



Data Application Lab

Analytics Foundation

Outline

1

Demystify “Data Scientist”

2

Python key packages

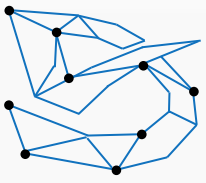
3

Pandas – data analytics

4

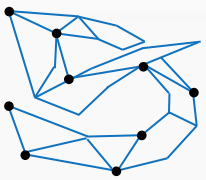
Case studies in Python



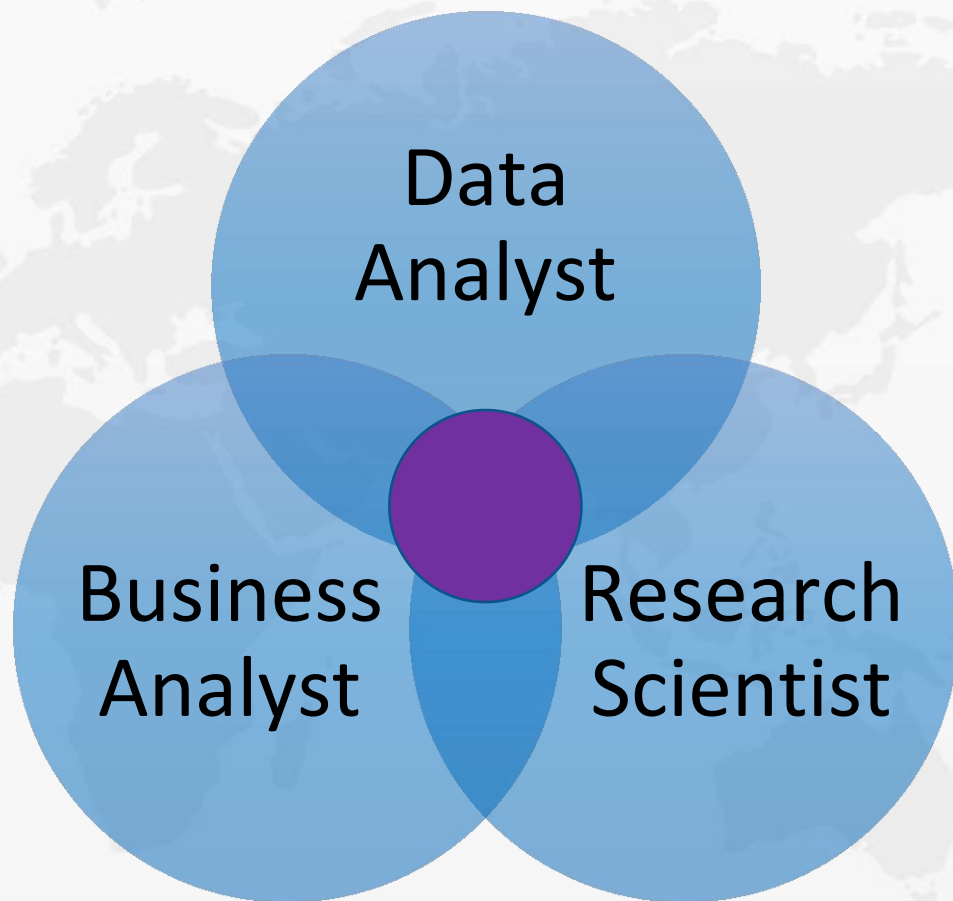


What is Data Scientist?

1



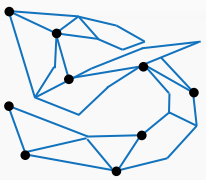
What is Data Scientist?



Analytical skills

Design algorithm

Business insight



Data Science Skills Spectrum

Capable to write Production Code
(Scala/Java/Go)

Big Data Techniques (HDFS, Spark)

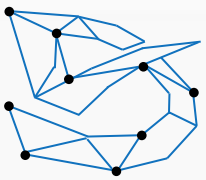
Machine Learning

Data Analytics (Python/R & SQL)

Dashboard / Presentation

• More engineer
focusing

More business
focusing



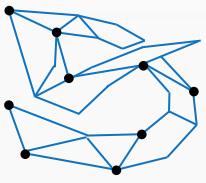
Recommended Stack for future DS

Data Science foundation

1. SQL – the universal language for analytics
2. Python – the most popular language for data science

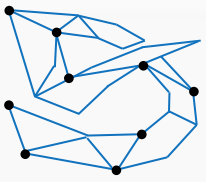
Analytics in the business world

1. Microsoft-Suite/G-Suite
2. Tableau (BI visualization)



Python in Data Science

2

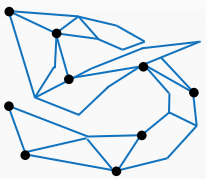


Why Python?



General purpose programming language

Purpose	
R focuses on better, user friendly data analysis, statistics and graphical models.	Python emphasizes productivity and code readability.
Used By?	
R has been used primarily in academics and research. However, R is rapidly expanding into the enterprise market. <i>"The closer you are to statistics, research and data science, the more you might prefer R."</i>	Python is used by programmers that want to delve into data analysis or apply statistical techniques, and by developers that turn to data science. <i>"The closer you are to working in an engineering environment, the more you might prefer Python."</i>



Life as a Data Scientist

“Ideal”

Diagonal
Line
Node



Face
Node

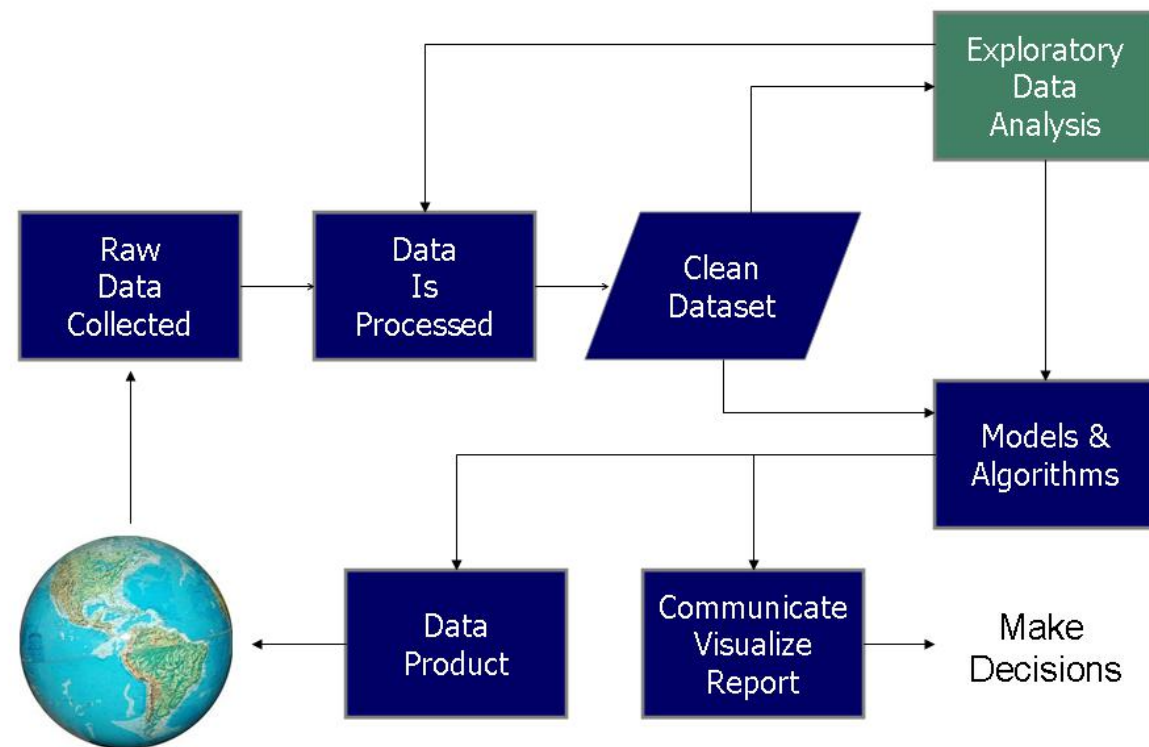


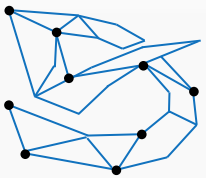
Cat
Node



“Reality”

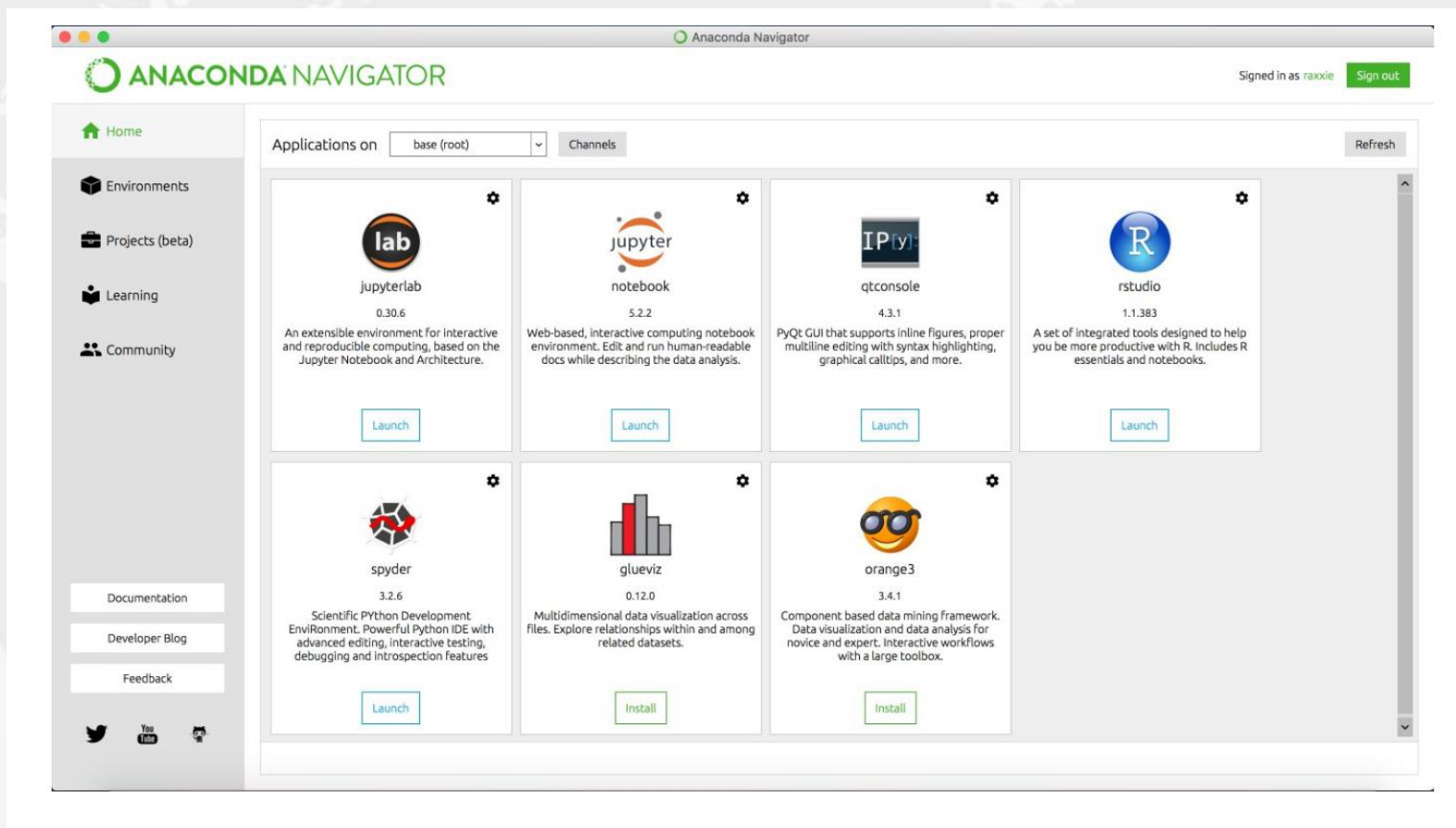
Data Science Process

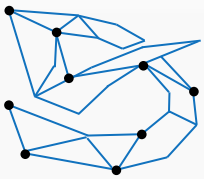




安装python

<https://www.anaconda.com/products/individual>





Jupyter VS Spyder

WIN-190815181010 - Jupyter

hub.gke.mybinder.org/user/hunters-forge-t-hunter-playbook-pn4e20u/notebooks/content/notebooks/windo... Guest

jupyter WIN-190815181010 (unsaved changes) Visit repo Copy Binder link

File Edit View Insert Cell Kernel Help Not Trusted Kernel

Remote Service creation

Metadata

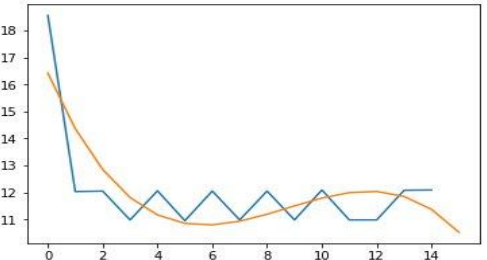
id	WIN-190815181010
author	Roberto Rodriguez @Cy3rWard0g
creation date	19/08/15
platform	Windows
playbook link	WIN-190813181020

Technical Description

Adversaries may execute a binary, command, or script via a method that interacts with Windows services, such as the Service Control Manager. This can be done by by adversaries creating a new service. Adversaries can create services remotely to execute code and move laterally across the environment.

Hypothesis

Adversaries might be creating new services remotely to execute code and move laterally in my environment



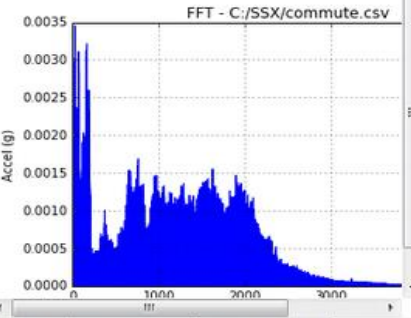
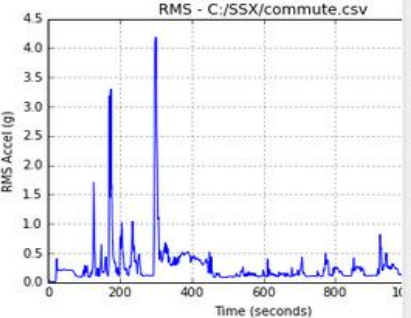
```
In [9]: 1 print( 'Next Month: %s' % str(X[len(X)-1]+1) )
      2 print( 'Next Estim: %.2f' % polyFun(X[len(X)-1]+1))
```

Next Month: 15
Next Estim: 10.52

Spyder (Python 3.5)

File Edit Search Source Run Debug Consoles Tools View Help

IPython console Console 1/A Editor - C:/SSX/Load_Plot_RMS_FFT4.py Variable explorer

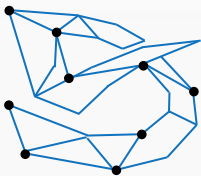


```
1 import matplotlib.pyplot as plt
2 import pandas as pd
3 import numpy as np
4 import tkinter as tk
5 from tkinter import filedialog
6 import time
7 import pyfftw
8
9 #Prompt user for file
10 root = tk.Tk()
11 root.withdraw()
12 file_path = filedialog.askopenfilename(filetype
13 print(file_path)
14
15 #Load Data
16 tic = time.clock()
17 df = pd.read_csv(file_path,delimiter=',',header
18 t = df["time"]
19 x = df["data"]
20 toc = time.clock()
21 print("Load Time:",toc-tic)
22
23 #Determine variables
24 N = np.int(np.prod(t.shape))#Length of the arra
25 Fs = 1/(t[1]-t[0]) #sample rate (Hz)
26 T = 1/Fs;
27 print("# Samples:",N)
28
29 #Plot Data
30 tic = time.clock()
31 plt.figure(1)
32 plt.plot(t, x)
33 plt.xlabel('Time (seconds)')
34 plt.ylabel('Accel (g)')
35 plt.title(file_path)
36 plt.grid()
37 toc = time.clock()
38 print("Plot Time:",toc-tic)
39
```

Name	Type	Size	Value
df	DataF...	(1155, 1)	Co1...
t	Series	(11553414, 2)	0 ...
x	Series	(11553414,)	0 ...
yf	com...	1	1 ...
tic	float	(5776708,)	13...
toc	float	1	15...
a	float32	(11553414,)	arr...
Fs	float64	1	100...
T	float64	(5776708,)	0.0...
t_RMS	float64	1	arr...
x_RMS	float64	1	arr...
xf	float64	1	arr...
N	int	1	115...
i	int	(1155, 1)	1154
w	int	1	100...
steps	int32	1	1155
file_path	str	(11553414,)	C:/...

Variable explorer History log File explorer

Permissions: RM End-of-lines: CRLF Encoding: UTF-8-GUESSED Line: 14 Column: 1 Memory: 85 %

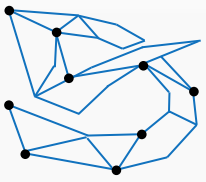


Data type

Data Structure	Ordered	Mutable	Constructor	Example
List	Yes	Yes	<code>[]</code> or <code>list()</code>	<code>[5.7, 4, 'yes', 5.7]</code>
Tuple	Yes	No	<code>()</code> or <code>tuple()</code>	<code>(5.7, 4, 'yes', 5.7)</code>
Set	No	Yes	<code>{}</code> * or <code>set()</code>	<code>{5.7, 4, 'yes'}</code>
Dictionary	No	Yes**	<code>{}</code> or <code>dict()</code>	<code>{'Jun': 75, 'Jul': 89}</code>

	列表	元组	集合	字典
英文	list	tuple	set	dict
可否读写	读写	只读	读写	读写
可否重复	是	是	否	是
存储方式	值	值	键(不能重复)	键值对(键不能重复)
是否有序	有序	有序	无序	无序, 自动正序
初始化	<code>[1, 'a']</code>	<code>('a', 1)</code>	<code>set([1,2])</code> 或 <code>{1,2}</code>	<code>{'a':1, 'b':2}</code>
添加	<code>append</code>	只读	<code>add</code>	<code>d['key'] = 'value'</code>
读元素	<code>l[2:]</code>	<code>t[0]</code>	无	<code>d['a']</code>

<https://blog.csdn.net/baoguaalalei>



Pandas: Panel Data System

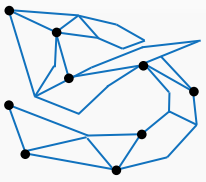
Rich data structures and functions to work with structured data fast, easy and expressive

Build on top of NumPy

Ideal tool for:
munging/cleaning/analyzing/modeling data

PANel DAta S





Quiz#1

Python

>>>

```
>>> colors = ["red", "green", "burnt sienna", "blue"]  
>>> colors[2]
```

What is the output of the `colors[2]` expression?

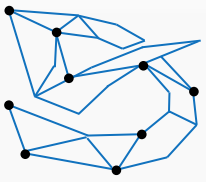
☒ It causes a run-time error.

☐ 'green'

☐ 'burnt sienna'

☐ 'red'

☐ 'blue'



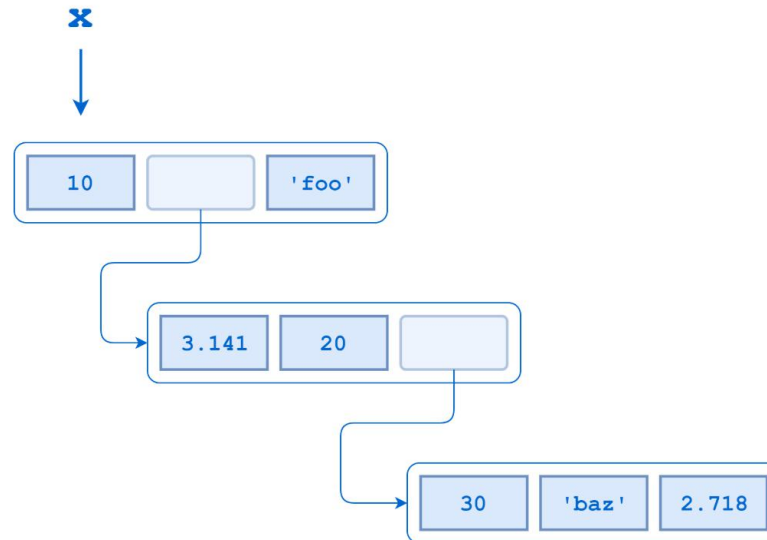
Quiz #2

Consider the following nested list definition:

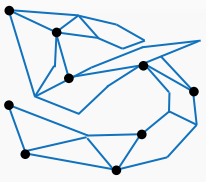
Python

```
x = [10, [3.141, 20, [30, 'baz', 2.718]], 'foo']
```

A schematic for this list is shown below:



What is the expression that returns the 'z' in 'baz' ?



Quiz #2 - Answer

Python

```
x = [10, [3.141, 20, [30, 'baz', 2.718]], 'foo']
```

Expression

Selects

`x[1]`

The second element of `x`:

`x[-2]`

`[3.141, 20, [30, 'baz', 2.718]]`

`x[1][2]`

The third element of that sublist:

`x[1][-1]`

`[30, 'baz', 2.718]`

`x[1][2][1]`

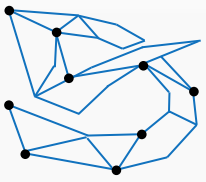
The second element of that sublist: `'baz'`

`x[1][2][-2]`

`x[1][2][1][2]`

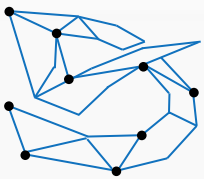
The third character of `'baz'`: `'z'`

`x[1][2][1][-1]`



Regular Expression

Identifiers	Modifiers	White space characters	Escape required
\d= any number (a digit)	\d represents a digit.Ex: \d{1,5} it will declare digit between 1,5 like 424,444,545 etc.	\n = new line	. + * ? [] \$ ^ () { } \
\D= anything but a number (a non-digit)	+ = matches 1 or more	\s= space	
\s = space (tab,space,newline etc.)	? = matches 0 or 1	\t =tab	
\S= anything but a space	* = 0 or more	\e = escape	
\w = letters (Match alphanumeric character, including "_")	\$ match end of a string	\r = carriage return	
\W =anything but letters (Matches a non-alphanumeric character excluding "_")	^ match start of a string	\f = form feed	
. = anything but letters (periods)	matches either or x/y	----- ---	
\b = any character except for new line	[] = range or "variance"	----- --	
\.	{x} = this amount of preceding code	----- ---	



Data Reshaping for Single Table

Group by
Aggregation
Pivot
Pivot Table
Stack
Unstack
Melt

Pivot

df

	foo	bar	baz	zoo
0	one	A	1	x
1	one	B	2	y
2	one	C	3	z
3	two	A	4	q
4	two	B	5	w
5	two	C	6	t

df.pivot(index='foo', columns='bar', values='baz')

bar	A	B	C
foo			
one	1	2	3
two	4	5	6

Melt

df3

	first	last	height	weight
0	John	Doe	5.5	130
1	Mary	Bo	6.0	150

df3.melt(id_vars=['first', 'last'])

	first	last	variable	value
0	John	Doe	height	5.5
1	Mary	Bo	height	6.0
2	John	Doe	weight	130
3	Mary	Bo	weight	150

Stack

df2

		A	B
first	second		
bar	one	1	2
	two	3	4
baz	one	5	6
	two	7	8

stacked = df2.stack()

first	second		
bar	one	A	1
		B	2
	two	A	3
		B	4
baz	one	A	5
		B	6
	two	A	7
		B	8

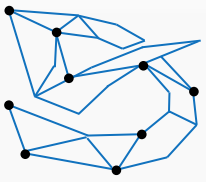
Unstack

stacked

first	second		
bar	one	A	1
		B	2
	two	A	3
		B	4
baz	one	A	5
		B	6
	two	A	7
		B	8

stacked.unstack()

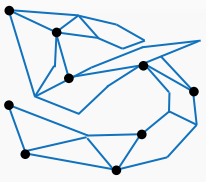
		A	B
first	second		
bar	one	1	2
	two	3	4
baz	one	5	6
	two	7	8



Quiz # 3

```
exp = lambda x: x ** 3  
print(exp(2))
```

- A. 6
- B. 222
- C. 8
- D. None of the above



Quiz#4

```
def func(message, num = 1):  
    print(message * num)
```

```
func('Welcome')
```

```
func('Viewers', 3)
```

A. Welcome

Viewers

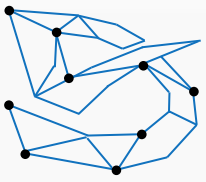
B. Welcome

ViewersViewersViewers

C. Welcome

Viewers,Viewers,Viewers

D. Welcome



Quiz #5

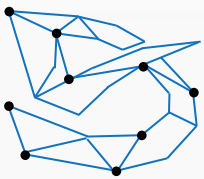
Q.10 Which of the following can be used to make a Dataframe?

☐ Series

☐ DataFrame

☐ Structured ndarray

☐ All of the above



Data Operation for Multiple Tables

Append
Concat
Join/merge

df1					Result				
	A	B	C	D		A	B	C	D
0	A0	B0	C0	D0	0	A0	B0	C0	D0
1	A1	B1	C1	D1	1	A1	B1	C1	D1
2	A2	B2	C2	D2	2	A2	B2	C2	D2
3	A3	B3	C3	D3	3	A3	B3	C3	D3

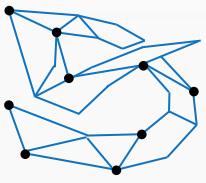
df2					Result				
	A	B	C	D		A	B	C	D
4	A4	B4	C4	D4	4	A4	B4	C4	D4
5	A5	B5	C5	D5	5	A5	B5	C5	D5
6	A6	B6	C6	D6	6	A6	B6	C6	D6
7	A7	B7	C7	D7	7	A7	B7	C7	D7

df3					Result				
	A	B	C	D		A	B	C	D
8	A8	B8	C8	D8	8	A8	B8	C8	D8
9	A9	B9	C9	D9	9	A9	B9	C9	D9
10	A10	B10	C10	D10	10	A10	B10	C10	D10
11	A11	B11	C11	D11	11	A11	B11	C11	D11

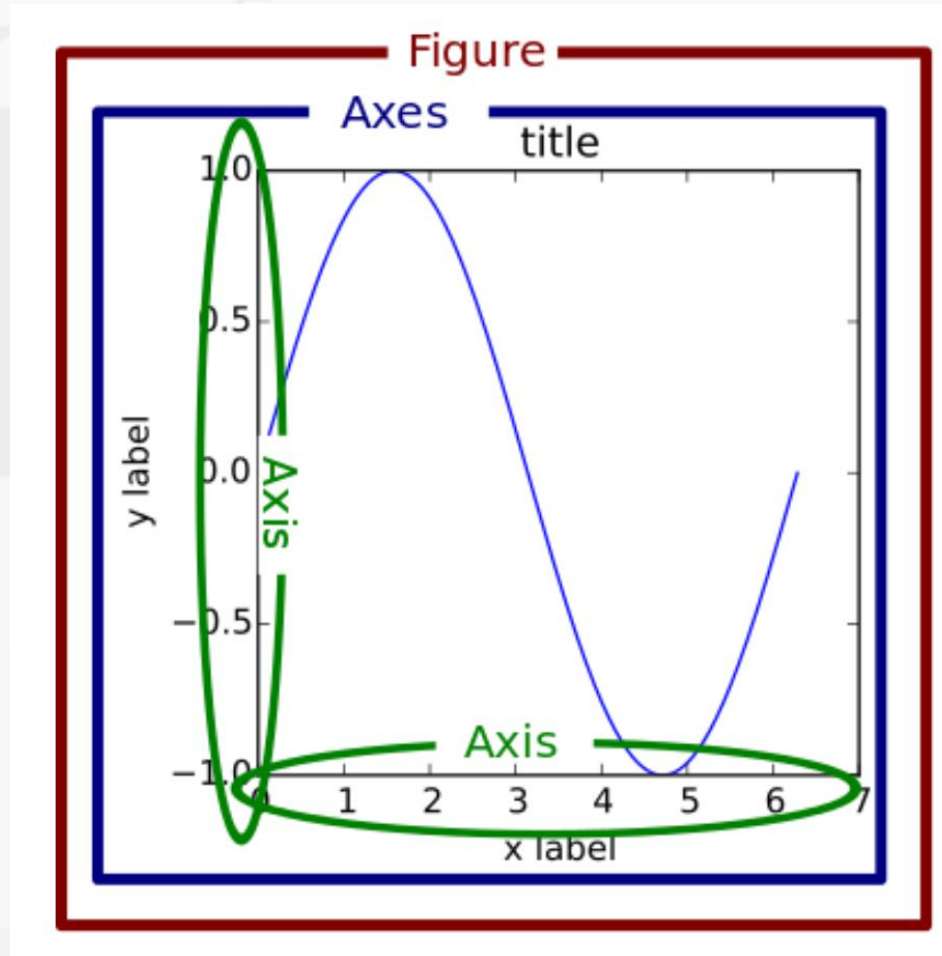
df1					df4				Result							
	A	B	C	D		B	D	F		A	B	C	D	B	D	F
0	A0	B0	C0	D0	2	B2	D2	F2	2	A2	B2	C2	D2	B2	D2	F2
1	A1	B1	C1	D1	3	B3	D3	F3	3	A3	B3	C3	D3	B3	D3	F3
2	A2	B2	C2	D2	6	B6	D6	F6								
3	A3	B3	C3	D3	7	B7	D7	F7								

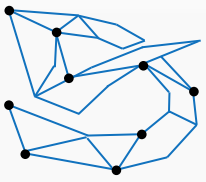
df1					df4				Result							
	A	B	C	D		B	D	F		A	B	C	D	B	D	F
0	A0	B0	C0	D0	2	B2	D2	F2	0	A0	B0	C0	D0	NaN	NaN	NaN
1	A1	B1	C1	D1	3	B3	D3	F3	1	A1	B1	C1	D1	NaN	NaN	NaN
2	A2	B2	C2	D2	6	B6	D6	F6	2	A2	B2	C2	D2	B2	D2	F2
3	A3	B3	C3	D3	7	B7	D7	F7	3	A3	B3	C3	D3	B3	D3	F3

df1					s1		Result					
	A	B	C	D		X		A	B	C	D	X
0	A0	B0	C0	D0	0	X0	0	A0	B0	C0	D0	X0
1	A1	B1	C1	D1	1	X1	1	A1	B1	C1	D1	X1
2	A2	B2	C2	D2	2	X2	2	A2	B2	C2	D2	X2
3	A3	B3	C3	D3	3	X3	3	A3	B3	C3	D3	X3



Matplotlib Plots





Seaborn Plot

Example gallery

