

# JSON, YAML and configuration files

## JSON

JSON stands for JavaScript Object Notation and is a lightweight format for storing and transporting data. Json is often used when data is sent from a server to a web page.

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

Write a JSON file with:

```
>>> import json

>>> content = {"name": "Joe", "age": 20}
>>> with open("filename.json", "w") as f:
...     json.dump(content, f, indent=2)
```

## YAML

Compared to JSON, YAML allows a much better human maintainability and gives ability to add comments. It is a convenient choice for configuration files where a human will have to edit.

There are two main libraries allowing to access to YAML files:

- [PyYaml](#)
- [Ruamel.yaml](#)

Install them using `pip install` in your virtual environment.

The first one is easier to use but the second one, Ruamel, implements much better the YAML specification, and allow for example to modify a YAML content without altering comments.

Open a YAML file with:

```
>>> from ruamel.yaml import YAML

>>> with open("filename.yaml") as f:
...     yaml=YAML()
...     yaml.load(f)
```

## Anyconfig

[Anyconfig](#) is a very handy package, allowing to abstract completely the underlying configuration file format. It allows to load a Python dictionary from JSON, YAML, TOML, and so on.

Install it with:

```
pip install anyconfig
```

Usage:

```
>>> import anyconfig
>>> conf1 = anyconfig.load("/path/to/foo/conf.d/a.yml")
```