## ▼ JSON data in Python

## JSON stands for JavaScript Object Notation

- Popular data format used for representing structured data
- Common format while transmiting and receiving data between a server and web application

In python, JSON exists as a string or file containing JSON object

```
#To work with JSON import json module
import json
import pandas as pd
#JSON exists as a string
p='{"name":"Ram","contact":[9123412345,7123471234]}'
d=json.loads(p)
print(d)
print(type(d))
     {'name': 'Ram', 'contact': [9123412345, 7123471234]}
     <class 'dict'>
#To read a file containing json object
f=open("/content/fruit.json","r+")
data=json.load(f)
print(data)
print(type(data))
     {'fruit': 'Apple', 'size': 'Large', 'color': 'Red'}
     <class 'dict'>
import pandas as pd
df2 = pd.DataFrame.from dict(data, orient="index")
print(df2)
     fruit Apple
     size
            Large
              Red
     color
#To read a file containing json object
f=open("/content/sampledata.json","r+")
data=json.load(f)
print(data)
print(type(data))
df=pd.DataFrame(data)
df
```

{'Duration': {'0': 60, '1': 60, '2': 60, '3': 45, '4': 45, '5': 60, '6': 60, '7': 45, '8 <class 'dict'>

Duration	Pulse	Maxpulse	Calories
60	110	130	409.1
60	117	145	479.0
60	103	135	340.0
45	109	175	282.4
45	117	148	406.0
60	105	140	290.8
60	110	145	300.4
60	115	145	310.2
75	120	150	320.4
75	125	150	330.4
	60 60 45 45  60 60 60	60 110 60 117 60 103 45 109 45 117  60 105 60 110 60 115 75 120	60 110 130 60 117 145 60 103 135 45 109 175 45 117 148  60 105 140 60 110 145 60 115 145 75 120 150

169 rows × 4 columns

```
df1 = pd.read_json('/content/sampledata.json')
print(df)
```

	Duration	Pulse	Maxpulse	Calories
0	60	110	130	409.1
1	60	117	145	479.0
2	60	103	135	340.0
3	45	109	175	282.4
4	45	117	148	406.0
• •	• • •	• • •	• • •	
164	60	105	140	290.8
165	60	110	145	300.4
166	60	115	145	310.2
167	75	120	150	320.4
168	75	125	150	330.4

[169 rows x 4 columns]