

# Nivell 1.

## Exercici 1:

Instal·la el programa Anaconda amb Python 3, i Jupyter Notebook.

Ja el tenia instal·lat, només és seguir els passos per la descàrrega a [Anaconda](#) que també instal·la Jupyter.

## Exercici 2:

Utilitzant Jupyter Notebook executa alguns càlculs senzills, a la vegada que et familiaritzes amb el llenguatge Markdown.

Realitza els següents exercicis:

- Calcula l'àrea d'un rectangle de costats 3 i 4,  $S = a \cdot b$
- Calcula l'àrea d'un cercle de radi 2,  $S = \pi r^2$
- Calcula  $\sqrt{9}$
- Calcula els 15 primers nombres de la sèrie de Fibonacci

```
In [1]: a=3
b=4
print('L'àrea del rectangle de costats %s i %s és %s' % (a, b, a*b))
```

L'àrea del rectangle de costats 3 i 4 és 12

```
In [2]: from math import pi
radio=2
area=pi*radio**2
print('L'àrea del cercle de radi {0} és {1:2.2f}'.format(radio,area))
```

L'àrea del cercle de radi 2 és 12.57

```
In [3]: from math import sqrt
n=9
print('L'arrel quadrada de {0} és {1}'.format(n,sqrt(n)))
```

L'arrel quadrada de 9 és 3.0

```
In [4]: n=15
a,b=0,1
lista=[a,b]
for i in range(n-2):
    a,b=b,a+b
    lista.append(b)
print('Els {0} primers nombres de la sèrie de Fibonacci són:'.format(n))
print(*lista)
```

Els 15 primers nombres de la sèrie de Fibonacci són:  
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377

## Exercici 3:

Prova de crear títols, llistes, canviar l'estil de la lletra o afegir imatges dins del Notebook.

# Cien años de soledad (h1)

## Gabriel García Márquez (h3)

*Considerada una de las cumbres de las letras.... (h6)*

## Llistat desordenat:

- Dilluns
- Dimarts
- Dimecres
- ...

## Llistat ordenat:

1. Gener
2. Febrer
3. Març
4. ...

# Diferents estils de lletra:

**En bold**

*En italic*

Subratllat

Blockquote

[Link a IT Academy.](#)

# Taula:

## Classificació Lliga:

Equip	Partits guanyats	Punts
<i>Atletico</i>	10	<b>15</b>
<i>Barcelona</i>	10	<b>14</b>
<i>R. Madrid</i>	10	<b>12</b>

# Afegir imatges:



# Nivell 2.

## Exercici 1:

Exporta el Notebook com a pdf i com a html.

Per exportar-lo com a HTML només hem d'anar a file/Download as/HTML i em genera el fitxe HTML. Per a pdf anant a file/Download as/PDF via Latex em dóna error perquè no troba la llibreria xelatex. He triat l'opció de exportar-lo a PDF via HTML en comptes via Latex. Aquesta opció no surt al menú, per aconseguir-ho m'he descarregat jupyter-as-pdf seguint els passos d'[aquí](#)

Per últim, una vegada que tenim la nova opció al menú només fent file/Download as/PDF via HTML em crea el fitxer PDF.

# Nivell 3.

## Exercici 1:

Instal·la Nbextensions al Notebook de Jupyter.

He instal·lat Nbextensions mitjançant la shell de Conda seguint els passos d'aquest [link](#).

També hem trobat aquest link [Nbextesions per a data scientist](#) on mostra una llista de les extensions més recomanades per a data science.

En el nostre cas instal·larem **CodeFolding** i **Variable Inspector**

Per a la primera opció, a la tag nbextensionsConfig, he activat l'extensió **CodeFolding** que permet plegar codi de funcions, loops, classes, etc. És molt útil per fer un codi més llegible i net.

```
In [5]: #Clicant sobre la fletxa el codi es plega i desplega
#plegat
def prueba(a,b):
    s=a+b
    return s
```

```
In [6]: #mateix codi desplegat
def prueba(a,b):
    s=a+b
    return s
```

```
In [7]: prueba(3,4)
```

Out[7]: 7

He afegit una imatge del codi tancat i obert ja que al github es veu només desplegat.

```
In [9]: from PIL import Image
Image.open('captura.png')
```

```
Out[9]: 1 ▾ #Clicant sobre la fletxa el codi es plega i desplega
2 #plegat
3 ▸ def prueba(a,b):↔
```

```
1 ▾ #mateix codi desplegat
2 ▾ def prueba(a,b):
3   s=a+b
4   return s
```

Per últim també hem triat l'extensió **Variable Inspector** que et permet veure en una petita finestra els valors, nom i tipus de les variables carregades a memòria. Per exemple després d'executar les primeres cel.les veient les variables creades i els valors que guarden sense necessitat d'executar un print o un type adhoc.

```
In [10]: Image.open('captura2.png')
```

```
Out[10]: Variable Inspector
v[-][x]

X ◆ Name ◆ Type ◆ Size ◆ Value
-----
x a int 28 233
x area float 24 12.566370614359172
x b int 28 377
x i int 28 12
v lista list 208 [0 1 1 2 3 5 8 13 21 34 55 89 144 233 377]
```

```
In [ ]:
```