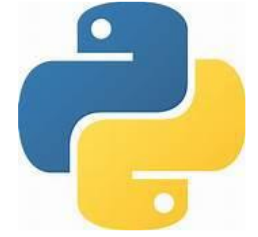




Démystifier les langages de haut niveau

Outils Numériques / Semestre 5
/ Institut d'Optique / B0_1

Distributions / Environnements



Distribution : ensemble de logiciels et de librairies
incluant des environnements et des interpréteurs



Bibliothèques : ensemble de
modules supplémentaires
*incluant des classes, des
fonctions...*



Environnement (IDE) : ensemble d'outils pour l'édition
et l'interprétation des commandes / programmes
incluant des interpréteurs et des éditeurs de texte



Quelques rappels sous Python



- Variables

```
a = 2 + 3  
print( a )  
print( 'a =', a )
```

```
5  
a = 5
```

- Listes

```
b = [1, 2, 3]  
print( b )
```

```
[1, 2, 3]
```

```
print( b[1] )
```

```
2
```

Quelques rappels sous Python



- Utilisation de bibliothèques

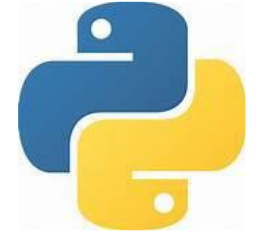
```
import numpy  
ma = numpy.array( [1, 2, 3] )
```

```
import numpy as np  
ma = np.array( [1, 2, 3] )
```

```
from matplotlib import pyplot  
pyplot.figure()
```

```
from matplotlib import pyplot as plt  
plt.figure()
```

Quelques rappels sous Python



- Installation de bibliothèques / packages

Dans un shell/prompt

> pip install numpy

Dans un shell/prompt (Anaconda)

> conda install numpy

Quelques rappels sous Python



- Utilisation des vecteurs / matrices

```
import numpy as np  
mb = np.array( [[1,2,3] , [4,5,6]] )  
mc = np.array( [[1,2,3] , [4,5,6]] )  
mm = mb + mc  
print( mm )
```

```
[[ 2  4  6]  
 [ 8 10 12]]
```

Quelques rappels sous Python



- Nombres complexes

```
import numpy as np  
mk = np.array([1j, 2, 3], dtype=complex)  
print( mk )
```

```
[ 0+1j  2+0j  3+0j]
```

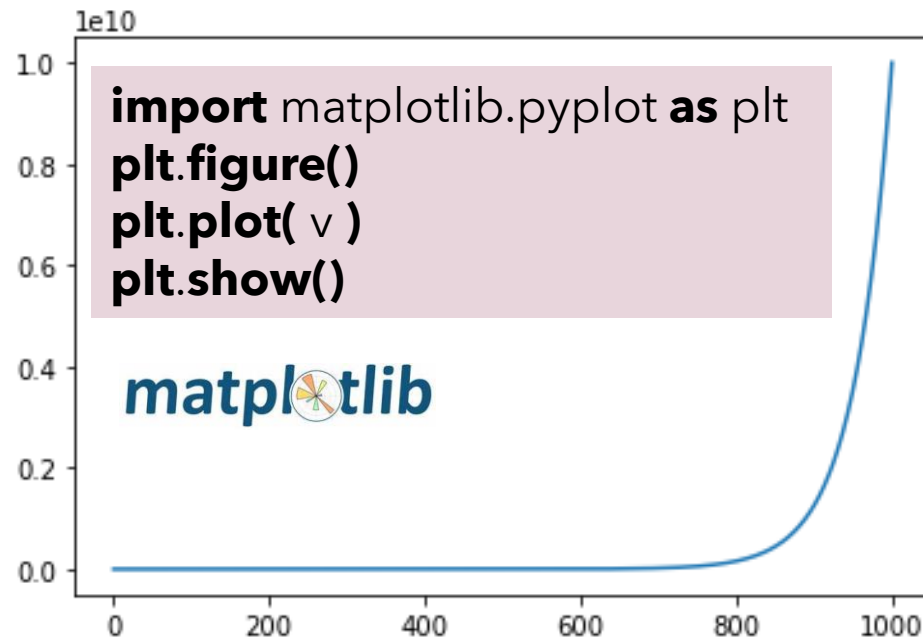
```
nk = 1j + 3  
print( nk )  
print( type( nk ) )
```

```
(1j + 3)  
<class 'complex'>
```

Quelques rappels sous Python

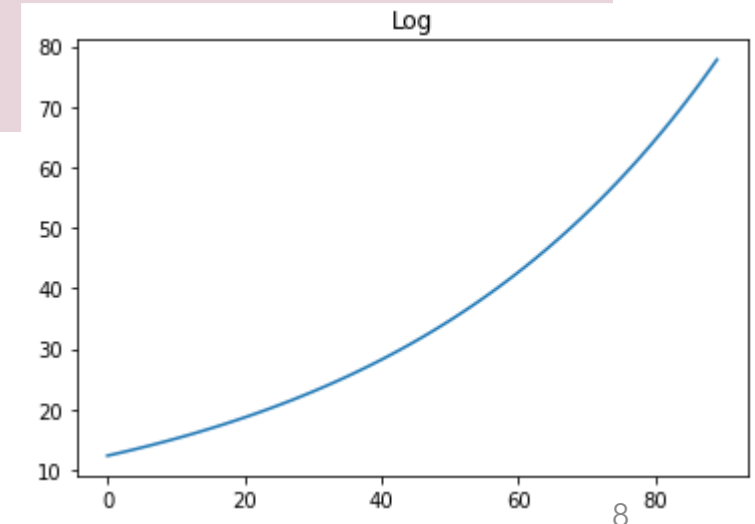


- Vecteurs (suite)



```
import numpy as np
v = np.logspace( 1, 10, 1001 )
```

```
v2 = v[ 10 : 100 ]
plt.figure()
plt.plot( v2 )
plt.show()
```



Trucs et Astuces



- Affichage des figures

Tools / Preferences

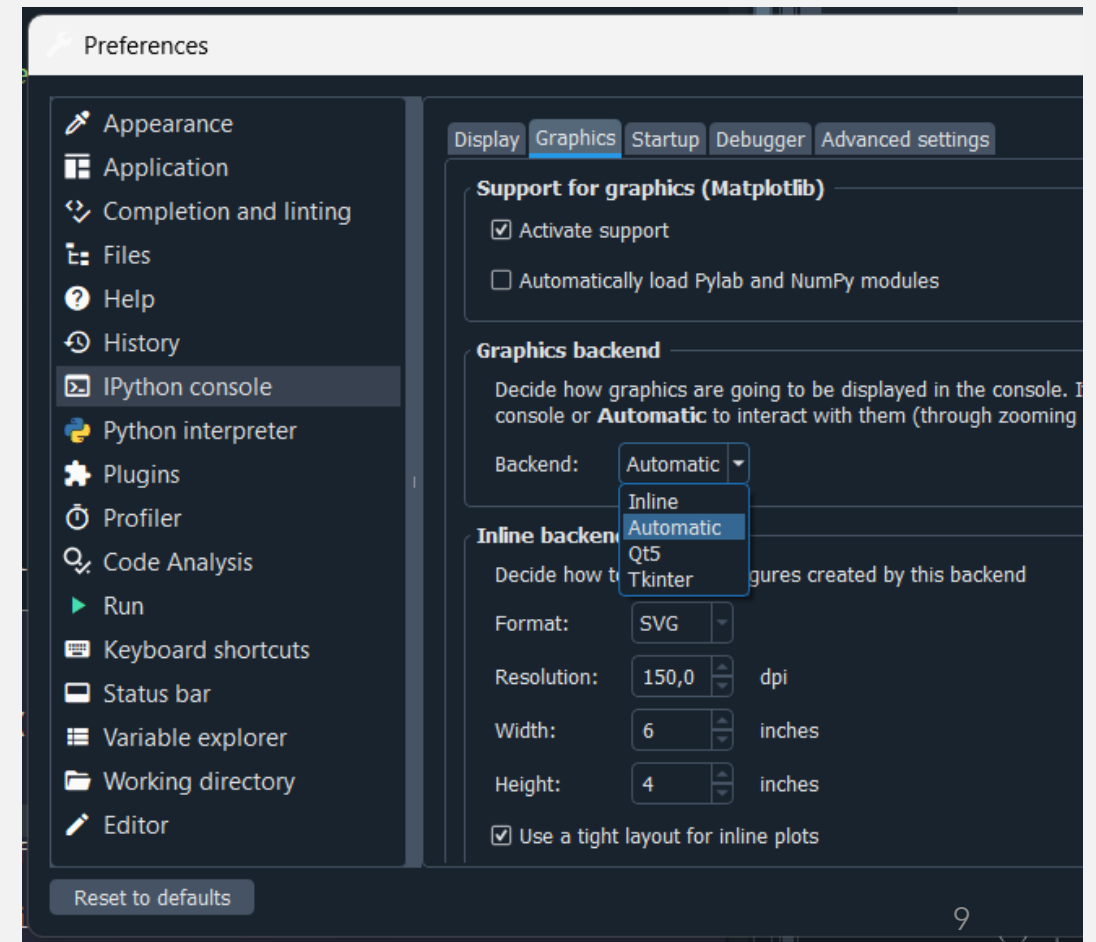
ou

Outils / Préférences

IPython console
Graphics

Activate Support

Backend : **Automatic**



Quelques rappels sous Python



- Try / Except (ValueError...)

HANDLING SPECIFIC EXCEPTIONS

- have **separate except clauses** to deal with a particular type of exception

```
try:
    a = int(input("Tell me one number: "))
    b = int(input("Tell me another number: "))
    print("a/b = ", a/b)
    print("a+b = ", a+b)
except ValueError:
    print("Could not convert to a number.")
except ZeroDivisionError:
    print("Can't divide by zero")
except:
    print("Something went very wrong.")
```

Doit-on faire confiance aux ordinateurs ?

- Testez les deux calculs suivants sous Python

> 3 - 2 - 1

> 0.3 - 0.2 - 0.1



Toutes les données sont traitées pareil !?

- Testez les deux calculs suivants sous Python

> 3 - 2 - 1

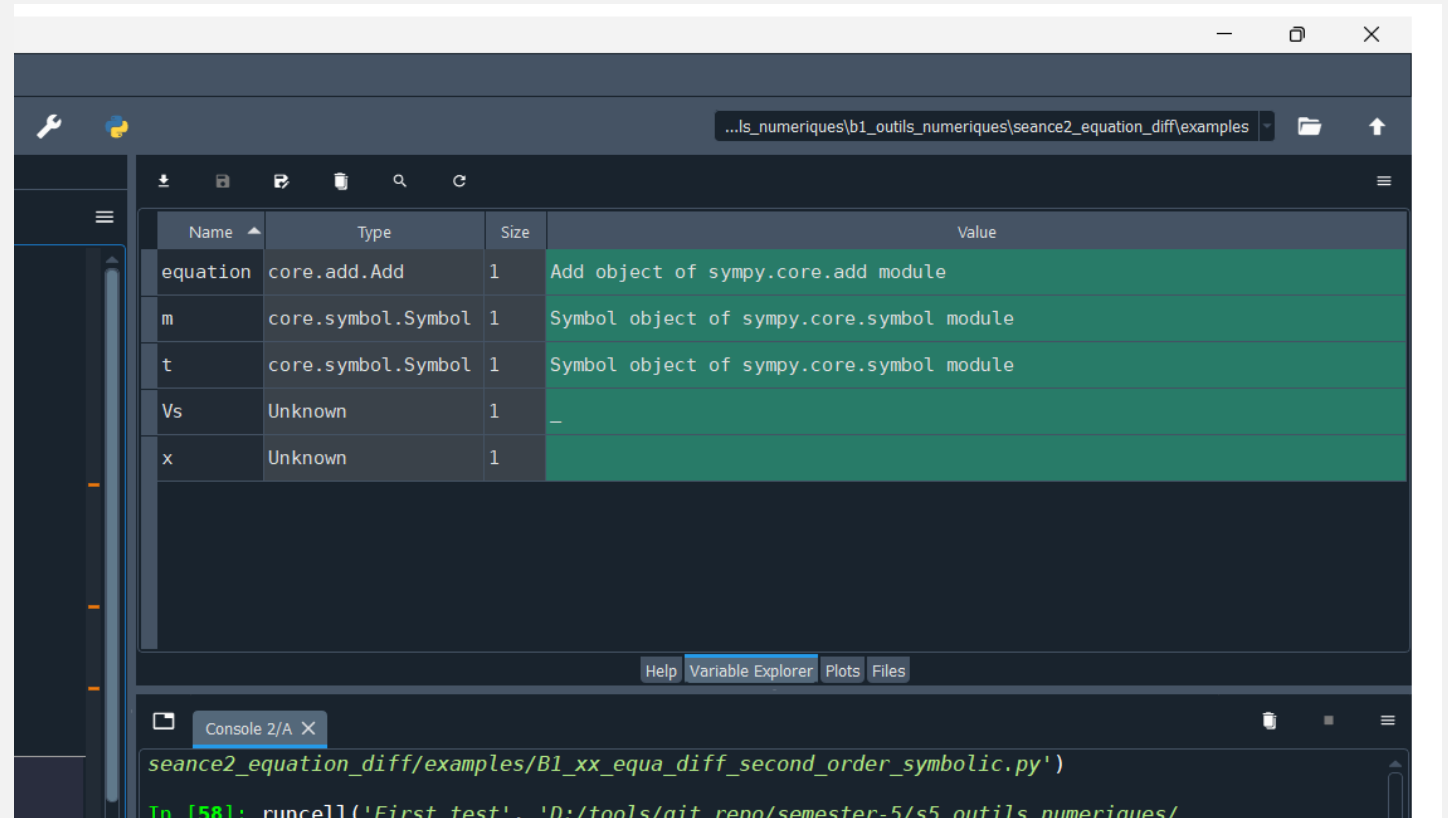
> 0.3 - 0.2 - 0.1



Trucs et Astuces



- *Variable explorer*



Listes, vecteurs... c'est pareil, non ?

- TO DO

Travailler avec des vecteurs

- Sum/Mean on axis=0 or 1

Résoudre des problèmes linéaires

- Linalg from numpy

Trucs et Astuces



- Sections

#%%

```
36
37 #%% Frequency Response / Bode
38 w = np.logspace(1, 6, 101)
39 mag, phase, w = ct.bode_plot(sysRC.getTF(), w, plot=True)
40 mag_db = 20*np.log(mag)
41 phase_deg = phase * 180 / np.pi
42 f = w/(2*np.pi)
43
```

*Exécutables indépendamment
(... ou presque)*

