

# JSON Files

This lesson focuses on JSON type files. It gives a complete explanation about how to read data from JSON files using the Pandas library of Python.

## WE'LL COVER THE FOLLOWING ^

- Introduction to JSON file
- Reading JSON file with Python

## Introduction to JSON file #

**JSON** (JavaScript Object Notation) is a popular format allowing for a more flexible schema. It is also easy for humans to read and write. A lot of the data sent around the web is transmitted as JSON. Here is an example:

```
{
  "glossary": {
    "title": "example glossary",
    "GlossDiv": {
      "title": "S",
      "GlossList": {
        "GlossEntry": {
          "ID": "SGML",
          "SortAs": "SGML",
          "GlossTerm": "Standard Generalized Markup Language",
          "Acronym": "SGML",
          "Abbrev": "ISO 8879:1986",
          "GlossDef": {
            "para": "A meta-markup language, used to create markup languages such as DocBook.",
            "GlossSeeAlso": ["GML", "XML"]
          },
          "GlossSee": "markup"
        }
      }
    }
  }
}
```

## Reading JSON file with Python #

Python can easily read these data from strings into dictionaries using the built-in `json` library:

```
import json

## Define the JSON object as a string
json_string = """{
  "glossary": {
    "title": "example glossary",
    "GlossDiv": {
      "title": "S",
      "GlossList": {
        "GlossEntry": {
          "ID": "SGML",
          "SortAs": "SGML",
          "GlossTerm": "Standard Generalized Markup Language",
          "Acronym": "SGML",
          "Abbrev": "ISO 8879:1986",
          "GlossDef": {
            "para": "A meta-markup language, used to create markup languages such
            "GlossSeeAlso": ["GML", "XML"]
          },
          "GlossSee": "markup"
        }
      }
    }
  }
}"""

# Read the JSON data into Python
json_data = json.loads(json_string)

print(json_data)
```

When your JSON data are in a string, you can use the `loads()` function to read it into a Python dictionary (**line 29**). JSON is the same format as a dictionary in that it consists of key, value pairs of various types. Our JSON object above has a key for `title` and the value is **example glossary**. It also has a key of `GlossList` that has a **dictionary object** for a value that contains its own key-value pairs. You can see how JSON can represent any type of nested schema you would want to have.

Now that you have your JSON object as a dictionary you can access the values

Now that you have your JSON object as a dictionary you can access the values in the same way we discussed in the [data structures lesson](#). If you have a JSON in a file, you read the data using the `load()` function. Here is how you would do so if you had a file called **data.json**:

```
with open('data.json') as f:  
    data = json.load(f)
```

Now that you're familiar with JSON type files, in the next section, we will look at reading in raw text files.