

Software Engineering  
Bootcamp



# Working with External Data Sources - I/O

# Objectives

- Learn how to read and write data to files

# File Modes

Table 1: Python File modes

Mode	Description
'r'	Opens a file for reading.
'w'	Open a file for writing. If file does not exist, it creates a new file. If file exists it truncates the file.
'a'	Open a file in append mode. If file does not exist, it creates a new file.
'+'	Open a file for reading and writing (updating)

# Opening Files

*Relative or Absolute path to the file  
(including the extension)*

```
open ( <file> , mode )
```

*A String (Character) indicates  
what you want to do with that file*

BOARD

# Resource Management

## Implicit Method

- ❖ The **with** statement is used for resource management in Python.
- ❖ It ensures that resources are properly cleaned up after use, even if an error occurs.

```
with open('filename.txt', 'r') as file:  
    content = file.read()
```

# Resource Management

## Explicit Method

- ❖ The explicit way involves manually opening and closing files using the **open()** function for opening and the **close()** method for closing.

```
file = open('file.txt', 'r')  
content = file.read()  
file.close()
```

## File Modes (r)

- ❖ **Reading from Text Files:** You can read text from a file using the **open()** function with the mode **'r'**

```
with open('filename.txt', 'r') as file:  
    content = file.read()
```

## File Modes (w)

- ❖ **Writing to Text Files:** You can write text to a file using the **open()** function with the mode **'w'**

```
with open('filename.txt', 'w') as file:  
    file.write("Hello, world!")
```



## File Modes (a)

- ❖ **Appending to Text Files:** You can append text to an existing file using the **open()** function with the mode **'a'**

```
with open('filename.txt', 'a') as file:  
    file.write("\nThis is a new line.")
```

# File Handling (Reading)

## Read from a File Python Methods

**read()**  
Reads the entire  
contents of the file  
and returns it as a  
string.

**readline()**  
Reads a single line  
from the file and  
returns it as a  
string.

**readlines()**  
Reads all lines  
from the file and  
returns them as a  
list of strings.

# Writing to Files

- ★ Often, we will want to **write** data to a **new file**.
- ★ Usually after we have done a lot of computations or data processing and we would like to **save** the work and **come back** to it at another point.
- ★ Writing to a file has a simple **multi-step process**.

# File Handling (Writing)

## Write to a File Python Methods

### **write()**

This method is used to write data to the file. It takes a string argument and adds it to the end of the file.

### **writelines()**

This method writes a sequence of strings to the file. It takes a list of strings as an argument and writes each string to the file.

# Prepping the file

- ★ We already know how to open a file and store the file object in a variable.
- ★ Now the main difference between Input and Output is the access mode now changes.
  - Instead of reading from the file, we are now writing to the file (w , w+ , a)
- ★ What comes next is then, actually writing to the file. Which we will take a look at now.

# Writing Example

```
with open("output.txt", "w") as file:

    file.write("Mankind knew, that they cannot change society.\n")
    file.write("So instead of reflecting on themselves.\n")
    file.write("They blamed the beasts")

print("Items written")  # Sanity check

# The write function, will write any data we provide
#   within parentheses to our file.
#       and since we're using a with as block.
#       we don't need to use .close()
```

# Things to Note

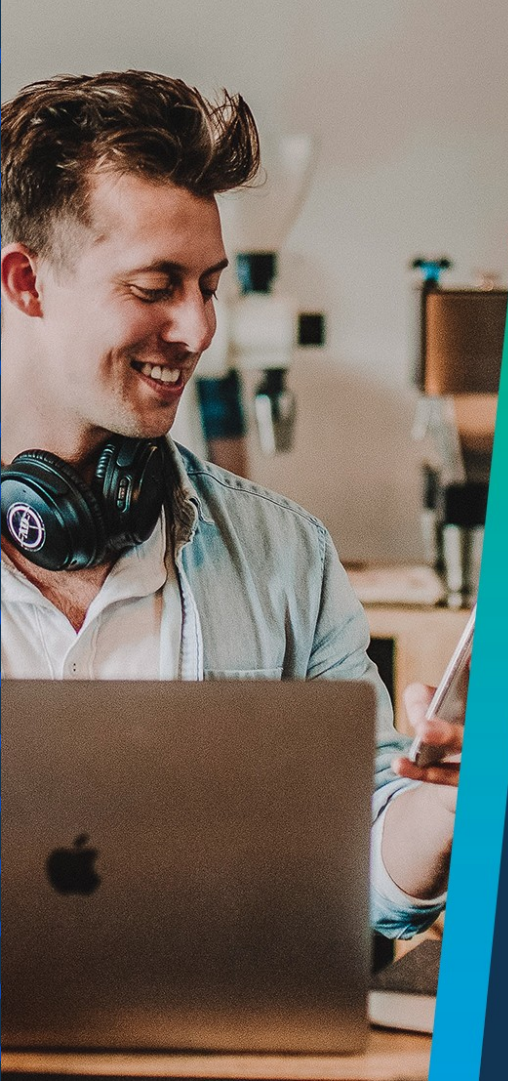
- ★ Remember that when the file is reopened and new data is written to the file, the previous data is then overwritten.
- ★ There is a way to preserve the previous data by using the append ( a ) access mode, this will simply append the new data to the end of the file, instead of overwriting.
- ★ Always remember to close your file when you are done using it.



# Q & A Section

Please use this time to ask any questions relating to the topic explained, should you have any





**Thank you  
for joining us**