CS 1340:Fall 2020:Lecture 08

Intro to Python for CS and Data Science

Mark Fontenot, PhD

Southern Methodist University

Types - The Highlights

Dictionaries

A great additional resource to understanding them: Python Dictionaries 101: A Detailed Visual Introduction

Write Some Code

```
# What's the average age of all the friends?
friends = {
    'Bob': 19,
    'Sally': 20,
    'Muhammad': 18
}
```

File Input

Let’s Make a Text File

Let's Make a Text File

- Right click in the Project Explorer (VS Code)
- Choose `New File`
- Type a name and hit enter/return
- Type your three favorite colors in the file, each on a separate line

Reading From a File

- right-click on the file you created and choose copy path.
- store the path in a variable
- open the file

```
file_path = '/Users/mark/Code/cs1340-f2020/lectures-drafting/abc.test'
color_file = open(file_path, 'r')
```

- `r` means `reading`
- `w` means `writing`
- `x` means creating and writing to a new file

Let's Read

```
# Continuing from the previous slide

color01 = color_file.readLine()
color02 = color_file.readLine()
color03 = color_file.readLine()
```


Strings - 1

- **string literal** - 'SMU'
  - something enclosed in quotes
- **sequence type** - an ordered collection of items
  - `'SMU'` is really like `S M U`
  - The position in the sequence is called the **index**
  - **Indexes** start at **0**

Strings - 2

(you know this already, but ...)

```
school_mascot = 'Peruna'
print (school_mascot)
print ('school_mascot')
```

```
Peruna
school_mascot
```


Strings - 3

- You can find the length of a *sequence* by using the `len()` function

```
school_mascot = 'Peruna'
print(len(school_mascot))
```

```
6
```

Strings - 4

- String Concatenation

```
f_name = 'Mark'
l_name = 'Fontenot'
full_name = f_name + ' ' + l_name
```

(as we saw last week...)

- you cannot directly modify a string
  - you have to create a new string with what you want
  - ... just a reminder

Lists - 1

- **list** - a container that stores related values together
- created using square brackets `[]`

```
universities = ['smu', 'tcu', 'utd', 'uta']
for school in universities:
    print (school)
```

```
smu
tcu
utd
uta
```