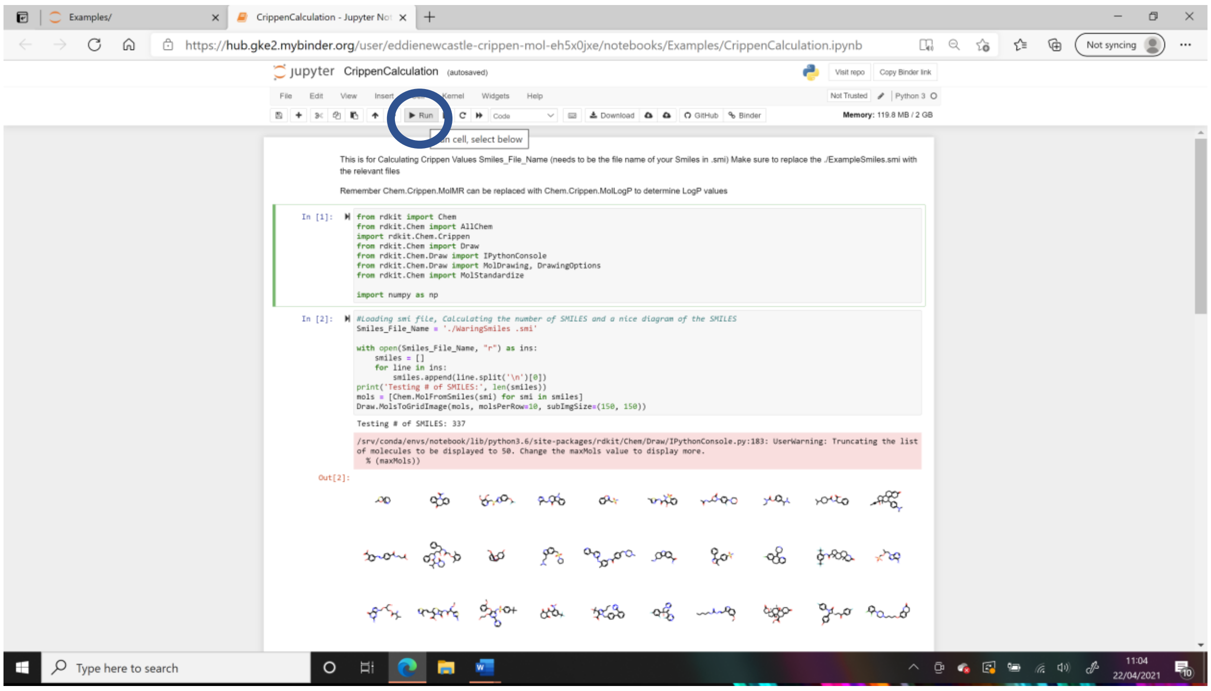
This is for Calculating Crippen Values Smiles_File_Name (needs to be the file name of your Smiles in .smi) Make sure to replace the ./ExampleSmiles.smi with the relevant files

Re[illegible]er Chem.Crippen.MolMR can be replaced with Chem.Crippen.MolLogP to determine LogP values

```
In [1]:  from r[illegible]kit import Chem
         from [illegible]kit.Chem import AllChem
         [illegible]t rdkit.Chem.Crippen
         from rdkit.Chem import Draw
         from rdkit.Chem.Draw import IPythonConsole
         from rdkit.Chem.Draw import MolDrawing, DrawingOptions
         from rdkit.Chem import MolStandardize

         import numpy as np

In [2]:  #Loading smi file, Calculating the number of SMILES and a nice diagram of the SMILES
         Smiles_File_Name = './WaringSmiles .smi'

         with open(Smiles_File_Name, "r") as ins:
             smiles = []
             for line in ins:
                 smiles.append(line.split('\n')[0])
         print('Testing # of SMILES:', len(smiles))
         mols = [Chem.MolFromSmiles(smi) for smi in smiles]
         Draw.MolsToGridImage(mols, molsPerRow=10, subImgSize=(150, 150))

         Testing # of SMILES: 337

         /srv/conda/envs/notebook/lib/python3.6/site-packages/rdkit/Chem/Draw/IPythonConsole.py:183: UserWarning: Truncating the list
         of molecules to be displayed to 50. Change the maxMols value to display more.
           % (maxMols))
Out[2]:
```

Examples/
CrippenCalculation - Jupyter Not
https://hub.gke2.mybinder.org/user/eddienewcastle-crippen-mol-eh5x0jxe/notebooks/Examples/CrippenCalculation.ipynb
Not syncing
jupyter CrippenCalculation (autosaved)
Visit repo
Copy Binder link
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
Download
GitHub
Binder
Memory: 119.8 MB / 2 GB
Type here to search
11:02
22/04/2021

9) Then hit run to cycle through each of the lines. Running through each of the lines runs through the code.



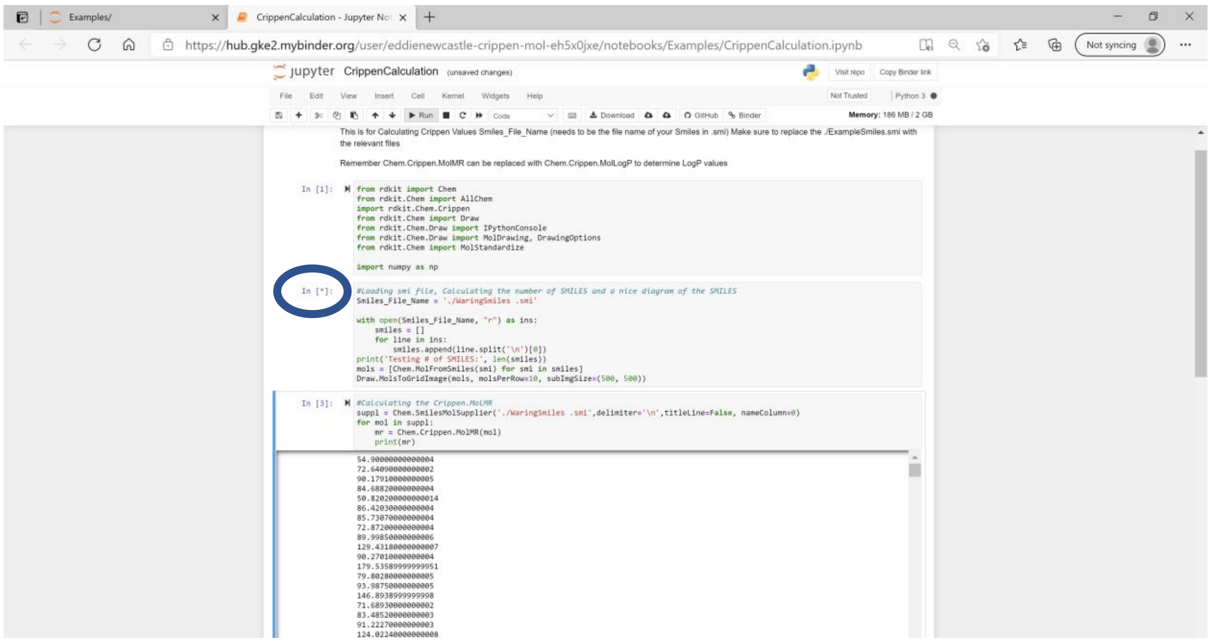
This is for Calculating Crippen Values Smiles_File_Name (needs to be the file name of your Smiles in .smi) Make sure to replace the ./ExampleSmiles.smi with the relevant files

Remember Chem.Crippen.MolMR can be replaced with Chem.Crippen.MolLogP to determine LogP values

```
In [1]:  from rdkit import Chem
         from rdkit.Chem import AllChem
         import rdkit.Chem.Crippen
         from rdkit.Chem import Draw
         from rdkit.Chem.Draw import IPythonConsole
         from rdkit.Chem.Draw import MolDrawing, DrawingOptions
         from rdkit.Chem import MolStandardize

         import numpy as np

In [2]:  #Loading smi file, Calculating the number of SMILES and a nice diagram of the SMILES
         Smiles_File_Name = './WaringSmiles .smi'

         with open(Smiles_File_Name, "r") as ins:
             smiles = []
             for line in ins:
                 smiles.append(line.split('\n')[0])
         print('Testing # of SMILES:', len(smiles))
         mols = [Chem.MolFromSmiles(smi) for smi in smiles]
         Draw.MolsToGridImage(mols, molsPerRow=10, subImgSize=(150, 150))

         Testing # of SMILES: 337

         /srv/conda/envs/notebook/lib/python3.6/site-packages/rdkit/Chem/Draw/IPythonConsole.py:183: UserWarning: Truncating the list
         of molecules to be displayed to 50. Change the maxMols value to display more.
           % (maxMols))
Out[2]:
```

Examples/
CrippenCalculation - Jupyter Not
https://hub.gke2.mybinder.org/user/eddienewcastle-crippen-mol-eh5x0jxe/notebooks/Examples/CrippenCalculation.ipynb
Not syncing
jupyter CrippenCalculation (autosaved)
Visit repo
Copy Binder link
File Edit View Insert Kernel Widgets Help
Not Trusted
Python 3
Run
n cell, select below
Code
Download
GitHub
Binder
Memory: 119.8 MB / 2 GB
Type here to search
11:04
22/04/2021

10) A (*) indicates that the program is running.



https://hub.gke2.mybinder.org/user/eddienewcastle-crippen-mol-eh5x0jxe/notebooks/Examples/CrippenCalculation.ipynb

CrippenCalculation (unsaved changes)

This is for Calculating Crippen Values Smiles_File_Name (needs to be the file name of your Smiles in .smi) Make sure to replace the ./ExampleSmiles.smi with the relevant files

Remember Chem.Crippen.MolMR can be replaced with Chem.Crippen.MolLogP to determine LogP values

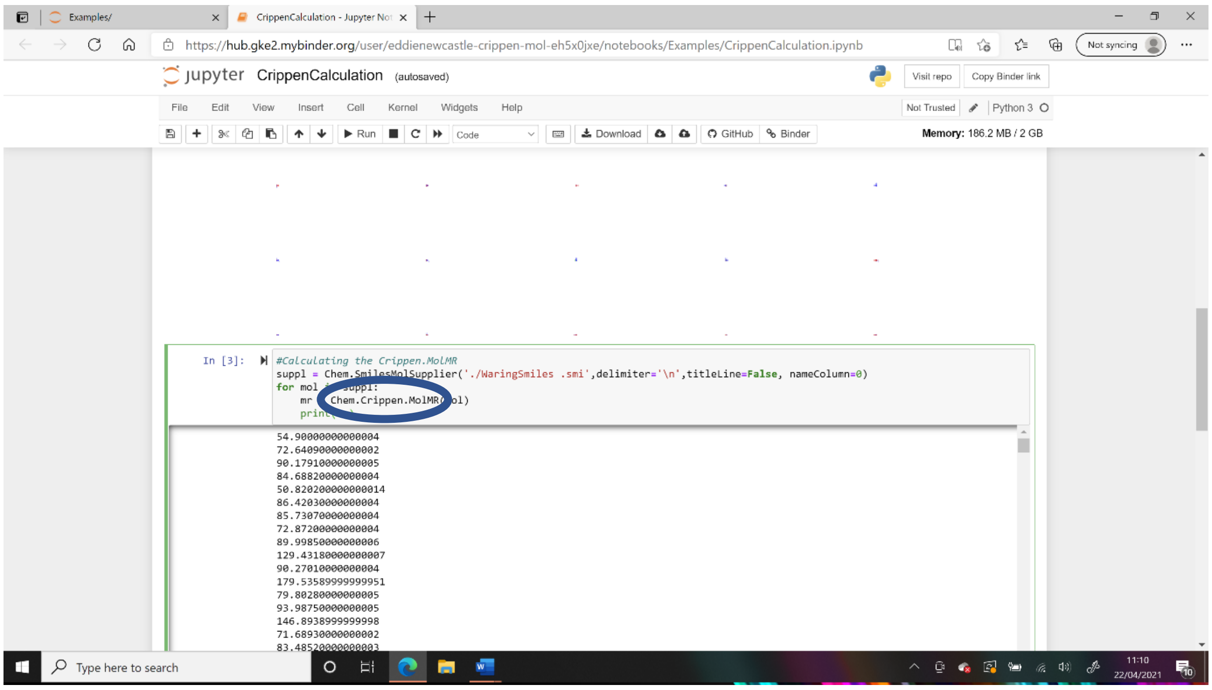
```
In [1]:  from rdkit import Chem
         from rdkit.Chem import AllChem
         import rdkit.Chem.Crippen
         from rdkit.Chem import Draw
         from rdkit.Chem.Draw import IPythonConsole
         from rdkit.Chem.Draw import MolDrawing, DrawingOptions
         from rdkit.Chem import MolStandardize

         import numpy as np

In [*]:  #Loading smi file, Calculating the number of SMILES and a nice diagram of the SMILES
         Smiles_File_Name = './WaringSmiles .smi'

         with open(Smiles_File_Name, "r") as ins:
             smiles = []
             for line in ins:
                 smiles.append(line.split('\n')[0])
         print('Testing # of SMILES:', len(smiles))
         mols = [Chem.MolFromSmiles(smi) for smi in smiles]
         Draw.MolsToGridImage(mols, molsPerRow=10, subImgSize=(500, 500))

In [3]:  #Calculating the Crippen.MolMR
         suppl = Chem.SmilesMolSupplier('./WaringSmiles .smi',delimiter='\n',titleLine=False, nameColumn=0)
         for mol in suppl:
             mr = Chem.Crippen.MolMR(mol)
             print(mr)

         54.900000000000004
         72.64090000000002
         90.17910000000005
         84.68820000000004
         50.820200000000014
         86.42030000000004
         85.73070000000004
         72.87200000000004
         89.99850000000006
         129.43180000000007
         90.27010000000004
         179.53589999999951
         79.80280000000005
         93.98750000000005
         146.8938999999998
         71.68930000000002
         83.48520000000003
         91.22270000000003
         124.02240000000008
```

This program takes your input SMILES file and calculates the Crippen.MolMR, calculates Molecular Formula and calculates the number of atoms.

To export the data you can easily highlight the printed data and then copy and paste it into Excel/Notepad or wherever you need it.

If you want to calculate Chem.Crippen.MolLogP

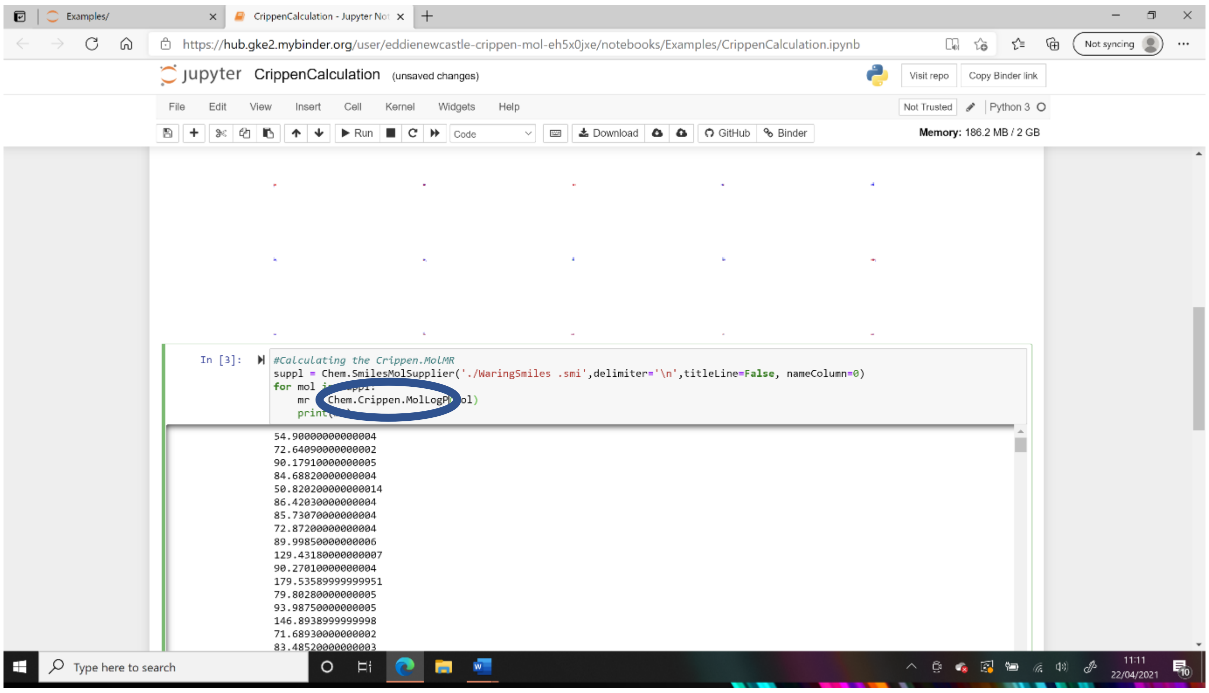
1) Replace Chem.Crippen.MolMR to Chem.Crippen.MolLogP



```
In [3]:  #Calculating the Crippen.MolMR
         suppl = Chem.SmilesMolSupplier('./WaringSmiles .smi',delimiter='\n',titleLine=False, nameColumn=0)
         for mol i  suppl:
             mr  Chem.Crippen.MolMR( ol)
             print(
```

```
54.90000000000004
72.64090000000002
90.17910000000005
84.68820000000004
50.820200000000014
86.42030000000004
85.73070000000004
72.87200000000004
89.99850000000006
129.43180000000007
90.27010000000004
179.53589999999951
79.80280000000005
93.98750000000005
146.8938999999998
71.68930000000002
83.48520000000003
```

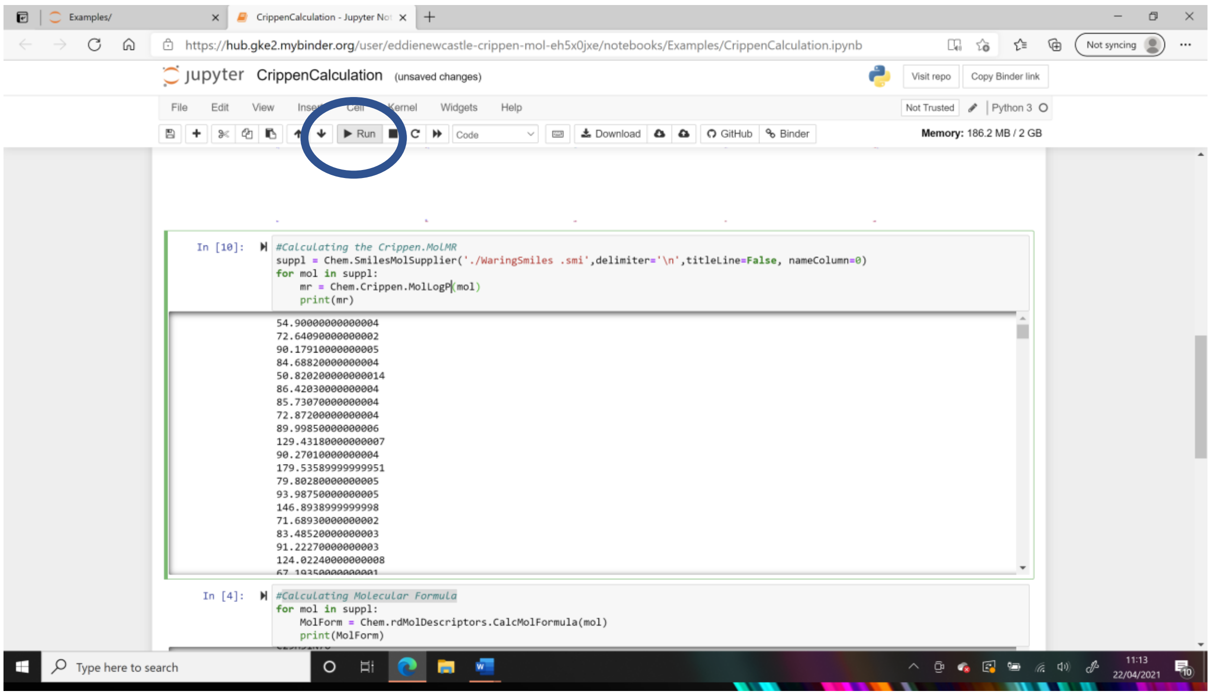
NOW IT SHOULD LOOK LIKE THIS



```
In [3]:  #Calculating the Crippen.MolMR
         suppl = Chem.SmilesMolSupplier('./WaringSmiles .smi',delimiter='\n',titleLine=False, nameColumn=0)
         for mol i   ppl:
             mr  Chem.Crippen.MolLogP( ol)
             print(
```

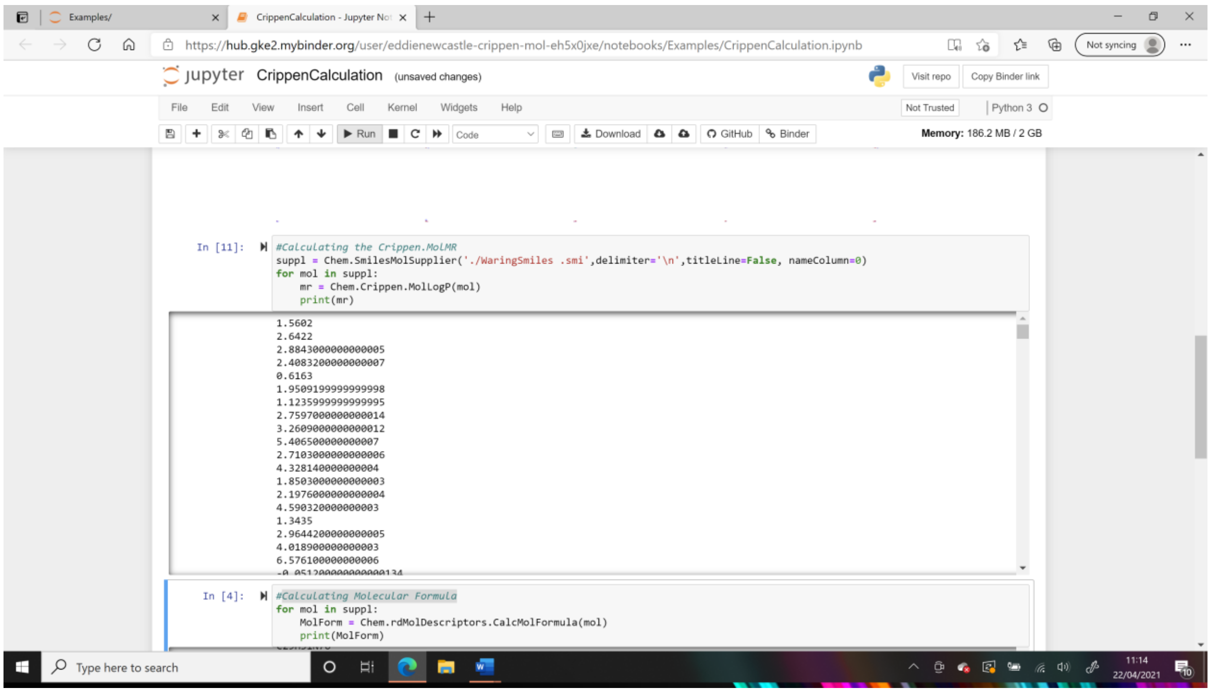
```
54.90000000000004
72.64090000000002
90.17910000000005
84.68820000000004
50.820200000000014
86.42030000000004
85.73070000000004
72.87200000000004
89.99850000000006
129.43180000000007
90.27010000000004
179.53589999999951
79.80280000000005
93.98750000000005
146.8938999999998
71.68930000000002
83.48520000000003
```

2) Then click run


Examples/
CrippenCalculation - Jupyter No
https://hub.gke2.mybinder.org/user/eddienewcastle-crippen-mol-eh5x0jxe/notebooks/Examples/CrippenCalculation.ipynb
Not syncing
jupyter CrippenCalculation (unsaved changes)
Visit repo
Copy Binder link
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3
Run
Code
Download
GitHub
Binder
Memory: 186.2 MB / 2 GB
Type here to search
11:13
22/04/2021

```
In [10]:  #Calculating the Crippen.MolMR
          suppl = Chem.SmilesMolSupplier('./WaringSmiles .smi',delimiter='\n',titleLine=False, nameColumn=0)
          for mol in suppl:
              mr = Chem.Crippen.MolLogP(mol)
              print(mr)

          54.90000000000004
          72.64090000000002
          90.17910000000005
          84.68820000000004
          50.820200000000014
          86.42030000000004
          85.73070000000004
          72.87200000000004
          89.99850000000006
          129.43180000000007
          90.27010000000004
          179.53589999999951
          79.80280000000005
          93.98750000000005
          146.8938999999998
          71.68930000000002
          83.48520000000003
          91.22270000000003
          124.02240000000008

In [4]:   #Calculating Molecular Formula
          for mol in suppl:
              MolForm = Chem.rdMolDescriptors.CalcMolFormula(mol)
              print(MolForm)
```

Then it should look like this


Examples/
CrippenCalculation - Jupyter No
https://hub.gke2.mybinder.org/user/eddienewcastle-crippen-mol-eh5x0jxe/notebooks/Examples/CrippenCalculation.ipynb
Not syncing
jupyter CrippenCalculation (unsaved changes)
Visit repo
Copy Binder link
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3
Run
Code
Download
GitHub
Binder
Memory: 186.2 MB / 2 GB
Type here to search
11:14
22/04/2021

```
In [11]:  #Calculating the Crippen.MolMR
          suppl = Chem.SmilesMolSupplier('./WaringSmiles .smi',delimiter='\n',titleLine=False, nameColumn=0)
          for mol in suppl:
              mr = Chem.Crippen.MolLogP(mol)
              print(mr)

          1.5602
          2.6422
          2.8843000000000005
          2.4083200000000007
          0.6163
          1.9509199999999998
          1.1235999999999995
          2.7597000000000014
          3.2609000000000012
          5.406500000000007
          2.7103000000000006
          4.328140000000004
          1.8503000000000003
          2.1976000000000004
          4.590320000000003
          1.3435
          2.9644200000000005
          4.018900000000003
          6.576100000000006

In [4]:   #Calculating Molecular Formula
          for mol in suppl:
              MolForm = Chem.rdMolDescriptors.CalcMolFormula(mol)
              print(MolForm)
```