

# JSON

---

JSON or JavaScript Object Notation, is a lightweight way to send and transfer data over the web. It is formatted in a way where each chunk of data is stored and retrieved in a key and value pair. The key acts like a title or name of the object, and the value will be the actual data of the object. JSON and dictionaries don't only store strings, but they can also store integers, booleans, lists, and even more dictionaries within! This document will contain a brief introduction into JSON, how we use JSON in our project, and how to modify and save JSON data with Python.

We use JSON to store data we want to modify and easily access, this includes dashboard names and their saved process variables. We can also consider JSON to be very similar to Python Dictionaries or dictionaries in most coding languages, using a key and value pair to save and access values.

## Using JSON with Python

---

In our project, we use JSON to store data which will then be accessed by Python. To do this, we import the JSON module like this:

```
import json
```

This allows json functions to be used within our Python script. We can use functions such as dump, which will dump our dictionary into a JSON file.

In addition, you should also import the module os, which will allow us to delete JSON files when a user deletes their dashboard.

```
import os
```

## Accessing JSON files in Python

---

As said above, JSON is very similar to Python dictionaries, this allows us to read or open the JSON file, make it into a Python dictionary, modify the dictionary to what we want, and then dump the contents back into the JSON file. To open the JSON file, we use the Python command:

```
with open(json_file, 'r') as json_data:  
    # Do some code
```

The code above shows the file or variable `json_file` being open in read mode (`r`), you are also able to specify if you want to write (`w`). In addition, it is also safe to only allow Python to have the necessary permissions into a file, this means only allowing a file to be read and not be written, or having a file to only be written and not read. This helps with the security of the file.

Nice work, you are now able to access the contents of the JSON file.

## Converting JSON data to a Python Dictionary

---

To convert the data within the JSON file to a Python Dictionary, we use the command `load`, which is from the `json` module. After we use the function `load`, we must put this data into a Python variable so it can be modified later on.

```
import json

with open(json_file, 'r') as json_data:
    python_dictionary = json.load(json_data)
```

The example above shows us converting the JSON data (`json_data`) and converting it to a Python dictionary (`python_dictionary`). This will now allow us to modify and send this data to our HTML page!

An example of this is shown below:

```
with open(dash_file, "r") as file_read:
    json_for_dash = json.load(file_read)
return render_template("public/edit_dash.html", data=json_for_dash,
    dashboard_name=dashboard)
```

## Modifying Python Dictionary Data

---

Now that we know how to access and convert JSON to something we can modify with Python. As we said before, dictionaries and JSON use a key value pair to store all their data, this means that when trying to access a value, we must call it by its key. In Python, we specify the key using `[]`. We push the key as a string into these brackets, which will then return the value we want.

Let's give a small demonstration. Let's say our dictionary looks like this

```
python_dict = {
    value1: 'Hello World',
    value2: 'My Second Value',
    value3: 3,
    value4: False
}
```