**Starting Out with Python 4e (Gaddis)**

**Chapter 8 More About Strings**

**TRUE/FALSE**

1. You cannot use a **for** loop to iterate over the characters in a string.

ANS: F

2. Indexing works with both strings and lists.

ANS: T

3. In slicing, if the end index specifies a position beyond the end of the string, Python will use the length of the string instead.

ANS: T

4. Indexing of a string starts at **1** so the index of the first character is **1**, the index of the second character is **2**, and so forth.

ANS: F

5. The index **-1** identifies the last character of a string.

ANS: T

6. The following expression is valid:

**string[i] = 'i'**

ANS: F

7. The following code will display **'yes + no'**:

**mystr = 'yes'**

**yourstr = 'no'**

**mystr += yourstr**

**print(mystr)**

ANS: F

8. If the **+** operator is used on strings, it produces a string that is a combination of the two strings used as its operands.

ANS: T

9. When accessing each character in a string, such as for copying purposes, you would typically use a **while** loop.

ANS: F

10. If a whole paragraph is included in a single string, the **split()** method can be used to obtain a list of the sentences in the paragraph.

ANS: T

11. The **strip()** method returns a copy of the string with all the leading whitespace characters removed but does not remove trailing whitespace characters.

ANS: F

**MULTIPLE CHOICE**

1. What are the valid indexes for the string **'New York'**?

|  |  |
| --- | --- |
| a. | **0** through **7** |
| b. | **0** through **8** |
| c. | **-1** through **-8** |
| d. | **-1** through **6** |

ANS: A

2. What will be displayed after the following code executes?

**mystr = 'yes'**

**yourstr = 'no'**

**mystr += yourstr \* 2**

**print(mystr)**

|  |  |
| --- | --- |
| a. | **yes + no \* 2** |
| b. | **yes + no yes + no** |
| c. | **yesnono** |
| d. | **yesnoyesno** |

ANS: C

3. What will be assigned to the variable **s\_string** after the following code executes?

**special = '1357 Country Ln.'**

**s\_string = special[ :4]**

|  |  |
| --- | --- |
| a. | **'7'** |
| b. | **'1357'** |
| c. | **5** |
| d. | **'7 Country Ln.'** |

ANS: B

4. What will be assigned to the variable **s\_string** after the following code executes?

**special = '1357 Country Ln.'**

**s\_string = special[4:]**

|  |  |
| --- | --- |
| a. | **' Country Ln.'** |
| b. | **'1357'** |
| c. | **'Coun'** |
| d. | **'57 C'** |

ANS: A

5. What will be assigned to the variable **s\_string** after the following code executes?

**special = '1357 Country Ln.'**

**s\_string = special[-3:]**

|  |  |
| --- | --- |
| a. | **'135'** |
| b. | **'753'** |
| c. | **'Ln.'** |
| d. | **'y Ln'** |

ANS: C

6. What will be assigned to the variable **some\_nums** after the following code executes?

**special = '0123456789'**

**some\_nums = special[0:10:2]**

|  |  |
| --- | --- |
| a. | **'0123456789'** |
| b. | **'24682468'** |
| c. | **'02468'** |
| d. | **'02020202020202020202'** |

ANS: C

7. If the start index is \_\_\_\_\_\_\_\_\_\_ the end index, the slicing expression will return an empty string.

|  |  |
| --- | --- |
| a. | equal to |
| b. | less than |
| c. | greater than |
| d. | less than or equal to |

ANS: C

8. What is the return value of the string method **lstrip()**?

|  |  |
| --- | --- |
| a. | the string with all whitespaces removed |
| b. | the string with all leading whitespaces removed |
| c. | the string with all leading tabs removed |
| d. | the string with all leading spaces removed |

ANS: B

9. What is the first negative index in a string?

|  |  |
| --- | --- |
| a. | **0** |
| b. | **-1** |
| c. | **-0** |
| d. | the size of the string minus one |

ANS: B

10. What will be assigned to the string variable **pattern** after the following code executes?

**i = 3**

**pattern = 'z' \* (5 \* i)**

|  |  |
| --- | --- |
| a. | **'zzzzzzzzzzzzzzz'** |
| b. | **'zzzzz'** |
| c. | **'z \* 15'** |
| d. | Nothing; this code is invalid |

ANS: A

11. Which method would you use to determine whether a certain substring is present in a string?

|  |  |
| --- | --- |
| a. | **endswith(substring)** |
| b. | **find(substring)** |
| c. | **replace(string, substring)** |
| d. | **startswith(substring)** |

ANS: B

12. Which method would you use to determine whether a certain substring is the suffix of a string?

|  |  |
| --- | --- |
| a. | **endswith(substring)** |
| b. | **find(substring)** |
| c. | **replace(string, substring)** |
| d. | **startswith(substring)** |

ANS: A

13. What list will be referenced by the variable **list\_strip** after the following code executes?

**my\_string = '03/07/2018'**

**list\_strip = my\_string.split('/')**

|  |  |
| --- | --- |
| a. | **['3', '7', '2018']** |
| b. | **['03', '07', '2018']** |
| c. | **['3', '/', '7', '/', '2018']** |
| d. | **['03', '/', '07', '/', '2018']** |

ANS: B

14. What will be the value of the variable **string** after the following code executes?

**string = 'abcd'**

**string.upper()**

|  |  |
| --- | --- |
| a. | **'abcd'** |
| b. | **'ABCD'** |
| c. | **'Abcd'** |
| d. | Nothing; this code is invalid |

ANS: B

15. What will be the value of the variable **string** after the following code executes?

**string = 'Hello'**

**string += ' world!'**

|  |  |
| --- | --- |
| a. | **'Hello'** |
| b. | **' world!'** |
| c. | **'Hello world!'** |
| d. | Nothing; this code is invalid |

ANS: C

16. What will display after the following code executes?

**password = 'ILOVEPYTHON'**

**if password.isalpha():**

**print('Invalid, must contain one number.')**

**elif password.isdigit():**

**print('Invalid, must have one non-numeric character.')**

**elif password.isupper():**

**print('Invalid, cannot be all uppercase characters.')**

**else:**

**print('Your password is secure!')**

|  |  |
| --- | --- |
| a. | **Invalid, must contain one number.** |
| b. | **Invalid, must have one non-numeric character.** |
| c. | **Invalid, must contain one number.**  **Invalid, cannot be all uppercase characters.** |
| d. | **Your password is secure!** |

ANS: A

**COMPLETION**

1. Each character in a string has a(n) \_\_\_\_\_\_\_\_\_\_ which specifies its position in the string.

ANS: index

2. A(n) \_\_\_\_\_\_\_\_\_\_ exception will occur if you try to use an index that is out of range for a particular string.

ANS: **IndexError**

3. The **isalpha()** method returns \_\_\_\_\_\_\_\_\_\_ if the string contains only alphabetic characters and is at least one character in length.

ANS: **True**

4. A(n) \_\_\_\_\_\_\_\_\_\_ is a span of characters that are taken from within a string.

ANS: slice

5. When the operand on the left side of the **\*** symbol is a string and the operand on the right side is an integer, the **\*** becomes the \_\_\_\_\_\_\_\_\_\_\_ operator.

ANS: repetition

6. The third number in string slicing brackets represents the \_\_\_\_\_\_\_\_\_\_\_ value.

ANS: step

7. The \_\_\_\_\_\_\_\_\_\_ operator can be used to determine whether one string is contained in another string.

ANS: **in**

8. The \_\_\_\_\_\_\_\_\_\_ method returns **True** if the string contains only numeric digits.

ANS: **isdigit()**

9. The \_\_\_\_\_\_\_\_\_\_ method returns the list of the words in a string.

ANS: **split()**

10. The \_\_\_\_\_\_\_\_\_\_ method returns a copy of the string with all the alphabetic letters converted to lower case.

ANS: **lower()**