

JSON In Python

What is JSON?

- **JSON** stands for **JavaScript Object Notation**.
- It is a lightweight format for storing and exchanging data.
- Python can work with JSON easily using the built-in `json` module.

Why JSON?

- Used in **APIs**, **websites**, **mobile apps** to transfer data.
- Human-readable and machine-readable.
- Works just like Python dictionaries (key-value pairs).

Importing JSON Module:

```
import json
```

Converting Python → JSON (Serialization / Dumping):

```
import json
```

```
data = {"name": "Ali", "age": 20, "city": "Karachi"}  
json_string = json.dumps(data)  
print(json_string)
```

Converting JSON → Python (Deserialization / Loading):

```
import json
```

```
json_data = '{"name": "Sara", "age": 22, "city": "Lahore"}'  
python_dict = json.loads(json_data)  
print(python_dict["name"])
```

Reading JSON from a File

```
import json  
with open("data.json", "r") as f:  
    content = json.load(f)  
    print(content)
```

Writing JSON to a File:

```
import json
```

```
data = {"course": "Python", "level": "Beginner"}  
with open("data.json", "w") as f:  
    json.dump(data, f)
```

Key Points to Remember:

- Use `dumps()` → Python to JSON string.
- Use `loads()` → JSON string to Python object.
- Use `dump()` → Python object to JSON file.
- Use `load()` → JSON file to Python object.
- JSON looks like Python dictionary but always uses **double quotes (" ")** for keys and strings.