Data Foundations: Review

Instructor: Anthony Rios

Midterm: What will be covered

- Basic data types and data structures (floats, strings, lists, sets, dicts, ...)
- Conditional Statements (if, elif, else) and boolean expressions (and, not/!, in, or)
- Looping constructs (for, while), as well as how to use range().
- File IO (Text files, CSVs, XML, JSON, JSONL)
- Functions
- Regular Expressions
- Classes

What will the Midterm be like?

- You will have 2 hours to complete the exam.
- The exam will be posted Monday and you will have until Friday to complete it.
  - Remember once you start the exam it must be completed in 2 hours
- The exam will contain approximately 35 questions.
- Most questions will be similar to the quizzes.
- Expect 3 programming questions. These will be simple questions, much easier than the homework.

Midterm Administrivia

Review

- Old Quiz Questions
- Old Quiz Questions
- Old Quiz Questions

Old Quiz Questions Review

myfile.json

```
[{”name”:”Anthony”, ”email”:”anthony@utsa.edu”}, {”name”:”John”,
”email”:”abc@utsa.edu”}, {”name”:”Jane”, ”email”:”test@fbi.gov”}]
```

example.py

```
import json
myFile = open(‘myfile.json’)
data = json.load(myFile)
myFile.close()
new_items = []
cnt = 0
for item in data:
        if ‘utsa.edu’ in item[‘email’]:
                cnt += 1
                new_item = item
                new_item[‘phone’] = “555-5555”
                print(‘cnt: {} item: {} ’.format(cnt, new_item))
        else:
                print(‘cnt: {} item: {} ’.format(cnt, item))
                new_items.append(item)
                cnt -= 1
print(‘len 1: {} len 2: {} cnt: {}’.format(len(data), len(new_items), cnt))
```

Old Quiz Questions Review

myfile.json

```
[{”name”:”Anthony”, ”email”:”anthony@utsa.edu”}, {”name”:”John”,
”email”:”abc@utsa.edu”}, {”name”:”Jane”, ”email”:”test@fbi.gov”}]
```

example.py

```
import json
myFile = open(‘myfile.json’)
data = json.load(myFile)
myFile.close()
new_items = []
cnt = 0
for item in data:
        if ‘utsa.edu’ in item[‘email’]:
                cnt += 1
                new_item = item
                new_item[‘phone’] = “555-5555”
                print(‘cnt: {} item: {} ’.format(cnt, new_item))
        else:
                print(‘cnt: {} item: {} ’.format(cnt, item))
                new_items.append(item)
                cnt -= 1
print(‘len 1: {} len 2: {} cnt: {}’.format(len(data), len(new_items), cnt))
```

```
anthony@MacBook:~$ python example.py
cnt 1 item {‘name’:‘Anthony’, ‘email’:‘anthony@utsa.edu’, ‘phone’:‘555-5555’}
```

myfile.json

```
[{"name":"Anthony", "email":"anthony@utsa.edu"}, {"name":"John",
"email":"abc@utsa.edu"}, {"name":"Jane", "email":"test@fbi.gov"}]
```

example.py

```
import json
myFile = open(‘myfile.json’)
data = json.load(myFile)
myFile.close()
new_items = []
cnt = 0
for item in data:
        if ‘utsa.edu’ in item[‘email’]:
                cnt += 1
                new_item = item
                new_item[‘phone’] = “555-5555”
                print(‘cnt: {} item: {} ’.format(cnt, new_item))
        else:
                print(‘cnt: {} item: {} ’.format(cnt, item))
                new_items.append(item)
                cnt -= 1
print(‘len 1: {} len 2: {} cnt: {}’.format(len(data), len(new_items), cnt))
```

```
anthony@MacBook:~$ python example.py
```

```
cnt 1 item {‘name’:‘Anthony’, ‘email’:‘anthony@utsa.edu’, ‘phone’:‘555-5555’}
cnt 2 item {‘name’:‘John’, ‘email’:‘abc@utsa.edu’, ‘phone’:‘555-5555’}
```

myfile.json

```
[{"name":"Anthony", "email":"anthony@utsa.edu"}, {"name":"John",
"email":"abc@utsa.edu"}, {"name":"Jane", "email":"test@fbi.gov"}]
```

example.py

```
import json
myFile = open('myfile.json')
data = json.load(myFile)
myFile.close()
new_items = []
cnt = 0
for item in data:
        if 'utsa.edu' in item['email']:
                cnt += 1
                new_item = item
                new_item['phone'] = "555-5555"
                print('cnt: {} item: {} '.format(cnt, new_item))
        else:
                print('cnt: {} item: {} '.format(cnt, item))
                new_items.append(item)
                cnt -= 1
print('len 1: {} len 2: {} cnt: {}'.format(len(data), len(new_items), cnt))
```

```
anthony@MacBook:~$ python example.py
```

```
cnt 1 item {'name':'Anthony', 'email':'anthony@utsa.edu', 'phone':'555-5555'}
cnt 2 item {'name':'John', 'email':'abc@utsa.edu', 'phone':'555-5555'}
cnt 2 item {'name':'Jane', 'email':'test@fbi.gov'}
```

myfile.json

```
[{”name”:”Anthony”, ”email”:”anthony@utsa.edu”}, {”name”:”John”,
”email”:”abc@utsa.edu”}, {”name”:”Jane”, ”email”:”test@fbi.gov”}]
```

example.py

```
import json
myFile = open(‘myfile.json’)
data = json.load(myFile)
myFile.close()
new_items = []
cnt = 0
for item in data:
        if ‘utsa.edu’ in item[‘email’]:
                cnt += 1
                new_item = item
                new_item[‘phone’] = “555-5555”
                print(‘cnt: {} item: {} ’.format(cnt, new_item))
        else:
                print(‘cnt: {} item: {} ’.format(cnt, item))
                new_items.append(item)
                cnt -= 1
print(‘len 1: {} len 2: {} cnt: {}’.format(len(data), len(new_items), cnt))
```

```
anthony@MacBook:∼$ python example.py
```

```
cnt 1 item {‘name’:‘Anthony’, ‘email’:‘anthony@utsa.edu’, ‘phone’:‘555-5555’}
cnt 2 item {‘name’:‘John’, ‘email’:‘abc@utsa.edu’, ‘phone’:‘555-5555’}
cnt 2 item item {‘name’:‘Jane’, ‘email’:‘test@fbi.gov’}
len 1: 3 len 2: 1 cnt: 1
```

Old Quiz Questions Review

example.py

```
import re
text = “The price was $199.99, not $199. I wish it was only a $1.50.”
print(re.findall(“\$[0-9]+\.[0-9][0-9]”, text))
```

```
anthony@MacBook:∼$ python example.py
[‘$199.99’, ‘$1.50’]
```

Midterm Administrivia

Review

- Old Quiz Questions
- Old Quiz Questions
- Old Quiz Questions

Old Quiz Questions Review

example.py

```
cnt = 0
accumulator = 0
print(“cnt {} accumulator: {}”.format(cnt, accumulator))
for i in range(5):
    if cnt % 2 == 1:
        accumulator += cnt + i
    cnt += 1
    print(“cnt: {} accumulator: {}”.format(cnt, accumulator))
```

```
anthony@MacBook:~$ python example.py
```

Old Quiz Questions Review

example.py

```
cnt = 0
accumulator = 0
print(“cnt: {} accumulator: {}”.format(cnt, accumulator))
for i in range(5):
    if cnt % 2 == 1:
        accumulator += cnt + i
    cnt += 1
    print(“cnt: {} accumulator: {}”.format(cnt, accumulator))
```

```
anthony@MacBook:~$ python example.py
cnt: 0 accumulator 0
```

Old Quiz Questions Review

example.py

```
cnt = 0
accumulator = 0
print(“cnt: {} accumulator: {}”.format(cnt, accumulator))
for i in range(5):
    if cnt % 2 == 1: #
        accumulator += cnt + i
    cnt += 1
    print(“cnt: {} accumulator: {}”.format(cnt, accumulator))
```

```
anthony@MacBook:~$ python example.py
cnt: 0 accumulator 0
```

Old Quiz Questions Review

Fill in the blank

example.py

```
text = “Teddy Bear”
char_counts = {}
for char in text:
        if char in char_counts:
                char_counts[char] _________
        else:
                char_counts[char] ________
print(“Number of ‘d’s: ”.format(char_counts[“d”]))
print(“Number of ‘c’s: ”.format(char_counts.______(“c”,_______)))
```

anthony@MacBook:∼$ python example.py

```
Number of ‘d’s: 2
Number of ‘c’s: 0
```

Old Quiz Questions Review

Fill in the blank

example.py

```
text = “Teddy Bear”
char_counts = {}
for char in text:
        if char in char_counts:
                char_counts[char] += 1
        else:
                char_counts[char] = 1
print( “Number of ‘d’s: ”.format(char_counts[“d”]))
print( “Number of ‘c’s: ”.format(char_counts.get( “c”,0)))
```

```
anthony@MacBook:∼$ python example.py
```

```
Number of ‘d’s: 2
Number of ‘c’s: 0
```

Old Quiz Questions review - EC

Write the output of the code below. If it returns an error, write “ERROR” instead.

example.py

```
Item = {}
Item[‘name’] = Anthony
print(item[0])
```

```
anthony@MacBook:∼$ python example.py
```

Old Quiz Questions review - EC

Write the output of the code below. If it returns an error, write “**ERROR**” instead.

example.py

```
Item = {}
Item[‘name’] = Anthony
print(item[0])
```

```
anthony@MacBook:~$ python example.py
ERROR
```

Midterm Administrivia

Review

- Old Quiz Questions
- Old Quiz Questions
- Old Quiz Questions

Old Quiz Questions review

What is the output of the following Python program?

example.py

```
a = [3,4]
b = [1,2]
a[0] = b
a[1] = [1,2]
print(a[0] == a[1])
```

```
anthony@MacBook:~$ python example.py
```

Old Quiz Questions review

What is the output of the following Python program?

example.py

```
a = [3,4]
b = [1,2]
a[0] = b
a[1] = [1,2]
print(a[0] == a[1])
```

```
anthony@MacBook:~$ python example.py
True
```

Old Quiz Questions review

What is the output of the following Python program?

example.py

```
for i in range(1,9,2):
         if i%2 == 0 and i > 4:
                print(“Hello!”)
        else:
                print(“i: {}”.format(i))
```

```
anthony@MacBook:~$ python example.py
```

Old Quiz Questions review

What is the output of the following Python program?

example.py

```
for i in range(1,9,2): # 1 to 9 by 2
        if i%2 == 0 and i > 4: # Print Hello for even numbers > 4
              print(“Hello!”)
       else:
              print(“i: {}”.format(i))
```

```
anthony@MacBook:~$ python example.py
```

Old Quiz Questions review

What is the output of the following Python program?

example.py

```
for i in range(1,9,2): # 1 to 9 by 2
        if i%2 == 0 and i > 4: # Print Hello for even numbers > 4
                print( “Hello!” )
       else:
                print( “i: {}”.format(i))
```

```
anthony@MacBook:∼$ python example.py
1
3
5
7
```

Old Quiz Questions review - EC

Highest precedence

1. ( ) (anything in brackets is done first)
2. ** (exponentiation)
3. -x, +x
4. *, /,
5. +, -
6. relational operators: <, >, <=, >=, !=, ==
7. logical not
8. logical and
9. logical or

Lowest precedence

Old Quiz Questions review - EC

$1 + 2 * 3 = (1 + \mathbf{(2 * 3)})$
$= (\mathbf{1 + (6)})$
$= 7$

$17 / 2 * 3 + 2 = ((\mathbf{(17 / 2)} * 3) + 2)$
$= (\mathbf{((8.5) * 3)} + 2)$
$= (\mathbf{(25.5) + 2})$
$= 27.5$

Common Issues: Homework

example.py

```
def myFunction(myFileName):
        print(“test 1”)
        myFile = open(myFileName)
        return myFile.read() # Function ends here
        myFile.close() # This line is NEVER processed
        print(“test 2”) # This line is NEVER processed
myVar = myFunction(“testfile.txt”)
```

```
anthony@MacBook:~$ python example.py
test 1
```

Common Issues: Homework

example.py

```
# Goal count each sublist
myList = [[1,1,1],[1,1,1],[1,1,1]]
for subList in myList:
    mySum = 0
    for item in myList:
        mySum += item
    print(“Sum {}”.format(mySum))
```

```
anthony@MacBook:~$ python example.py
3
3
3
```

Common Issues: Homework

example.py

```
# Goal count each sublist
myList = [[1,1,1],[1,1,1],[1,1,1]]
# Sum keeps accumulating (NEVER reset)
mySum = 0
for subList in myList:
        for item in myList:
                mySum += item
        print(“Sum {}”.format(mySum))
```

anthony@MacBook:∼$ python example.py

```
3
6
9
```

Common Issues: Homework

example.py

```
text = “This is a test String ABCD.”
tokText = text.split()
tokText = [item.lower() for item in tokText] # A list comprehension
print(tokText)
```

```
anthony@MacBook:∼$ python example.py
[‘this’, ‘is’, ‘a’, ‘test’, ‘string’, ‘abcd.’]
```

Common Issues: Homework

example.py

```
text = “This is a test String ABCD.”
tokText = text.lower().split()
print(tokText)
```

```
anthony@MacBook:∼$ python example.py
[‘this’, ‘is’, ‘a’, ‘test’, ‘string’, ‘abcd.’]
```