📌JSON in Python (Colorful, Canva-Style Notes)

🕐What is JSON?

- **JSON** = *JavaScript Object Notation*
- A lightweight format for **storing and exchanging data**.
- Looks similar to Python **dicts/lists**, but is actually a **string**.
- Widely used in **APIs, config files, and data storage**.
- Python has a built-in module `` for working with it.

🔗Python Object → JSON String

```
import json

student = {
    "name": "Zahid",
    "age": 24,
    "courses": ["Math", "Physics", "CS"]
}

# Convert Python dict to JSON string
json_string = json.dumps(student)
print(json_string)
```

Output:

```
{"name": "Zahid", "age": 24, "courses": ["Math", "Physics", "CS"]}
```

🔗JSON String → Python Object

```
import json

json_string = '{"name": "Zahid", "age": 24, "courses": ["Math", "Physics",
"CS"]}'

student = json.loads(json_string)
print(student["name"])  # Zahid
```

📂Using JSON with Files

```
import json

data = {"books": ["Math", "Physics", "Chemistry"]}

# Save to file
with open("data.json", "w") as f:
    json.dump(data, f)

# Load from file
with open("data.json", "r") as f:
    new_data = json.load(f)

print(new_data)
```

🔑Key Points

- JSON is **text format** for representing data.
- Very close to **Python dicts/lists**.
- Perfect for **file storage** and **web data exchange**.

📈JSON vs Python (Quick Glance)

Python Dict / List	JSON Format String
`{"name": "Zahid", "age": 24}`	`"{\"name\": \"Zahid\", \"age\": 24}"`
Keys: no quotes needed (but can)	Keys must be **double-quoted**
Values: `True`, `False`, `None`	Values: `true`, `false`, `null`

💡 **In simple words:** JSON is just a **textual representation of Python-like data**, easy to share across systems.⚠ **Important:**

- `dumps` and `loads` are for **strings in memory**.
- `dump` and `load` are for **files**.

Purpose	
`json.dumps()`	**Python object → JSON string**
`json.loads()`	**JSON string → Python object**

Purpose	
`json.dump()`	**Python object → JSON file**
`json.load()`	**JSON file → Python object**