

ECE 20875

Python for Data Science

Chris Brinton and Qiang Qiu

**(Adapted from material developed by Profs. Milind Kulkarni,
Stanley Chan, Chris Brinton, David Inouye)**

file I/O

file I/O

- In Python, we can read and write from files
- In Python (and most programming languages), file operation takes place in the following order:
 1. Open a file
 2. Read or write (perform operation)
 3. Close the file



closing a file

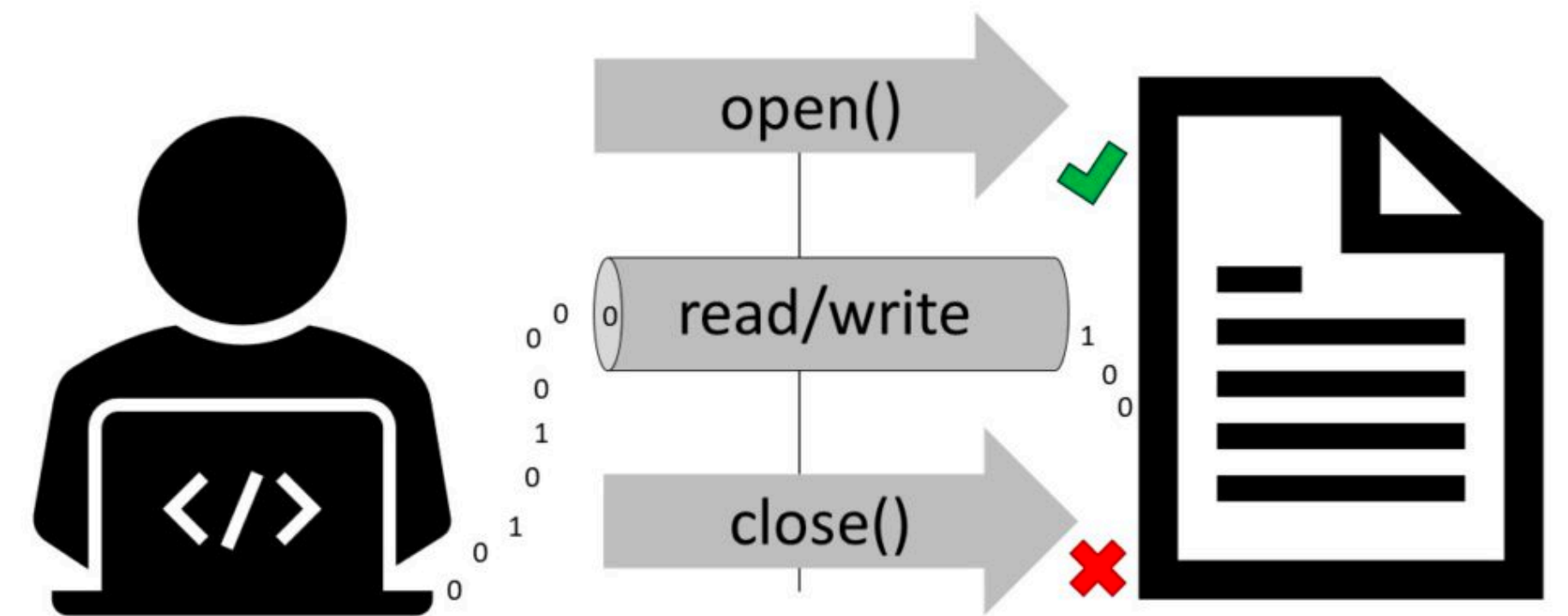
- Closing a file: `close()` method
- Free up resources that were tied up with the file
- Exception handling: Use `try ... finally` block

`try:`

```
f = open("test.txt", 'w')  
# perform file operations
```

`finally:`

```
f.close()
```



writing a file

- Writing files: open in `write` or `append` mode
 - `'w'` will overwrite existing file, while `'a'` will add to the end of it
 - The `write("text")` method will write text to the file

```
with open("test.txt", 'w') as f:  
    f.write("my first file\n")  
    f.write("This file\n\n")  
    f.write("contains three lines\n")
```

reading a file

- Reading files: open in read mode
- Can optionally specify the number of characters to read

```
f = open("test.txt", 'r')
f.read(4)    # read the first 4 characters
f.read(4)    # read the next 4 characters
f.read()     # read in the rest until the end
f.close()
```

```
f = open("test.txt", 'r')
f.readline() # reads the first line (delimited by \n)
f.readlines() # reads the remaining lines, returns as list
f.close()
```