Introduction to Jupyter notebooks

Here is a Jupyter notebooks tutorial. Notebooks can be used for mixing code, text, and images. A notebook consists of a number of cells. Each cell is of one of four types: Code, Markdown, Raw NB Convert, and Heading. The type is selected in the dropdown menu above. We will only need Code and Markdown.

- Cells of all types can be evaluated
- Press the button Run to evaluate the current cell (or type Shift + return)
- Press the button FastForward to evaluate all cells in the notebook

Cells of type Code

- You can write Python code in these cells
- The code is evaluated just like in a Python interpreter

In [25]:
```
3+3 #pocket calculator
```

Out[25]:
```
6
```

In [26]:
```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
  print(x)
```

```
apple
banana
cherry
```

Try typing print and then Shift + tab.

Cells of type Markdown

- Markdown is a simple language for producing text that can be read by browsers
- You can learn to use it in 10 minutes!
- Markdown tutorial: https://guides.github.com/features/mastering-markdown/

Headings

Write # Big heading, ## Smaller heading, etc. Don't forget the space. Use 1-6 hashes.

Styles

You can make words **bold** or *italic*

Unordered lists

- Item 1

- Item 2
  - Item 2a
  - Item 2b

Ordered lists

1. Item 1
2. Item 2
3. Item 3
   A. Item 3a
   B. Item 3b

Quotes

As Kanye West said:

> We're living the future so the present is our past.

Tables

Column 1	Column 2
Row 1 Col 1	Row 1 Col 2
Row 2 Col 1	Row 2 Col 2

Images

Links

URL-addresses become clickable: http://github.com. You can also use hidden links

Latex

Latex works fine in Markdown! θ

TEST

small

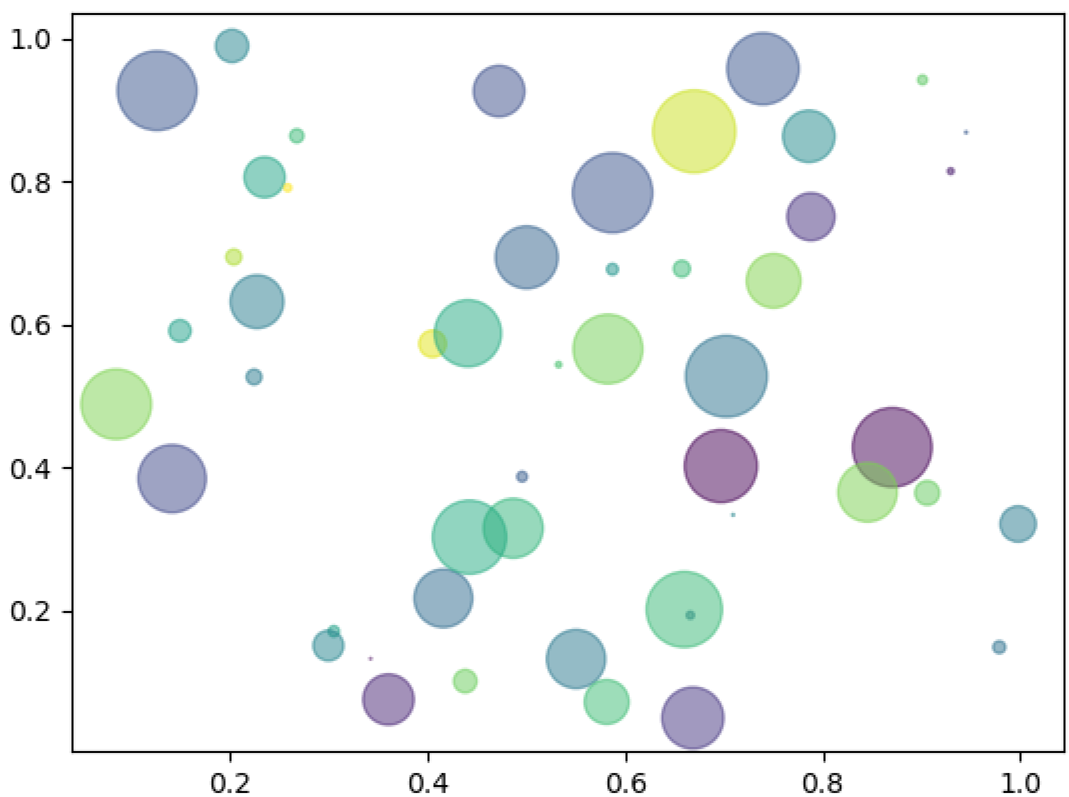
In [27]:
```
%lsmagic
```

Out[27]:
```
Available line magics:
%alias  %alias_magic  %autoawait  %autocall  %automagic  %autosave  %bookmark  %cat  %cd
  %clear  %colors  %conda  %config  %connect_info  %cp  %debug  %dhist  %dirs  %doctest_
mode  %ed  %edit  %env  %gui  %hist  %history  %killbgscripts  %ldir  %less  %lf  %lk  %
ll  %load  %load_ext  %loadpy  %logoff  %logon  %logstart  %logstate  %logstop  %ls  %ls
magic  %lx  %macro  %magic  %man  %matplotlib  %mkdir  %more  %mv  %notebook  %page  %pa
```

```
stebin  %pdb  %pdef  %pdoc  %pfile  %pinfo  %pinfo2  %pip  %popd  %pprint  %precision  %
prun  %psearch  %psource  %pushd  %pwd  %pycat  %pylab  %qtconsole  %quickref  %recall
%rehashx  %reload_ext  %rep  %rerun  %reset  %reset_selective  %rm  %rmdir  %run  %save
%sc  %set_env  %store  %sx  %system  %tb  %time  %timeit  %unalias  %unload_ext  %who  %
who_ls  %whos  %xdel  %xmode

Available cell magics:
%%!  %%HTML  %%SVG  %%bash  %%capture  %%debug  %%file  %%html  %%javascript  %%js  %%la
tex  %%markdown  %%perl  %%prun  %%pypy  %%python  %%python2  %%python3  %%ruby  %%scrip
t  %%sh  %%svg  %%sx  %%system  %%time  %%timeit  %%writefile

Automagic is ON, % prefix IS NOT needed for line magics.
```

In [28]:
```
%matplotlib inline
```

In [29]:
```
import numpy as np
import matplotlib.pyplot as plt

N = 50
x = np.random.rand(N)
y = np.random.rand(N)

colors = np.random.rand(N)

area = (30 * np.random.rand(N))**2  # 0 to 15 point radii

plt.scatter(x, y, s=area, c=colors, alpha=0.5)

plt.show()
```



In [30]:
```
%%HTML
<iframe width="560" height="315" src="https://www.youtube.com/watch?v=9ll_pth4Sss" frame
```

In [31]:
```
%%timeit
square_evens = [n*n for n in range(1000)]
```

```
62.5 µs ± 2.41 µs per loop (mean ± std. dev. of 7 runs, 10000 loops each)
```

In [32]:
```
import pandas as pd
import numpy as np

df = pd.DataFrame(np.random.randn(10, 5))

df
```

Out[32]:

	0	1	2	3	4
0	-0.657295	-0.137970	0.335545	1.147721	-0.509878
1	-0.833602	-0.937949	-0.743393	1.223935	-0.534960
2	-0.469307	1.119968	-0.732124	0.282951	-0.173881
3	0.141843	-2.215394	-0.778181	-0.105122	0.476768
4	1.559953	-0.530989	-1.248301	-0.295968	-1.157848
5	1.944005	-0.687398	-0.211380	0.369830	-0.529440
6	0.245025	1.225401	-0.940832	-0.056975	-0.322635
7	-0.162730	0.270449	-0.722988	0.959309	-1.213432
8	-0.606884	1.054427	-0.625560	0.709589	-0.692293
9	1.193705	-1.488465	0.150382	0.971385	-1.892882

In [ ]:
```

```