

# FILE FORMATS

An Introduction to Computer Science



Let's learn about File Formats.

# File Formats

As you can see in this CSV file...



Ah yes, you're right, I see...



When data is placed into a file, it often follows a regular structure. That way, other users of the file can navigate the data more easily. This structure is known as a file format. Typically, a file's format is used to determine its extension. This is why Python files have a ".py" at the end - even though they are text, they are structured according to the Python language.

## Recall: Opening a File

```
data_file = open('filename.format')
```

Open  
function

Path



Usually, regardless of the file format, we begin using it by opening the file.  
Recall how we open text files using the open function, giving us a file object.  
Previously, we then read the file directly.  
Instead, we will now use special modules to read a file.

## JSON Format

```
import json  
  
json_file = open('data.json')  
  
data = json.load(json_file)
```



The JSON format is a particularly useful format, and is available in many programming languages.

The JSON format lets you store lists, dictionaries, and simple primitive types in a file.

When you load the data, you get the same structure back.

The example shown demonstrates how easy this is.

The "data" variable could now hold a complex nested structure of data, without having to embed it in the program.

## Quickly Visualizing Data

```
print(json.dumps(data[0], indent=2))
```



```
{  
  "Name": "Klaus",  
  "Age": 17,  
  "Big?": true  
}
```



Have a complex data structure, like a list of dictionaries?

Here's a quick pattern for easily visualizing the first element.

A very curious thing to note is second parameter for this function call.

This is known as a "keyword argument", and we have to explicitly give both the parameter and the argument.

The syntax is similar to an assignment, except all of it is embedded in the function call.

## CSV Format

```
import csv

csv_file = open('data.csv')
for line in csv.reader(csv_file, delimiter=","):
    print(line)
```

CSV Reader has  
many features



Comma-Separated Values, or CSV, files are another very popular way to share data. An advantage of CSV files is that they can be opened in programs like Excel, edited like text files, or processed in programming languages like Python.

To process a CSV File, you need to use the csv reader function, which can be a little complex.

This is a situation where referring to documentation and using Google is necessary.

## Other Formats

- TXT
- JSON
- CSV
- HDF5
- HTML
- XML
- DAT
- LOG
- SQL
- JAR
- ZIP
- RAR
- BMP
- PNG
- MP3
- WAV
- XLS
- XLSX
- CFG
- INI
- RTF
- MD
- RST
- ...



There are many, many other file formats out there.

Some text formats are easy to process, but others require special libraries.

When encountering a new format, you should always check to see if there is a module available to load the data.