

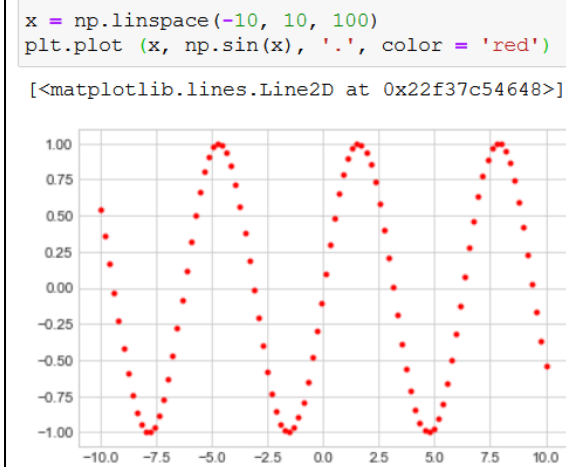
Objective: This worksheet introduces the basics of the scatter plots using **Matplotlib** package of Python.

- The Matplotlib package is imported for its interfaces and the style is set as it was explained in the previous worksheet.

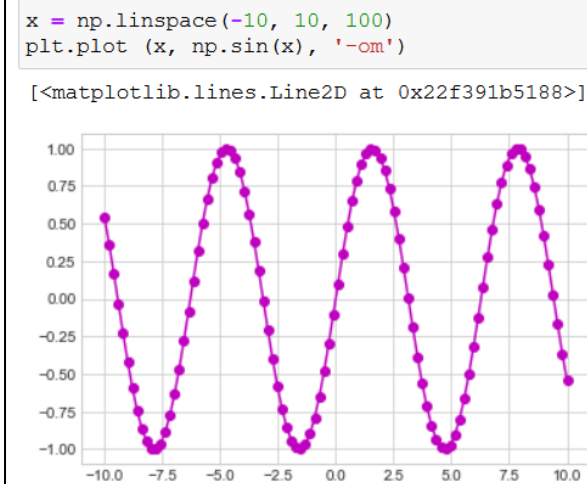
```
%matplotlib inline
import matplotlib as mpl
import matplotlib.pyplot as plt
import numpy as np

plt.style.use('seaborn-whitegrid')
```

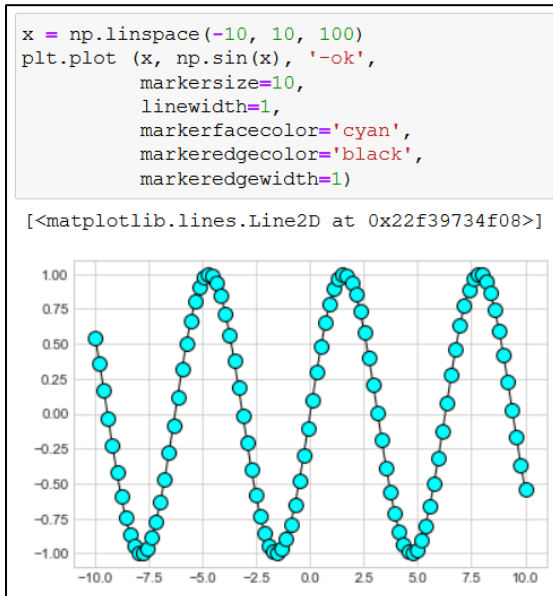
- The **plot ()** function can be used to draw a scatter plot also by providing a **3rd argument** which describes the symbol to be used for the plotting. For example $\sin(x)$ is to be drawn as a scatter plot. It has to be in the red colour and the plotting symbol is dot. This can be achieved in the following way:



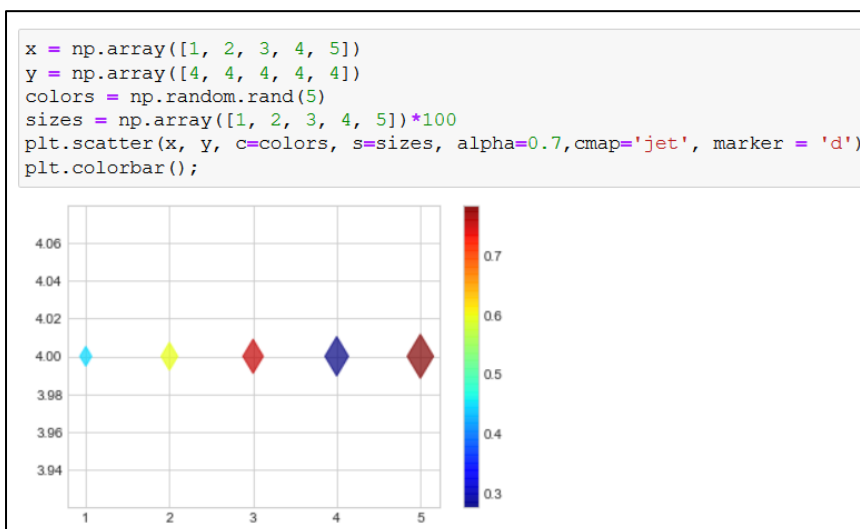
- The symbols available for the plotting can be seen by executing **plt.plot?**
- The options can be used in more compact ways. For example, if dots are to be joined by dash and the colour has to be magenta. All these effects can be achieved using **'-om'** option. In it, - is the dash, o is the circle and m is for magenta colour.



- There are several symbol attributes (marker attributes) that can be provided as arguments for the scatter plot. It is shown below as an example and they are intuitive enough to understand their corresponding meanings. You can try out different options to see the impact.



- Matplotlib provides a function `scatter()` that can also be used to draw scatter plot. It is more powerful than the `plot()` function to draw a scatter plot because each point can be changed for its colour, size and dimension. Review the example below:



- In the above example, `colors` is an array five random values, `sizes` defines five different sizes, `alpha` is used for the transparency level from 0 to 1 (0: transparent, 1: opaque), `cmap` is colour map and `marker` is used as the diamond shape (d).
- The function `colorbar()` works only when `c` is provided as float and it provides the colour mapping to the size of the points.
- `plot()` function is more efficient than the `scatter()` function for larger datasets. This is because the additional flexibility that is provided in the latter.