Day 12- Python JSON

• Python has a built-in package called json, which can be used to work with JSON data

```
Import json
```

• If you have a JSON string, you can parse it by using the json.loads() method

In [1]:
```
import json
# some JSON:
x =  '{ "name":"Sachin", "age":18, "city":"Mumbai"}'
# parse x:
y = json.loads(x)
# the result is a Python dictionary:
print(y["age"])
```

```
18
```

• If you have a Python object, you can convert it into a JSON string by using the json.dumps() method.

In [2]:
```
import json
# a Python object (dict):
x = {
  "name": "John",
  "age": 30,
  "city": "New York"
}

# convert into JSON:
y = json.dumps(x)
# the result is a JSON string:
print(y)
```

```
{"name": "John", "age": 30, "city": "New York"}
```

• You can convert Python objects of the following types, into JSON strings:

- dict
- list
- tuple
- string
- int
- float
- True
- False
- None
- Use the indent parameter to define the numbers of indents

• You can also define the separators, default value is (", ", ": "), which means using a comma and a space to separate each object, and a colon and a space to separate keys from values

• json.dumps(x, indent=4, separators=(". ", " = "))

• The json.dumps() method has parameters to order the keys in the result

• json.dumps(x, indent=4, sort_keys=True)

Exercise:

In [1]:
```
import json
dict1 = {'name':"Rahul",'age':'18','city':"Delhi"}
list1 = ['a','b','c','d','e']
tuple1 = (1,2,'Apple',3)
string1 = "Sunny Day"
int1 = 2
float1 = 5.66
True1 = True
False1 = False
None1 = None

dict2 = json.dumps(dict1)
list2 = json.dumps(list1)
tuple2 = json.dumps(tuple1)
string2 = json.dumps(string1)
int2 = json.dumps(int1)
float2 = json.dumps(float1)
True2 = json.dumps(True1)
False2 = json.dumps(False1)
None2 = json.dumps(None1)

print(dict2)
print(list2)
print(tuple2)
print(string2)
print(int2)
print(float2)
print(True2)
print(False2)
print(None2)
```

```
{"name": "Rahul", "age": "18", "city": "Delhi"}
["a", "b", "c", "d", "e"]
[1, 2, "Apple", 3]
"Sunny Day"
2
5.66
true
false
null
```

In [ ]:
```
import json
x={
    "firstName": "Jane",
    "lastName": "Doe",
    "hobbies": ["running", "sky diving", "singing"],
    "age": 35,
    "TakenTest":True,
    "average":70.99

},{
    "firstName" : "Rosh",
    "lastName" : "Vartin",
    "hobbies" : ["running", "Drawing", "dancing"],
    "age" : 39,
    "TakenTest":True,
    "average":66.99
}
with open("file.json","w") as f:
    json.dump(x,f,indent=4)


from pymongo import MongoClient
myclient=MongoClient("mongodb://localhost:27017/")
db=myclient["mongdb"]
Collection=db["info"]
with open("file.json") as f:
    data=json.load(f)
if isinstance(data,list):
    Collection.insert_many(data)
else:
    Collection.insert_one(data)
```

Completed Day 12's notes & exercises

THANK YOU!

Check out My Repository at https://github.com/AakankshaJarode/BestEnlist_Python_Internship.git (https://github.com/AakankshaJarode/BestEnlist_Python_Internship.git)

Chech out My LinkedIn Page at https://www.linkedin.com/in/aakanksha-jarode-1b0195179 (https://www.linkedin.com/in/aakanksha-jarode-1b0195179)