File Input and Output (I/O)

CSCI 1030U - Intro to Computer Science
@IntroCS

Randy J. Fortier
@randy_fortier

OntarioTech UNIVERSITY

Outline

- File I/O
  - Reading from files
  - Writing to files

File Input and Output

Ontario Tech UNIVERSITY

Reading from Files

- Opening a file
  - `file = open('data.txt', 'r')`

- Reading from a file

– `file.read()`	Read the entire file (into a string for text files)
– `file.seek(10)`	Jump to the 10th character/byte
– `file.read(5)`	Read 5 characters/bytes
– `file.readline()`	Read 1 line (newline separated)
– `file.readlines()`	Read all lines into a list (newline separated)
– `list(file)`	Read all lines into a list (newline separated)

Reading from Files: An Example

- This code reads a file, line-by-line, and then closes it

```
file = open('data.txt', 'r')
for line in file:
    print(line)
file.close()
```

Reading from Files: Another Example

- This code does the same thing, but closing is implicit:

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

Writing to Files: An Example

- This code saves data to a file:

```
items = [1,2,3,4,5]
with open('data.txt', 'w') as file:
    for item in items:
        file.write(str(item) + '\n')
```

JSON

- JSON is a simple notation for anything tree-structured, similar to XML in capability, but much simpler
  - This will be covered in more detail in other courses
- Example:

```
{
    "sid": "100000001",
    "first_name": "Carla",
    "last_name": "Rodriguez",
    "grades": ["100.0","90.0","83.0"],
}
```

Reading JSON

- Example:

```
import json

with open('data/carla.json', 'r') as file:
    carla = json.load(file)
    print(carla)
```

Writing JSON

- Example:

```
carla = {
    'sid': '100000001',
    'first_name': 'Carla',
    'last_name': 'Rodriguez',
    'grades': ['100.0','90.0','83.0']
}
with open('data/carla_output.json', 'w') as file:
    json.dump(carla, file)
```

Programming Exercise 08a.1

- Write some code to output the following data values to a file (`grade_output.csv`) in comma-separated value format:

```
sids = ['100000000', '100000001', '100000002', '100000003',
'100000004', '100000005', '100000006', '100000007', '100000008',
'100000009']
midterm_marks = [52.0, 48.5, 54.25, 61.5, 64.0, 77.75, 29.0, 91.25,
68.25, 59.75]
```

Programming Exercise 08a.2

- Write some code to input the same data file (`grade_output.csv`) that was output in the previous exercise, and put those values into a list of dictionaries:

```
100000000,52.0
100000001,48.5
100000002,54.25
100000003,61.5
100000004,64.0
100000005,77.75
…
```

Wrap-up

- File I/O
  - Reading from files
  - Writing to files

Coming Up

- Exceptions
  - Catching exceptions
  - Throwing exceptions
  - Custom exceptions
- Testing
  - Unit testing