

ECE 20875

Python for Data Science

Chris Brinton and David Inouye

file I/O

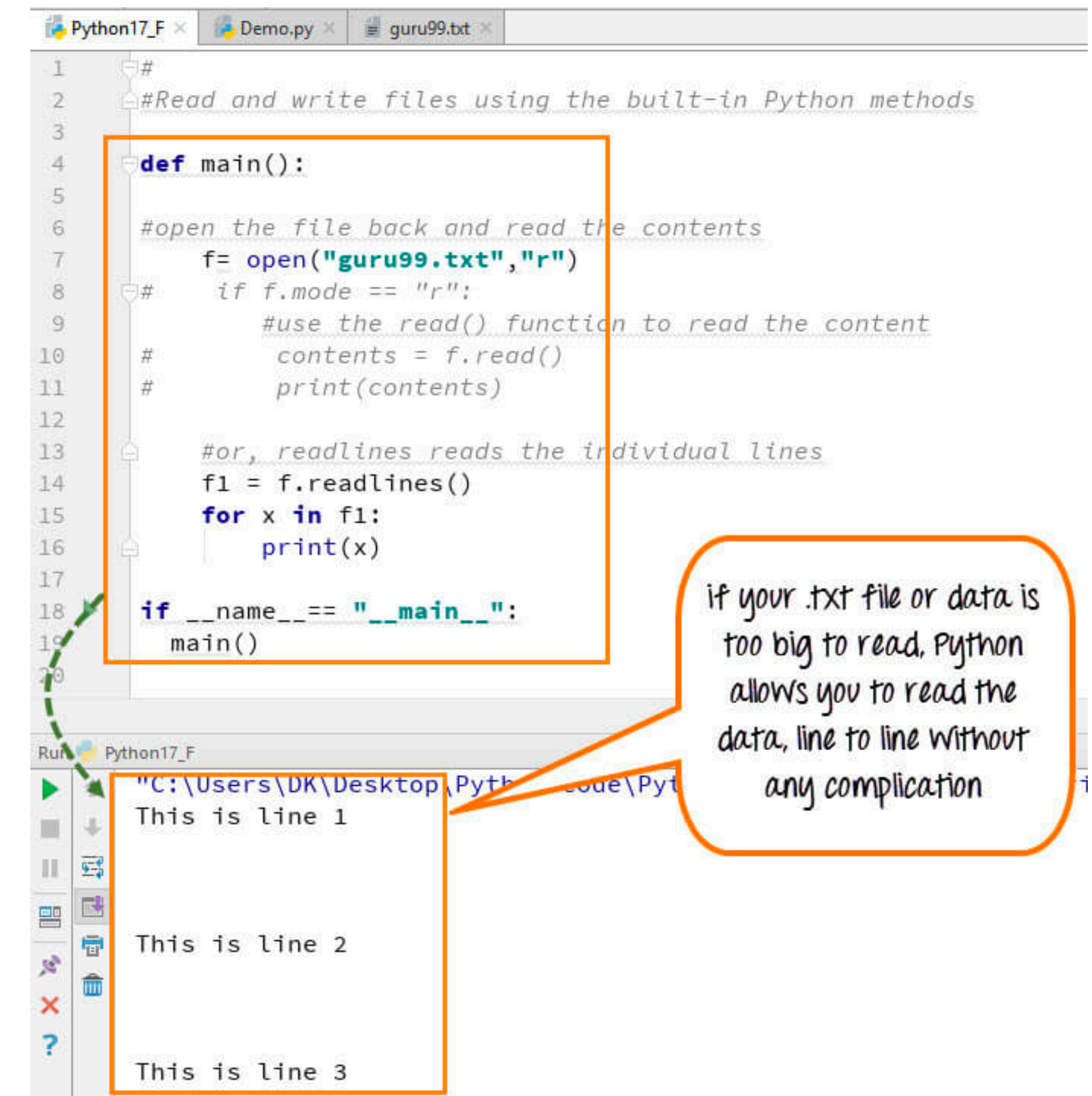
file I/O

- 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



opening a file

- Use the `open()` method
- Returns a **file object (handle)** used to read or write
- Specify the mode: most common are read 'r', write 'w', append 'a'
- `f = open("test.txt", 'w')` # write in text mode



```
1 #
2 #Read and write files using the built-in Python methods
3
4 def main():
5
6     #open the file back and read the contents
7     f= open("guru99.txt","r")
8     # if f.mode == "r":
9         #use the read() function to read the content
10        # contents = f.read()
11        # print(contents)
12
13    #or, readlines reads the individual lines
14    f1 = f.readlines()
15    for x in f1:
16        print(x)
17
18    if __name__ == "__main__":
19        main()
20
```

if your .txt file or data is too big to read, Python allows you to read the data, line to line without any complication

```
"C:\Users\DK\Desktop\Python17_F\code\Python17_F\
This is line 1

This is line 2

This is line 3
```

closing a file

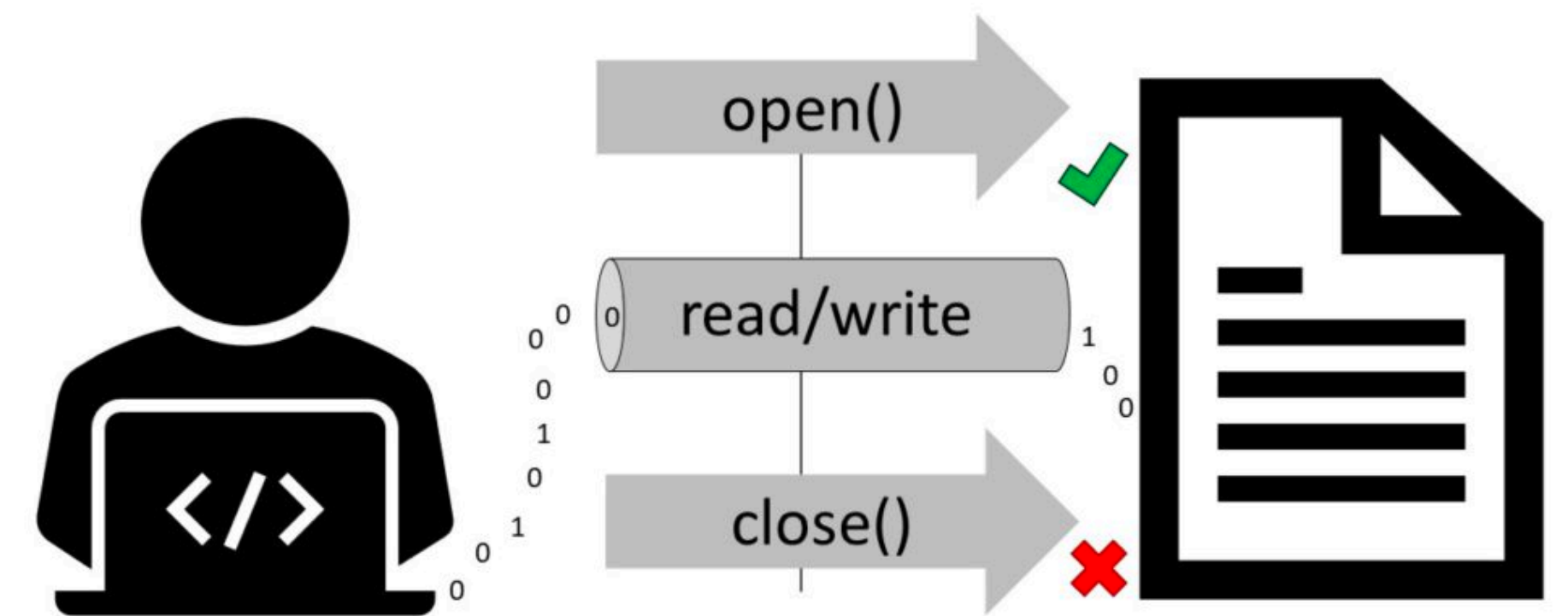
- Closing a file: `close()` method
- Free up resources that were tied up with the file
- Exception handling: Use `try ... finally` block

`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()
```