Journey PPT

Shell Training Bootcamp (August 14 to September 20, 6 weeks)

Omeir Fawaz

Associate Data Engineer

INTERNAL

Safety and Health On Site

Ensure confidential discussions are not overheard

Make sure your workspace is ergonomically sound

Ensure adequate lighting in the room when you work

Clean surfaces frequently

Have an emergency and evacuation plan in place

Ensure understanding of fire safety

- Know what the fire alarm sounds like
- Make sure that you can hear the fire alarm
- Make sure your smoke alarms work
- Maintain clear walkways and fire exits

On the Move

Do not take this call,
or any other call,
while driving – ever

Do not use any hands-free device – Bluetooth, built-in, etc. – whilst driving

Continue to follow COVID guidelines

In the event of any kind of emergency, please leave the call – promptly and safely

Week 6

(Sep 19 – Sep 20)

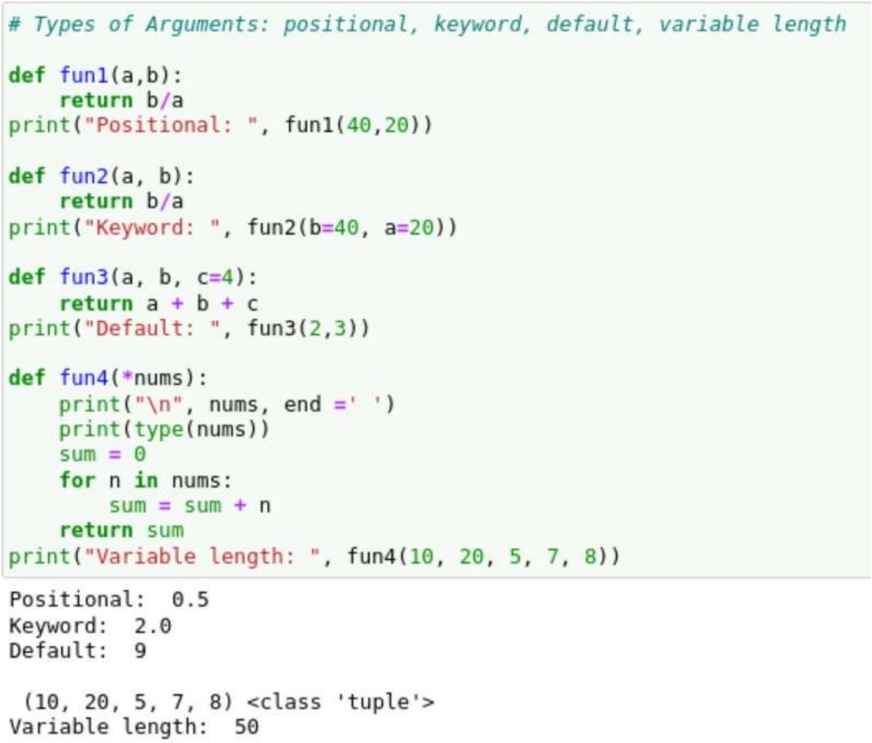
01 Python

02 Pandas and Matplotlib

6.1 Python and Jupyter Notebook

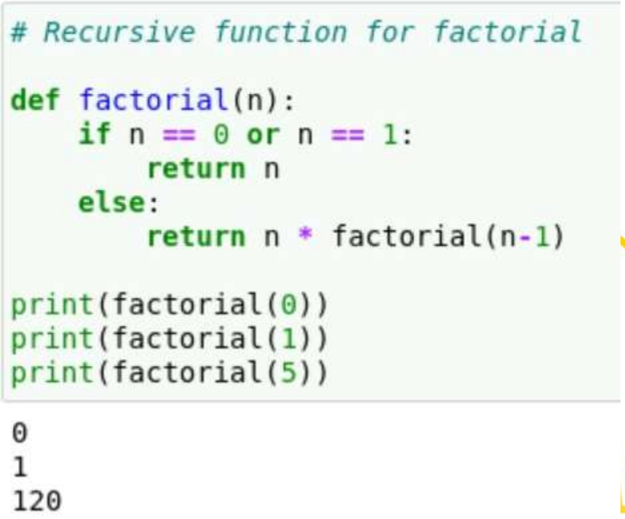
Learnt about the following:

1. Functions: built-in and user-defined; classes
2. Types of arguments: positional, keyword, default, and variable length
3. Sequence generators, decorators, recursive functions



```
# Types of Arguments: positional, keyword, default, variable length

def fun1(a,b):
    return b/a
print("Positional: ", fun1(40,20))

def fun2(a, b):
    return b/a
print("Keyword: ", fun2(b=40, a=20))

def fun3(a, b, c=4):
    return a + b + c
print("Default: ", fun3(2,3))

def fun4(*nums):
    print("\n", nums, end =' ')
    print(type(nums))
    sum = 0
    for n in nums:
        sum = sum + n
    return sum
print("Variable length: ", fun4(10, 20, 5, 7, 8))
```

```
Positional:  0.5
Keyword:  2.0
Default:  9

 (10, 20, 5, 7, 8) <class 'tuple'>
Variable length:  50
```



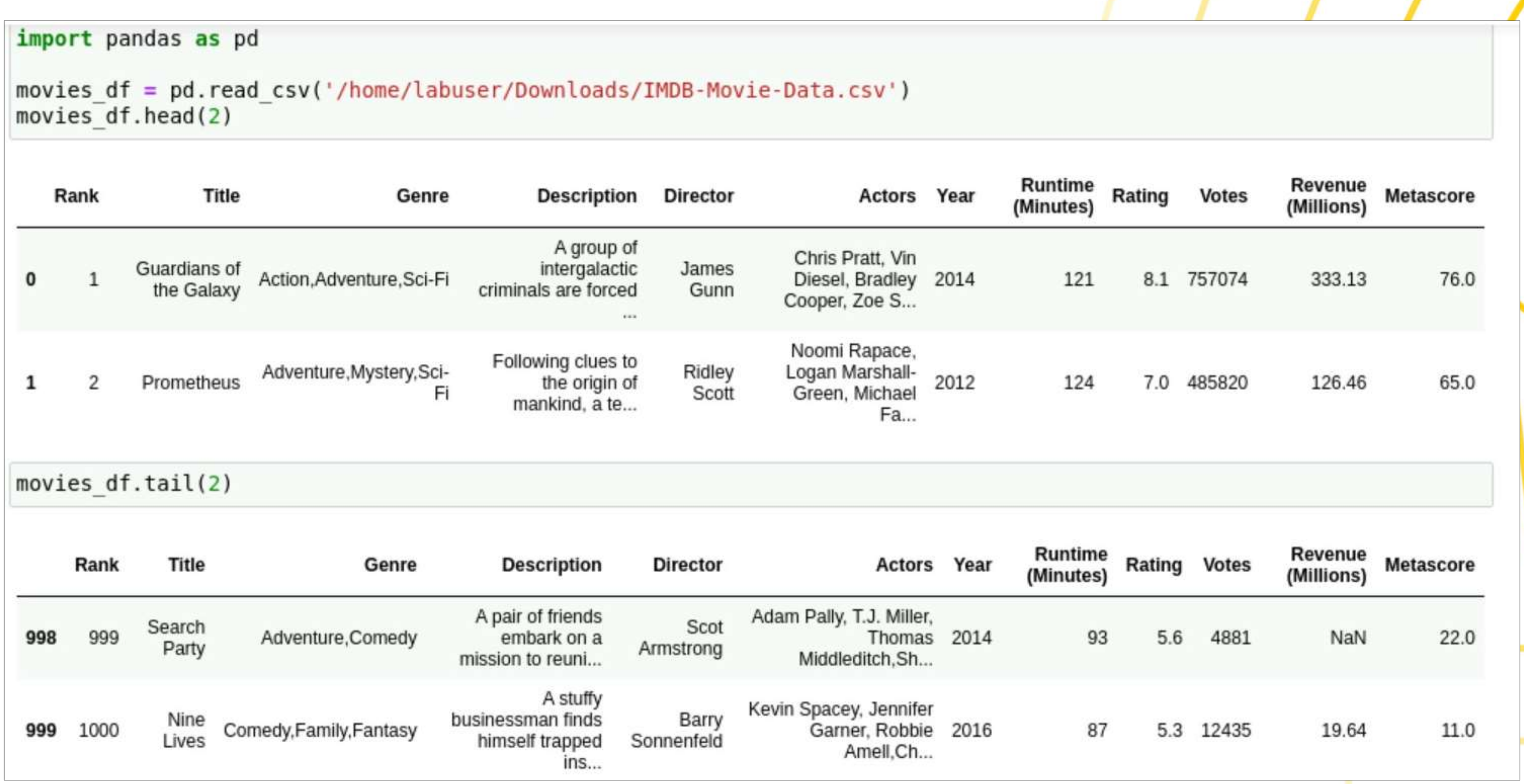
```
# Recursive function for factorial

def factorial(n):
    if n == 0 or n == 1:
        return n
    else:
        return n * factorial(n-1)

print(factorial(0))
print(factorial(1))
print(factorial(5))
```

```
0
1
120
```

6.2 Pandas

Learnt about the following:

1. Dataframes, operations on columns
2. Matplotlib visualisations, dynamic visualisations

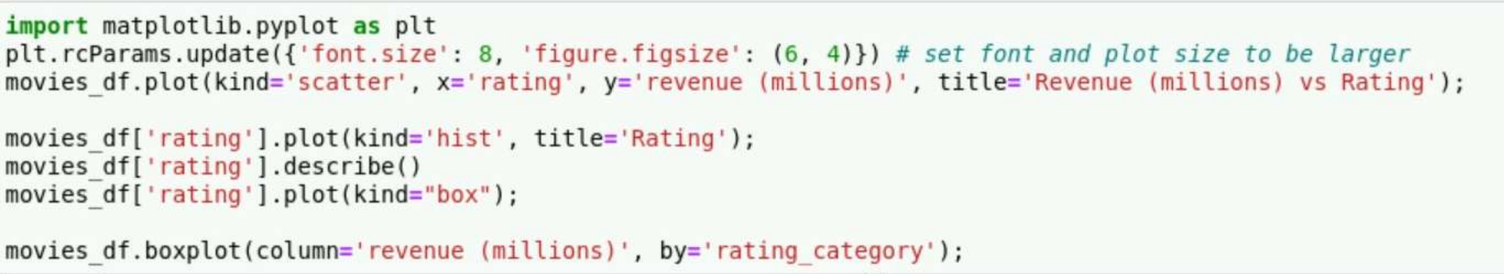


```
import pandas as pd

movies_df = pd.read_csv('/home/labuser/Downloads/IMDB-Movie-Data.csv')
movies_df.head(2)
```

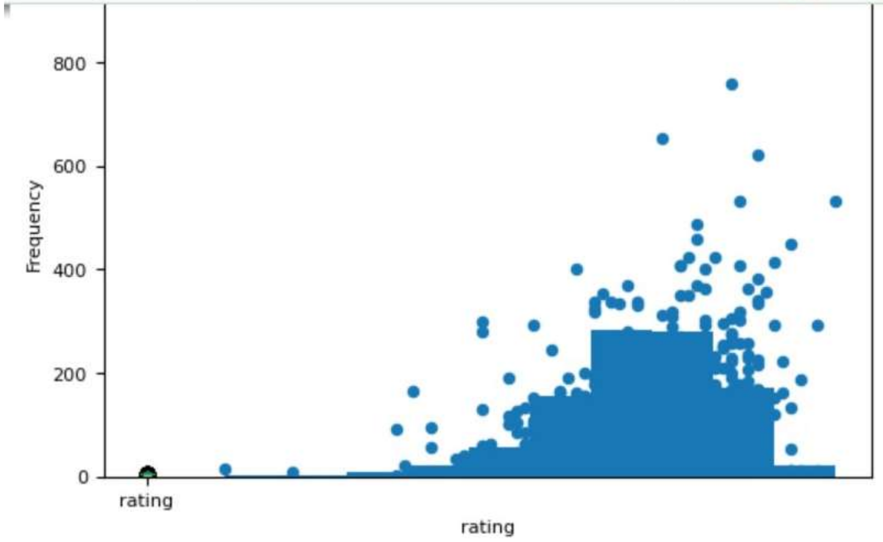
	Rank	Title	Genre	Description	Director	Actors	Year	Runtime (Minutes)	Rating	Votes	Revenue (Millions)	Metascore
0	1	Guardians of the Galaxy	Action,Adventure,Sci-Fi	A group of intergalactic criminals are forced ...	James Gunn	Chris Pratt, Vin Diesel, Bradley Cooper, Zoe S...	2014	121	8.1	757074	333.13	76.0
1	2	Prometheus	Adventure,Mystery,Sci-Fi	Following clues to the origin of mankind, a te...	Ridley Scott	Noomi Rapace, Logan Marshall-Green, Michael Fa...	2012	124	7.0	485820	126.46	65.0

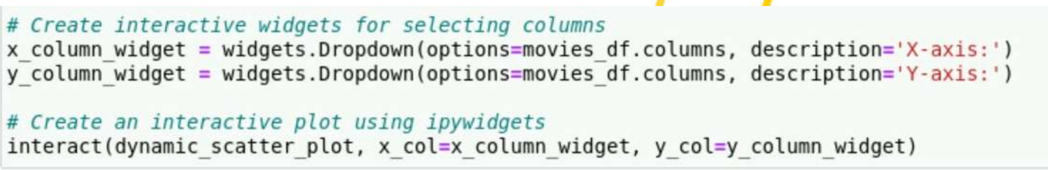
```
movies_df.tail(2)
```

	Rank	Title	Genre	Description	Director	Actors	Year	Runtime (Minutes)	Rating	Votes	Revenue (Millions)	Metascore
998	999	Search Party	Adventure,Comedy	A pair of friends embark on a mission to reuni...	Scot Armstrong	Adam Pally, T.J. Miller, Thomas Middleditch,Sh...	2014	93	5.6	4881	NaN	22.0
999	1000	Nine Lives	Comedy,Family,Fantasy	A stuffy businessman finds himself trapped ins...	Barry Sonnenfeld	Kevin Spacey, Jennifer Garner, Robbie Amell,Ch...	2016	87	5.3	12435	19.64	11.0

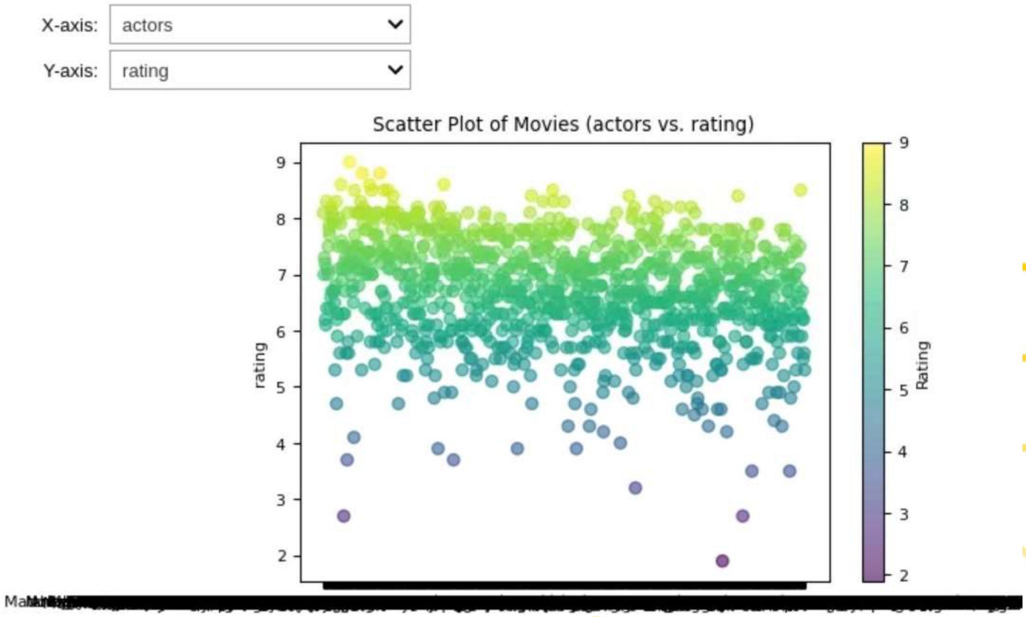


```
import matplotlib.pyplot as plt
plt.rcParams.update({'font.size': 8, 'figure.figsize': (6, 4)}) # set font and plot size to be larger
movies_df.plot(kind='scatter', x='rating', y='revenue (millions)', title='Revenue (millions) vs Rating');

movies_df['rating'].plot(kind='hist', title='Rating');
movies_df['rating'].describe()
movies_df['rating'].plot(kind="box");

movies_df.boxplot(column='revenue (millions)', by='rating_category');
```





```
# Create interactive widgets for selecting columns
x_column_widget = widgets.Dropdown(options=movies_df.columns, description='X-axis:')
y_column_widget = widgets.Dropdown(options=movies_df.columns, description='Y-axis:')

# Create an interactive plot using ipywidgets
interact(dynamic_scatter_plot, x_col=x_column_widget, y_col=y_column_widget)
```



X-axis: actors

Y-axis: rating