

Name: _____ Section: 1 2

1 = Mutchler, 2nd–3rd periods. 2 = Mutchler, 4th–5th periods.

Use this quiz to help make sure you understand the videos/reading. **Answer all questions.** Make additional notes as desired. **Not sure of an answer?** Ask your instructor to explain in class and revise as needed then.

Please print two-sided if practical.

Throughout, where you are asked to “circle your choice”, you can circle or underline it (whichever you prefer).

Textbook Reading: Section 2.4 – Strings (pages 48 – 54)

1. What does `int('four')` evaluate to (or cause to happen)?

2. Consider the following

```
s = 'Nadima'
```

```
s.blah()
```

True or false: No matter what string method *blah* is, the value of `s` after the above executes will be `'Nadima'`. **True** **False** (circle your choice)

3. Write a *single* line of code that, given a variable `t` whose value is a string, re-assigns `t` so that its new value is that same string but in all upper-case.

4. Examine Appendix A of your textbook or google for a table of the ASCII values.

a. What is the integer that represents the question-mark character? _____

b. What character does the integer 70 represent? _____

c. What does `ord('?')` evaluate to? _____

d. What does `chr(70)` evaluate to? _____

e. What does `'g' < 'G'` evaluate to? _____

5. `'\n'` is the way we represent the _____ character in Python code.

6. Which is older, ASCII or Unicode? (circle your choice)

7. Which is more powerful in what it can represent, ASCII or Unicode? (circle your choice)

8. Python 3 strings are sequences of which: ASCII or Unicode characters? (circle your choice)

Online reading: *String Methods* from **Section 4.7.1** of the online Python documentation at <http://docs.python.org>, and relevant parts of *Chapter 8 Strings* from the online *Learning with Python 3*.

The remainder of this quiz consists of Python statements. For each statement:

- **Figure out what you think the statement would print**, as best you can guess from your reading or additional googling.
- To the right of the statement, write down your best guess for what would get printed.

A major part of this problem is practice finding information that is not in your textbook. Programmers spend a lot of time doing this kind of thing.

Note: It will be tempting to “just get the answers” by simply typing the code in a Python console instead of first trying to figure it out yourself. **This quiz will be mostly worthless to you** if that is what you do. In class, you will check your own answers from the file we supply that contains all these statements.

9. Consider:

```
s1 = 'the mighty Nile river'
```

Given the above, what would each of the following print?

```
print(s1.lower()) _____
```

```
print(s1.capitalize()) _____
```

```
print(s1) _____
```

```
print(s1.title()) _____
```

```
print(s1.replace('i', '')) _____
```

```
print(s1.count('i')) _____
```

```
print(s1.count('n')) _____
```

```
print(s1.lower().count('n')) _____
```

```
print(s1.endswith('ver')) _____
```

```
print(s1.find('ive')) _____
```

```
print(s1.find('i'), s1.find('ever')) _____
```

10. Consider:

```
s1 = 'the mighty Nile river'  
word_list = s1.split(' ')
```

Given the above, what would each of the following print?

```
print(word_list)
```

```
print('!!!'.join(word_list))
```

11. Consider:

```
s2 = 'abcd'
```

Given the above, what would each of the following print?

```
print(list(s2))
```

```
print('_'.join(list(s2)))
```

```
for c in s2:  
    print(c, end=':')  
print()
```

```
print(s2 < 'cde')
```

```
print(s2 < 'Olin')
```

```
print(s2 < 'Olin'.lower())
```

```
print('a' in 'apple')
```

```
print('pl' in 'apple')
```

```
print('pa' in 'apple')
```

```
print('' in 'apple')
```

```
print('apple' in 'apple')
```

12. Consider:

```
s2 = 'abcd'
```

```
s3 = s2.upper()
```

Given the above, what would each of the following print?

Hint: `==` checks whether two objects have the same “contents”, while `is` checks whether two objects are the same object, i.e., at the same place in memory.)

```
print(s2 == s3) _____
```

```
print(s2 == s3.lower()) _____
```

```
print([1, 2] == [1, 2]) # same contents? _____
```

```
print([1, 2] is [1, 2]) # same object? _____
```

```
print('abc' == 'abc') _____
```

```
print('abc' is 'abc') _____
```

```
print('cba' == 'abc') _____
```

```
print('*' + ' abc\n ' + '*') _____
```

```
print('*' + ' abc\n '.strip() + '*') _____
```

```
print('*' + s2.center(8) + '*') _____
```

```
print('*' + s2.ljust(8) + '*') _____
```

```
print('*' + s2.rjust(8) + '*') _____
```

13. Consider:

```
s4 = '1\n23\n456\n7890'
```

Given the above, what would each of the following print?

```
print(s4)
```

```
print(s4.splitlines())
```

```
print(s4.splitlines()[2])
```

```
print(s4.splitlines()[2][1])
```

14. Consider:

```
import math
```

```
m = math.sqrt(7)
```

Note that the square root of 7 is about 2.645751311064591. Given the above, what would each of the following print?

```
print ('The square root of {} is {}'.format(7, m))
```

```
print ('The square root of {:.5} is {:.3f}'.format(7, m))
```

```
print ('The square root of {} is {:.3f}'.format(7, m))
```

```
print ("The data is {} and {}".format(92.4, 88))
```

```
print ("The data is {1} and {0}".format(92.4, 88))
```
