

# Éléments et molécules

## Librairies initiale de Christoph Gohlke (2005-2013)

Les propriétés des éléments chimiques, de molécules peuvent être dressées, listées,... par un programme si on dispose des données. [Christoph Gohlke](#) (University of California, Irvine), a écrit deux librairies, [elements.py](#) et [molmass.py](#), pour respectivement accéder aux propriétés des éléments, et calculer les masses moléculaires, compositions élémentaires, spectres de distribution des masses de molécules,... Gohlke fournit également un interface utilisateur graphique sous forme d'un [tableau périodique](#), et un [interface web](#) pour afficher les données sur les masses moléculaires. Tous ces programmes sont mis à disposition aux conditions de la [licence BSD](#) (licence libre très permissive). **Les 4 programmes, dans leur version de mars 2013 sont regroupés dans [cette archive](#).**

Voici deux exemples simples de programmes utilisant ces librairies :

[molmass-elements-01.py](#)

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
# utilisations de elements.py & molmass.py
# source : http://www.lfd.uci.edu/~gohlke/
# mars 2020 : adaptation python 3
from elements import ELEMENTS

len(ELEMENTS)
str(ELEMENTS[109])
ele = ELEMENTS['C']
print(ele.number, ele.symbol, ele.name, ele.eleconfig)
print(ele.eleconfig_dict)
print(sum(ele.mass for ele in ELEMENTS))
for ele in ELEMENTS:
    ele.validate()
    ele = repr(ele)
    print(ele, type(ele))
    input("next ?")
```

[molmass-elements-02.py](#)

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
# utilisations de elements.py & molmass.py
# source : http://www.lfd.uci.edu/~gohlke/
# mars 2020 : adaptation python 3
from molmass import Formula

f = Formula("D2O")
#f = Formula("H2SO4")
```

```
print(f.formula)

print(f.empirical)

print(f.mass) # average mass
print(f.isotope.massnumber) # nominal mass
print(f.isotope.mass) # monoisotopic mass
print(f.atoms)
print(f.composition())
print(f.spectrum())
```

Un autre programme, testant plus de possibilités :

[molmass-elements-03.py](#)

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
# utilisations de elements.py & molmass.py
# source : http://www.lfd.uci.edu/~gohlke/
# test des routines sur les formules chimiques
# mars 2020 : adaptation python 3
from molmass import *

def info(object, spacing=10, collapse=1):
    """Print methods and doc strings.

    Takes module, class, list, dictionary, or string."""
    methodList = [method for method in dir(object) if
callable(getattr(object, method))]
    processFunc = collapse and (lambda s: " ".join(s.split())) or
(lambda s: s)
    print("\n".join(["%s %s" %
(method.ljust(spacing),
processFunc(str(getattr(object,
method).__doc__)))
for method in methodList]))

f = Formula('CHCl3')
#f = Formula('C2H2Cl6')
line='_____ '

print(line)
print(dir(f))
print(line)
print(info(f))
```

```
print(line)
print(f.composition())
print(line)
print(f.isotope)
print(line)
print(f.isotope.mass)
print(line)
print(f.isotope.massnumber)
print(line)
print(f.mass)
print(line)
print(f.formula, f.gcd, f.empirical)
print(line)
print(f.spectrum())
print(line)
print(f.atoms)
print(line)
print(analyze ('CHCl3'))
print(line)
print(f, type(f))
print(line)
```

From:

<https://dvillers.umons.ac.be/wiki/> - Didier Villers, UMONS - wiki

Permanent link:

[https://dvillers.umons.ac.be/wiki/teaching:progappchim:elements\\_molecules?rev=1583750281](https://dvillers.umons.ac.be/wiki/teaching:progappchim:elements_molecules?rev=1583750281)

Last update: 2020/03/09 11:38

