

Reading and Writing to Files: Takeaways



by Dataquest Labs, Inc. - All rights reserved © 2021

Syntax

- Get the local encoding:

```
import locale
print(locale.getpreferredencoding())
```

- Open a file with a context manager:

```
with open('data.txt') as file:
    for line in file:
        print(line)
```

- Read the bytes of a file:

```
with open('dialog.txt', mode='rb') as file:
    raw_bytes = file.read()
```

- Write to a file:

```
with open('data.txt', mode='w') as file:
    file.write('line 1\n')
    file.write('line 2\n')
```

- Append to a file:

```
with open('data.txt', mode='a') as file:
    file.write('line 3\n')
    file.write('line 4\n')
```

- Identifying the encoding of a file:

```
import chardet
with open(file_name, mode='rb') as file:
    raw_bytes = file.read(32)
    encoding_name = chardet.detect(raw_bytes)['encoding']
```

- Convert a CSV file from one encoding to another:

```
import csv
with open('original.csv', encoding=original_enc) as file:
    rows = list(csv.reader(file))
with open('new.csv', mode='w', encoding=new_enc) as file:
    writer = csv.writer(file)
    for row in rows:
        writer.writerow(row)
```

Concepts

- A context manager provided a better way to open files as it automatically closes them for us afterwards.

- The `open()` function provides several options for interacting with files:
 - `r` : This is the default options to read files.
 - `w` : This options is used to write to a file. Be careful as doing so will erase its contents. If the file does not exist, it will be created.
 - `a` : Append data to the end of the file.
 - `rb` : Read the bytes of the file.
 - `wb` : Write bytes to the file.

Resources

- [The open function](#)
- [Python context managers](#)
- [Selecting the right open mode](#)