

PYTHON TEST - 2.2 (STRINGS IN PYTHON)

Total points 50/50 

Strings in Python

STUDENT NAME *

VIVA
.....

 1. Which of the following is a valid way to create a string in Python? * 1/1

- a) s = "Hello"
- b) s = 'Hello'
- c) s = ""Hello""
- d) All of the above



✓ 2. Strings in Python are: *

1/1

- a) Mutable
- b) Immutable
- c) Both
- d) None



✓ 3. What will print(len("Python")) output? *

1/1

- a) 5
- b) 6
- c) 7
- d) Error



✓ 4. Which operator is used for string concatenation in Python? *

1/1

- a) +
- b) *
- c) &
- d) %



✓ 5. Which operator is used for string repetition? *

1/1

- a) +
- b) *
- c) %
- d) &



✓ 6. What will `s = "Python"; print(s[0])` output? *

1/1

- a) P
- b) y
- c) n
- d) Error



✓ 7. What is the output of `"Python"[2:5]`? *

1/1

- a) tho
- b) yth
- c) tho
- d) tho (with space)



✓ 8. Which indexing is allowed in Python strings? *

1/1

- a) Positive indexing only
- b) Negative indexing only
- c) Both positive and negative indexing
- d) None

✓

✓ 9. What will "Python"[-1] return? *

1/1

- a) P
- b) n
- c) o
- d) Error

✓

✓ 10. What will "Python"[:3] return? *

1/1

- a) Pyt
- b) hon
- c) Pyth
- d) Py

✓

✓ 11. Which method converts a string into uppercase? *

1/1

- a) upper()
- b) uppercase()
- c) toUpperCase()
- d) capitalize()



✓ 12. "HELLO".lower() returns: *

1/1

- a) HELLO
- b) hello
- c) Hello
- d) Error



✓ 13. Which method removes whitespace from both ends of a string? *

1/1

- a) strip()
- b) lstrip()
- c)rstrip()
- d) trim()



✓ 14. "python".capitalize() will return: *

1/1

- a) python
- b) Python
- c) PYTHON
- d) Error



✓ 15. Which method is used to check if a string starts with a specific substring?

*1/1

- a) starts()
- b) beginswith()
- c) startswith()
- d) isstart()



✓ 16. "hello world".find("world") returns: *

1/1

- a) 5
- b) 6
- c) True
- d) Error



✓ 17. If substring is not found, .find() returns: *

1/1

- a) -1
- b) None
- c) 0
- d) Error

✓

✓ 18. Which method raises an error if substring not found? *

1/1

- a) find()
- b) index()
- c) search()
- d) match()

✓

✓ 19. "python".count("p") returns: *

1/1

- a) 0
- b) 1
- c) 2
- d) Error

✓

✓ 20. "banana".count("an") returns: *

1/1

- a) 1
- b) 2
- c) 3
- d) 0



✓ 21. "Hello {}".format("Python") gives: *

1/1

- a) Hello Python
- b) Hello {}
- c) Hello
- d) Error



✓ 22. f"Hello {'World'}" gives: *

1/1

- a) Hello World
- b) Hello {'World'}
- c) Error
- d) World Hello



✓ 23. Which placeholder is used in old-style string formatting? *

1/1

- a) {}
- b) %
- c) #
- d) \$



✓ 24. "%.2f" % 3.14159 returns: *

1/1

- a) 3.14
- b) 3.14159
- c) 3.142
- d) Error



✓ 25. Which is NOT a valid way of string formatting in Python? *

1/1

- a) f-strings
- b) .format()
- c) % formatting
- d) stringify()



✓ 26. "py" in "python" returns: *

1/1

- a) True
- b) False
- c) None
- d) Error



✓ 27. "Java" in "Python" returns: *

1/1

- a) True
- b) False
- c) None
- d) Error



✓ 28. "apple" < "banana" returns: *

1/1

- a) True
- b) False
- c) Error
- d) None



✓ 29. "Python" == "python" returns: *

1/1

- a) True
- b) False
- c) Error
- d) None



✓ 30. "abc" > "abd" returns: *

1/1

- a) True
- b) False
- c) Error
- d) None



✓ 31. "Hi"*3 returns: *

1/1

- a) HiHiHi
- b) Hi3
- c) Error
- d) Hi Hi Hi



✓ 32. `len("")` returns: *

1/1

- a) 0
- b) 1
- c) None
- d) Error

✓

✓ 33. `"".join(["a","b","c"])` returns: *

1/1

- a) abc
- b) a b c
- c) [abc]
- d) None

✓

✓ 34. `" ".join(["a","b","c"])` returns: *

1/1

- a) abc
- b) a b c
- c) Error
- d) [a b c]

✓

✓ 35. "hello".replace("l","x") returns: *

1/1

- a) hexxo
- b) hexlo
- c) hxxlo
- d) Error



✓ 36. Which method checks if all characters are digits? *

1/1

- a) isnum()
- b) isdigit()
- c) isdecimal()
- d) isalpha()



✓ 37. "abc123".isalnum() returns: *

1/1

- a) True
- b) False
- c) Error
- d) None



✓ 38. "123".isalpha() returns: *

1/1

- a) True
- b) False
- c) Error
- d) None



✓ 39. " ".isspace() returns: *

1/1

- a) True
- b) False
- c) Error
- d) None



✓ 40. "Python".swapcase() returns: *

1/1

- a) PYTHON
- b) python
- c) pYTHON
- d) Error



✓ 41. `print("a\nb")` will display: *

1/1

- a) ab
- b) a b
- c) a (newline) b
- d) Error



✓ 42. Raw strings in Python are created using: *

1/1

- a) "string"
- b) r"string"
- c) raw("string")
- d) str.raw("string")



✓ 43. What is the output of `print("C:\\new")`? *

1/1

- a) C:\new
- b) C:\new
- c) C:new
- d) Error



✓ 44. Which character is used for escape sequences? *

1/1

- a) /
- b) \
- c) %
- d) \$



✓ 45. Triple quotes are used for: *

1/1

- a) Single-line strings
- b) Multi-line strings
- c) Comments only
- d) None



✓ 46. `type("Python")` returns: *

1/1

- a) str
- b) string
- c) text
- d) chr



✓ 47. ord('A') returns: *

1/1

- a) 65
- b) 97
- c) 66
- d) Error

✓

✓ 48. chr(97) returns: *

1/1

- a) 'a'
- b) 'A'
- c) '97'
- d) Error

✓

✓ 49. "hello".title() returns: *

1/1

- a) hello
- b) Hello
- c) HELLO
- d) hELLO

✓

✓ 50. "hello".zfill(8) returns: *

1/1

a) 000hello



b) hello000

c) hello

d) Error

This content is neither created nor endorsed by Google. - [Contact form owner](#) - [Terms of Service](#) - [Privacy Policy](#).

Google Forms



