

Data Science Tools and Ecosystem

Introduction :

In this notebook, Data Science Tools and Ecosystem are summarized.

Exercise 1: Create a Jupyter Notebook

Ans : Created a notebook called "**DataScienceEcosystem.ipynb**" Jupyter Notebook file.

DONE!!!

Exercise 2: Create markdown cell with title of the notebook

Ans : The title "**Data Science Tools and Ecosystem**" is clearly visible.

DONE!!!

Exercise 3 - Create a markdown cell for an introduction

Ans : Created an introductory sentence about the notebook such as follow: "In this notebook, Data Science Tools and Ecosystem are summarized."

DONE!!!

Exercise 4 - Create a markdown cell to list data science languages

Ans : Some of the popular languages that Data Scientists uses are follow in an ordered List:

1. Python
2. SQL
3. R
4. Julia
5. JavaScript
6. Scala
7. Java
8. Go
9. MATLAB
10. C/C++
11. Statistical Analytical System (SAS)

DONE!!!

Exercise 5 - Create a markdown cell to list data science libraries

Ans : Some of the commonly used libraries used by Data Scientists include:

1. Numpy
2. Pandas
3. Matplotlib

DONE!!!

Exercise 6 - Create a markdown cell with a table of Data Science tools

Ans : Create a single column table in this cell with the first row containing the header Data Science Tools. The subsequent three rows in the table should indicate three development environment open source tools used in data science.

Data Science Tools
Jupyter Notebook
PowerBI Tool
Intel OpenVINO Tool
R Studio
Spider

DONE!!!

Exercise 7 - Create a markdown cell introducing arithmetic expression examples

Ans : Below are a few examples of evaluating arithmetic expressions in Python

```
In [1]: # Arithmetic Operations

# Addition using add(), Substraction using sub(),
# Multiplication using mul(), Division using div(),
# floor division using floor_div() and modulus using mod() functions.

def add(n1, n2):
    return (n1 + n2)

def sub(n1, n2):
    return (n1 - n2)

def mul(n1, n2):
    return (n1 * n2)
```

```
def div(n1, n2):
    return (n1 / n2)

def floor_div(n1, n2):
    return (n1 // n2)

def mod(n1, n2):
    return (n1 % n2)

# inputs
n1 = 20
n2 = 10

# printing the values

print(f"After Addition the result of {n1}, and {n2} is {add(n1, n2)}")
print(f"After Substraction the result of {n1}, and {n2} is {sub(n1, n2)}")
print(f"After Multiplication the result of {n1}, and {n2} is {mul(n1, n2)}")
print(f"After Division the result of {n1}, and {n2} is {div(n1, n2)}")
print(f"After Floor Division the result of {n1}, and {n2} is {floor_div(n1, n2)}")
print(f"After Modulus the result of {n1}, and {n2} is {mod(n1, n2)}")
```

```
After Addition the result of 20, and 10 is 30
After Substraction the result of 20, and 10 is 10
After Multiplication the result of 20, and 10 is 200
After Division the result of 20, and 10 is 2.0
After Floor Division the result of 20, and 10 is 2
After Modulus the result of 20, and 10 is 0
```

DONE!!!

Exercise 8 - Create a code cell to multiply and add numbers

Ans : In this code cell evaluate the expression

$(3*4)+5$

Insert a comment line before the expression to explain the operation e.g. This a simple arithmetic expression to mutiply then add integers. Then execute the cell to ensure the expression returns the expected output of **17**.

```
In [2]: def mul_add(mul_var1, mul_var2, add_var):
        return ((mul_var1 * mul_var2) + add_var)

# input
mul_var1 = 3
mul_var2 = 4
add_var = 5

print(f"The value of the expression (3 * 4) + 5 is {mul_add(mul_var1, mul_var2, add
```

The value of the expression (3 * 4) + 5 is 17

DONE!!!

Exercise 9 - Create a code cell to convert minutes to hours

Ans : In this code cell write an expression that converts 200 minutes into hours. Insert a comment line before the expression to explain the operation e.g. This will convert 200 minutes to hours by dividing by 60.

```
In [3]: def convert_minutes_into_hours(minutes):  
        return (minutes / 60)  
  
minutes = 200  
  
print(f"{minutes} minutes means it is {convert_minutes_into_hours(minutes)} hours o  
200 minutes means it is 3.3333333333333335 hours or ~ 3 hours approx.
```

DONE!!!

Exercise 10 - Insert a markdown cell to list Objectives

Below the introduction cell created in Exercise 3, insert a new markdown cell to list the objectives that this notebook covered (i.e. some of the key takeaways from the course). In this new cell start with an introductory line titled: Objectives: in bold font. Then using an unordered list (bullets) indicate 3 to 5 items covered in this notebook, such as List popular languages for Data Science.

List popular languages for Data Science

- Python
- SQL
- R
- C/C++
- Statistical Analytical System (SAS)

DONE!!!

Exercise 11 - Create a markdown cell to indicate the Author's name

Ans :

Author :

Arpit Dubey

Email : aarpitdubey@gmail.com

Github : <https://github.com/aarpitdubey>

LinkedIn : <https://www.linkedin.com/in/aarpitdubey/>

DONE!!!

Exercise 12 - Share your notebook through GitHub

Upload your notebook to a public repository on GitHub.

Note : Please keep the GitHub repo link of the notebook handy. You will need to submit this link as a part of the assignment evaluation.

Ans : https://github.com/aarpitdubey/tools_for_datascience_peer_graded_assignment.git

DONE!!!

Exercise 13 -Take a screenshot of the first page of the notebook and save it as 1-notebook.png(Images can be saved with either the .jpg or .png extension.)

Ans :

In []:

In []: