Revised: 2/17/2018

File Access

Michael C. Hackett

Computer Science Department

Community College of Philadelphia

Topics

- Random Access Reading data from text files.
  - Reading, Writing, and Appending Data

- Modifying Existing Files.

Colors/Fonts

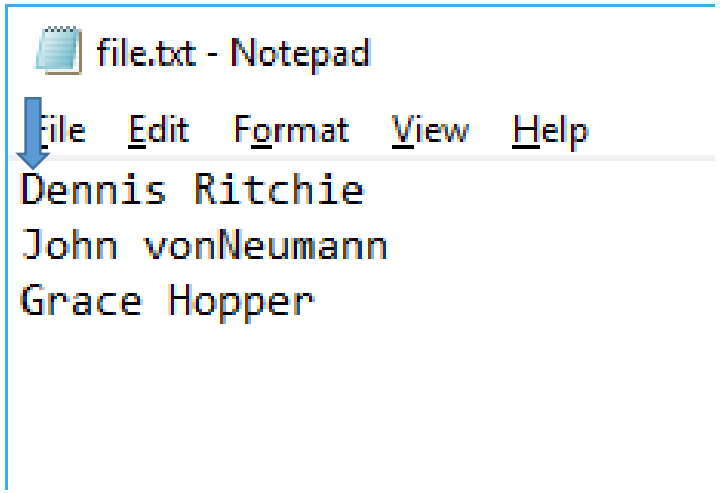
- Global Variable Names – Brown
- Local Variable Names – Lt Blue
- Literals – Blue
- Keywords – Orange
- Operators/Punctuation – Black
- Functions – Purple
- Parameters – Gold
- Comments – Gray
- Modules – Pink

Source Code – **Consolas**
Output – Courier New

Reading Data Randomly from a Text File

- A file object has a pointer that keeps track of the location we are currently reading from the file.
  - When the file is first opened, the pointer is at position 0 (The first character of the first line in the file.

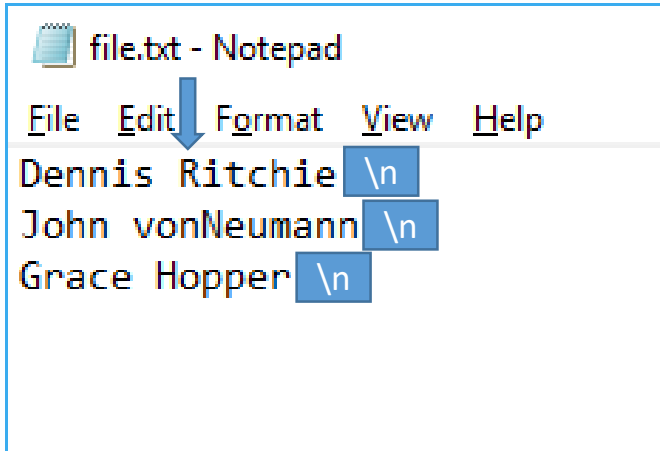
```
my_text_file = open("file.txt", "r")
```



Reading Data Randomly from a Text File

- To move the pointer, call the file object's seek function.
  - The int argument tells the pointer what position to move to.

```
my_text_file = open("file.txt", "r")
my_text_file.seek(7)
```

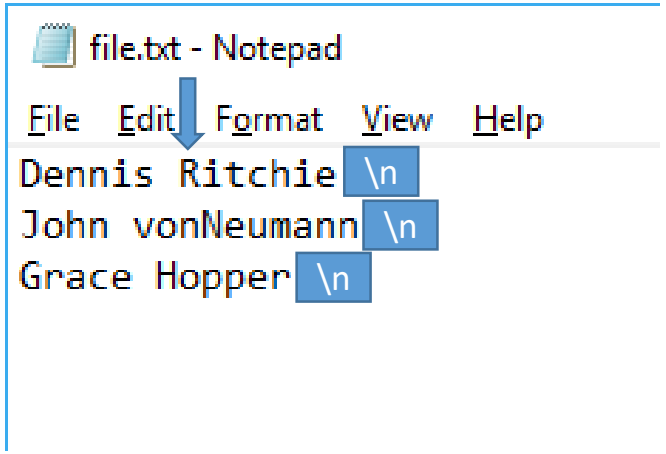


Reading Data Randomly from a Text File

- This new position is where data will be read from the file.

```
my_text_file = open("file.txt", "r")
my_text_file.seek(7)
line = my_text_file.readline().rstrip("\n")
print(line)
my_text_file.close()
```

```
Ritchie
```

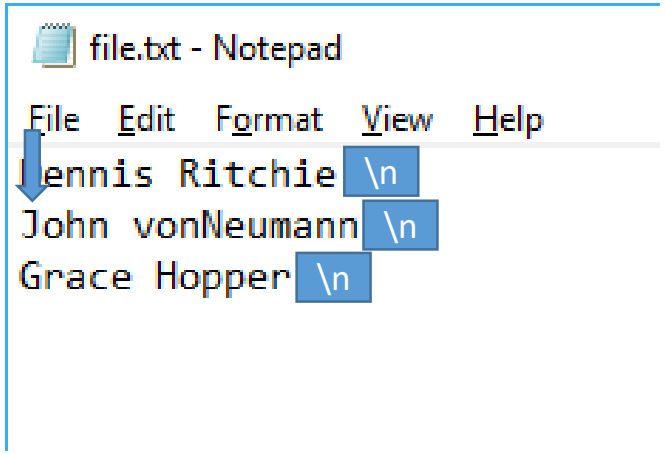


Reading Data Randomly from a Text File

- To see the pointer's current position, use the file object's tell function.

```
my_text_file = open("file.txt", "r")
my_text_file.seek(7)
line = my_text_file.readline().rstrip("\n")
print(line)
current_position = my_text_file.tell()
print(current_position)
my_text_file.close()
```

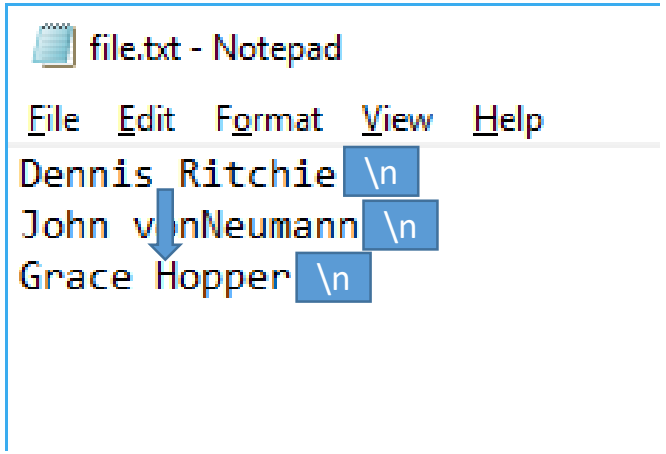
```
Ritchie
16
```



Reading Data Randomly from a Text File

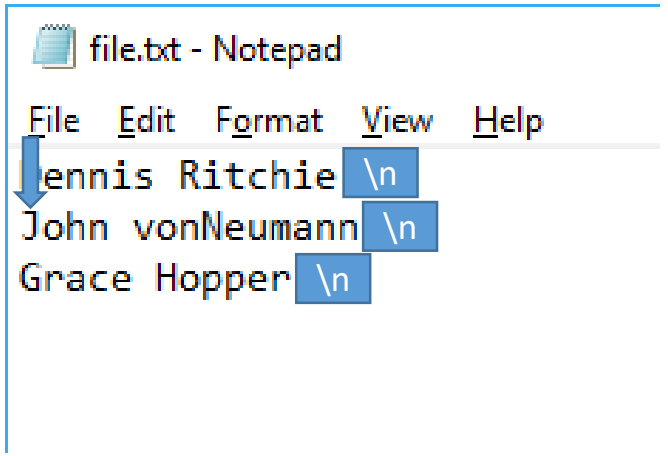
```
my_text_file = open("file.txt", "r")
my_text_file.seek(7)
line = my_text_file.readline().rstrip("\n")
print(line)
current_position = my_text_file.tell()
print(current_position)
my_text_file.seek(39)
line = my_text_file.readline().rstrip("\n")
print(line)
my_text_file.close()
```

```
Ritchie
16
Hopper
```



Reading Data Randomly from a Text File

```
my_text_file = open("file.txt", "r")
my_text_file.seek(7)
line = my_text_file.readline().rstrip("\n")
print(line)
current_position = my_text_file.tell()
print(current_position)
my_text_file.seek(39)
line = my_text_file.readline().rstrip("\n")
print(line)
my_text_file.seek(16)
line = my_text_file.readline().rstrip("\n")
print(line)
my_text_file.close()
```



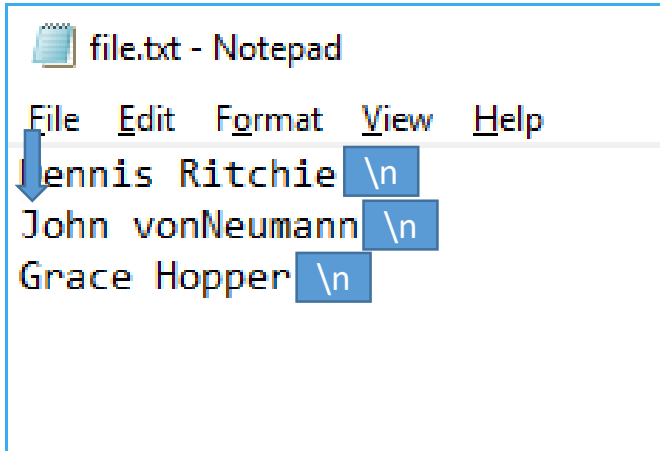

```
Ritchie
16
Hopper
John vonNeumann
```

Reading Data Randomly from a Text File

- The file object's read function normally returns the entire file as one string.
- You can provide the number of characters to read by passing a numeric argument to the read function.

```
my_text_file = open("file.txt", "r")
my_text_file.seek(16)
line = my_text_file.read(4)
print(line)
my_text_file.close()
```

```
John
```

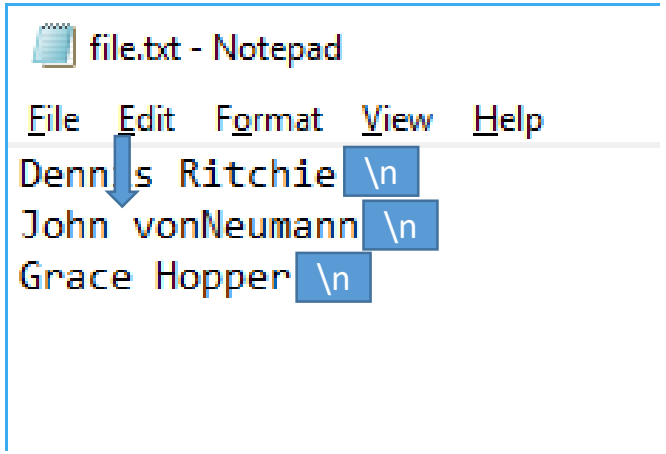


Reading Data Randomly from a Text File

- The pointer will be left at the next character.

```
my_text_file = open("file.txt", "r")
my_text_file.seek(16)
line = my_text_file.read(4)
print(line)
line = my_text_file.read(5)
print(line)
my_text_file.close()
```

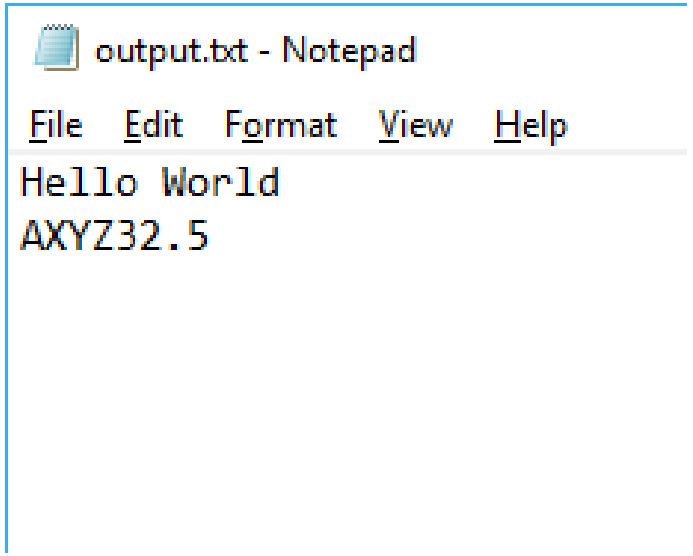
```
John
 vonN
```



Writing Data Randomly to a New Text File

- Most files are written using sequential access.
  - Writing using random access is possible.

```
my_output_file = open("output.txt", "w")
my_output_file.write("Hello World\n")
my_output_file.write("ABCD")
my_output_file.write(str(32.5))
my_output_file.seek(14)
my_output_file.write("XYZ")
my_output_file.close()
```



Modifying Existing Text Files

- There is no mode for modifying existing text files.
  - Write mode causes existing information to be erased.
  - Append mode only adds data to the file.
    - It does not allow changing the existing data.

- There is a (not entirely simple) process to modify an existing file.

Modifying Existing Text Files

1. Open the existing file in read-only mode.
2. Open a new, temporary file in write mode.
   A. Read each line from the existing file and write the lines to the new, temporary file.
   B. Make any modifications or edits during this process.
3. Close both files.
4. Delete the original file using Python's os module.
5. Rename the new file with the original's name using Python's os module.

Modifying Existing Text Files

```
import os
original_file = open("output.txt", "r")  1
new_file = open("temp.txt", "w")  2
for line in original_file :
    #Make any changes to the line
    new_file.write(line)
new_file.close()
original_file.close()
os.remove("output.txt")  4
os.rename("temp.txt", "output.txt")  5
```

2A/B

3

Modifying Existing Text Files

```
import os
original_file = open("output.txt", "r") 1
new_file = open("temp.txt", "w") 2
for line in original_file :
  new_file.write(line)
new_file.seek(17)
new_file.write("0000")
new_file.close()
original_file.close()
os.remove(original_file.name) 4
os.rename(new_file.name, original_file.name)5
```

2A

2B

3