Read JSON data from a file and convert it into dict using json.load()

- Using a json.load() method, we can read JSON data from text, JSON, or binary file.
- The json.load() method returns data in the form of a Python dictionary.
- Later we use this dictionary to access and manipulate data in our application or system.

Now, let's see the example. For this example, I am reading the "developer.json" file present on my hard drive.

This file contains the following JSON data.

developer.js

```
{
   "name": "jane doe",
   "salary": 9000,
   "skills": [
      "Raspberry pi",
      "Machine Learning",
      "Web Development"
   ],
   "email": "JaneDoe@pynative.com",
   "projects": [
      "Python Data Mining",
      "Python Data Science"
   ]
}
```

developer.py

```
import json
print("Started Reading JSON file")
with open("developer.json", "r") as read_file:
   print("Converting JSON encoded data into Python dictionary")
   developer = json.load(read_file)
   print("Decoded JSON Data From File")
```

```
    for key, value in developer.items():
        print(key, ":", value)
    print("Done reading json file")
```

Output:

```
Started Reading JSON file
Converting JSON encoded data into Python dictionary
Decoded JSON Data From File
name : jane doe
salary : 9000
skills : ['Raspberry pi', 'Machine Learning', 'Web Development']
email : JaneDoe@pynative.com
projects : ['Python Data Mining', 'Python Data Science']
Done reading json file
```

Access JSON data directly using key name:

Use the following code If you want to access the JSON key directly instead of iterating the entire JSON from a file

developer.py

```
import json
print("Started Reading JSON file")
with open("developer.json", "r") as read_file:
    print("Converting JSON encoded data into Python dictionary")
    developer = json.load(read_file)
    print("Decoding JSON Data From File")
    print("Printing JSON values using key")
    print(developer["name"])
    print(developer["salary"])
    print(developer["skills"])
```

```
print(developer["email"])
print("Done reading json file")
```

Output:

```
Started Reading JSON file
Converting JSON encoded data into Python dictionary
Decoding JSON Data From File
Printing JSON values using key
jane doe
9000
['Raspberry pi', 'Machine Learning', 'Web Development']
JaneDoe@pynative.com
Done reading json file
```

Note : You can read the JSON data from text, json, or a binary file using the same way mentioned above.

Convert JSON String to Python dictionary using json.loads():

- Sometimes we receive JSON response in string format.
- So to use it in our application, we need to convert JSON string into a Python dictionary.
- Using the json.loads() method, we can deserialize native String, byte, or bytearray instance containing a JSON document to a Python dictionary.
- We can refer to the conversion table mentioned at the start of an article.

Example:

```
import json
developerJsonString = """{
   "name": "jane doe",
   "salary": 9000,
   "skills": [
      "Raspberry pi",
```

```
        "Machine Learning",
        "Web Development"
    ],
    "email": "JaneDoe@pynative.com",
    "projects": [
        "Python Data Mining",
        "Python Data Science"
    ]
}
"""
print("Started converting JSON string document to Python dictionary")
developerDict = json.loads(developerJsonString)
print("Printing key and value")
print(developerDict["name"])
print(developerDict["salary"])
print(developerDict["skills"])
print(developerDict["email"])
print(developerDict["projects"])
print("Done converting JSON string document to a dictionary")
```

Output:

```
Started converting JSON string document to Python dictionary
Printing key and value
jane doe
9000
['Raspberry pi', 'Machine Learning', 'Web Development']
JaneDoe@pynative.com
['Python Data Mining', 'Python Data Science']
```

Done converting JSON string document to a dictionary

Parse and Retrieve nested JSON array key-values:

Let’s assume that you’ve got a JSON response that looks like this:

```
developerInfo = """{
   "id": 23,
   "name": "jane doe",
   "salary": 9000,
   "email": "JaneDoe@pynative.com",
   "experience": {"python":5, "data Science":2},
   "projectinfo": [{"id":100, "name":"Data Mining"}]
}
"""
```

For example, You want to retrieve the project name from the developer info JSON array to get to know on which project he/she is working.

Let’s see now how to read nested JSON array key-values.

In this example, we are using a developer info JSON array, which has project info and experience as nested JSON data.

Code:

```
import json
print("Started reading nested JSON array")
developerDict = json.loads(developerInfo)
print("Project name: ", developerDict["projectinfo"][0]["name"])
print("Experience: ", developerDict["experience"]["python"])
print("Done reading nested JSON Array")
```

Output:

```
Started reading nested JSON array
```

Project name: Data Mining

Experience: 5

Done reading nested JSON Array