

Python MCQ (Multiple Choice Questions)

Here are 1000 MCQs on Python (Chapterwise).

1. Who developed Python Programming Language?

- a) Wick van Rossum
- b) Rasmus Lerdorf
- c) Guido van Rossum
- d) Niene Stom

[View Answer](#)

Answer: c

Explanation: Python language is designed by a Dutch programmer Guido van Rossum in the Netherlands.

2. Which type of Programming does Python support?

- a) object-oriented programming
- b) structured programming
- c) functional programming
- d) all of the mentioned

[View Answer](#)

Answer: d

Explanation: Python is an interpreted programming language, which supports object-oriented, structured, and functional programming.

3. Is Python case sensitive when dealing with identifiers?

- a) no
- b) yes
- c) machine dependent
- d) none of the mentioned

[View Answer](#)

Answer: b

Explanation: Case is always significant while dealing with identifiers in python.

4. Which of the following is the correct extension of the Python file?

- a) .python
- b) .pl
- c) .py
- d) .p

[View Answer](#)

Answer: c

Explanation: '.py' is the correct extension of the Python file. Python programs can be written in any text editor. To save these programs we need to save in files with file extension '.py'.

5. Is Python code compiled or interpreted?

- a) Python code is both compiled and interpreted
- b) Python code is neither compiled nor interpreted
- c) Python code is only compiled
- d) Python code is only interpreted

[View Answer](#)

Answer: a

Explanation: Many languages have been implemented using both compilers and interpreters, including C, Pascal, and Python.

advertisement

6. All keywords in Python are in _____

- a) Capitalized
- b) lower case
- c) UPPER CASE
- d) None of the mentioned

[View Answer](#)

Answer: d

Explanation: True, False and None are capitalized while the others are in lower case.

7. What will be the value of the following Python expression?

`4 + 3 % 5`

- a) 7
- b) 2
- c) 4
- d) 1

[View Answer](#)

Answer: a

Explanation: The order of precedence is: %, +. Hence the expression above, on simplification results in $4 + 3 = 7$. Hence the result is 7.

8. Which of the following is used to define a block of code in Python language?

- a) Indentation
- b) Key
- c) Brackets
- d) All of the mentioned

[View Answer](#)

9. Which keyword is used for function in Python language?

- a) Function
- b) def
- c) Fun
- d) Define

[View Answer](#)

Answer: b

Explanation: The def keyword is used to create, (or define) a function in python.

10. Which of the following character is used to give single-line comments in Python?

- a) //
- b) #
- c) !
- d) /*

[View Answer](#)

Answer: b

Explanation: To write single-line comments in Python use the Hash character (#) at the beginning of the line. It is also called number sign or pound sign. To write multi-line comments, close the text between triple quotes.

Example: """ comment

text """

11. What will be the output of the following Python code?

```
i = 1
while True:
    if i%3 == 0:
        break
    print(i)

    i += 1
```

- a) 1 2 3
- b) error
- c) 1 2
- d) none of the mentioned

[View Answer](#)

Answer: b

Explanation: SyntaxError, there shouldn't be a space between + and = in +=.

advertisement

12. Which of the following functions can help us to find the version of python that we are currently working on?

- a) sys.version(1)
- b) sys.version(0)
- c) sys.version()
- d) sys.version

[View Answer](#)

Answer: d

Explanation: The function sys.version can help us to find the version of python that we are currently working on. It also contains information on the build number and compiler used. For example, 3.5.2, 2.7.3 etc. this function also returns the current date, time, bits etc along with the version.

13. Python supports the creation of anonymous functions at runtime, using a construct called

-
- a) pi
 - b) anonymous
 - c) lambda
 - d) none of the mentioned

[View Answer](#)

Answer: c

Explanation: Python supports the creation of anonymous functions (i.e. functions that are not bound to a name) at runtime, using a construct called lambda. Lambda functions are restricted to a single expression. They can be used wherever normal functions can be used.

14. What is the order of precedence in python?

- a) Exponential, Parentheses, Multiplication, Division, Addition, Subtraction
- b) Exponential, Parentheses, Division, Multiplication, Addition, Subtraction
- c) Parentheses, Exponential, Multiplication, Division, Subtraction, Addition

d) Parentheses, Exponential, Multiplication, Division, Addition, Subtraction

[View Answer](#)

Answer: d

Explanation: For order of precedence, just remember this PEMDAS (similar to BODMAS).

15. What will be the output of the following Python code snippet if x=1?

advertisement

`x<<2`

- a) 4
- b) 2
- c) 1
- d) 8

[View Answer](#)

Answer: a

Explanation: The binary form of 1 is 0001. The expression `x<<2` implies we are performing bitwise left shift on x. This shift yields the value: 0100, which is the binary form of the number 4.

16. What does pip stand for python?

- a) Pip Installs Python
- b) Pip Installs Packages
- c) Preferred Installer Program

- d) All of the mentioned

[View Answer](#)

Answer: c

Explanation: pip is a package manager for python. Which is also called Preferred Installer Program.

17. Which of the following is true for variable names in Python?

- a) underscore and ampersand are the only two special characters allowed
- b) unlimited length
- c) all private members must have leading and trailing underscores
- d) none of the mentioned

[View Answer](#)

Answer: b

Explanation: Variable names can be of any length.

18. What are the values of the following Python expressions?

$2^{**}(3^{**}2)$

$(2^{**}3)^{**}2$

$2^{**}3^{**}2$

- a) 512, 64, 512
- b) 512, 512, 512
- c) 64, 512, 64
- d) 64, 64, 64

[View Answer](#)

Answer: a

Explanation: Expression 1 is evaluated as: $2^{**}9$, which is equal to 512. Expression 2 is evaluated as $8^{**}2$, which is equal to 64. The last expression is evaluated as $2^{**}(3^{**}2)$. This is because the associativity of ** operator is from right to left. Hence the result of the third expression is 512.

19. Which of the following is the truncation division operator in Python?

- a) |
- b) //
- c) /
- d) %

[View Answer](#)

Answer: b

Explanation: // is the operator for truncation division. It is called so because it returns only the integer part of the quotient, truncating the decimal part. For example: $20//3 = 6$.

20. What will be the output of the following Python code?

```
l=[1, 0, 2, 0, 'hello', '', []]  
list(filter(bool, l))
```

- a) [1, 0, 2, 'hello', '', []]
- b) Error
- c) [1, 2, 'hello']
- d) [1, 0, 2, 0, 'hello', '', []]

[View Answer](#)

Answer: c

Explanation: The code shown above returns a new list containing only those elements of the list l which do not amount to zero. Hence the output is: [1, 2, 'hello'].

21. Which of the following functions is a built-in function in python?

- a) factorial()
- b) print()
- c) seed()
- d) sqrt()

[View Answer](#)

Answer: b

Explanation: The function seed is a function which is present in the random module. The functions sqrt and factorial are a part of the math module. The print function is a built-in function which prints a value directly to the system output.

22. Which of the following is the use of id() function in python?

- a) Every object doesn't have a unique id
- b) Id returns the identity of the object
- c) All of the mentioned
- d) None of the mentioned

[View Answer](#)

23. The following python program can work with ___ parameters.

```
def f(x):
    def f1(*args, **kwargs):
        print("Sanfoundry")
        return x(*args, **kwargs)
    return f1
```

- a) any number of
- b) 0
- c) 1
- d) 2

[View Answer](#)

Answer: a

Explanation: The code shown above shows a general decorator which can work with any number of arguments.

24. What will be the output of the following Python function?

```
min(max(False,-3,-4), 2,7)
```

- a) -4
- b) -3
- c) 2
- d) False

[View Answer](#)

25. Which of the following is not a core data type in Python programming?

- a) Tuples
- b) Lists
- c) Class
- d) Dictionary

[View Answer](#)

Answer: c

Explanation: Class is a user-defined data type.

26. What will be the output of the following Python expression if x=56.236?

```
print("%.2f"%x)
```

- a) 56.236
- b) 56.23
- c) 56.0000
- d) 56.24

[View Answer](#)

Answer: d

Explanation: The expression shown above rounds off the given number to the number of decimal places specified. Since the expression given specifies rounding off to two decimal places, the output of this expression will be 56.24. Had the value been $x=56.234$ (last digit being any number less than 5), the output would have been 56.23.

27. Which of these is the definition for packages in Python?

- a) A set of main modules
- b) A folder of python modules
- c) A number of files containing Python definitions and statements
- d) A set of programs making use of Python modules

[View Answer](#)

Answer: b

Explanation: A folder of python programs is called as a package of modules.

28. What will be the output of the following Python function?

```
len(["hello", 2, 4, 6])
```

- a) Error
- b) 6
- c) 4
- d) 3

[View Answer](#)

Answer: c

Explanation: The function len() returns the length of the number of elements in the iterable. Therefore the output of the function shown above is 4.

29. What will be the output of the following Python code?

```
x = 'abcd'  
for i in x:
```

```
print(i.upper())
```

a)

a
B
C
D

b) a b c d
c) error
d)

A
B
C
D

View Answer

Answer: d

Explanation: The instance of the string returned by upper() is being printed.

30. What is the order of namespaces in which Python looks for an identifier?

- a) Python first searches the built-in namespace, then the global namespace and finally the local namespace
- b) Python first searches the built-in namespace, then the local namespace and finally the global namespace
- c) Python first searches the local namespace, then the global namespace and finally the built-in namespace
- d) Python first searches the global namespace, then the local namespace and finally the built-in namespace

View Answer

Answer: c

Explanation: Python first searches for the local, then the global and finally the built-in namespace

31. What will be the output of the following Python code snippet?

```
for i in [1, 2, 3, 4][::-1]:  
    print (i)
```

- a) 4 3 2 1
- b) error
- c) 1 2 3 4
- d) none of the mentioned

[View Answer](#)

Answer: a

Explanation: `[::-1]` reverses the list.

32. What will be the output of the following Python statement?

1. `>>>"a"+"bc"`

- a) bc
- b) abc
- c) a
- d) bca

[View Answer](#)

Answer: b

Explanation: `+` operator is concatenation operator.

33. Which function is called when the following Python program is executed?

```
f = foo()  
format(f)
```

- a) `str()`
- b) `format()`
- c) `__str__()`
- d) `__format__()`

[View Answer](#)

Answer: c

Explanation: Both `str(f)` and `format(f)` call `f.__str__()`.

34. Which one of the following is not a keyword in Python language?

- a) `pass`

- b) eval
- c) assert
- d) nonlocal

[View Answer](#)

Answer: b

Explanation: eval can be used as a variable.

35. What will be the output of the following Python code?

```

1. class tester:
2.     def __init__(self, id):
3.         self.id = str(id)
4.         id="224"
5.
6. >>>temp = tester(12)
7. >>>print(temp.id)

```

- a) 12
- b) 224
- c) None
- d) Error

[View Answer](#)

Answer: a

Explanation: Id in this case will be the attribute of the instance.

36. What will be the output of the following Python program?

```

def foo(x):
    x[0] = ['def']
    x[1] = ['abc']
    return id(x)
q = ['abc', 'def']
print(id(q) == foo(q))

```

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

[View Answer](#)

Answer: d

Explanation: The same object is modified in the function.

37. Which module in the python standard library parses options received from the command line?

- a) getarg
- b) getopt
- c) main
- d) os

[View Answer](#)

Answer: b

Explanation: getopt parses options received from the command line.

38. What will be the output of the following Python program?

```
z=set('abc')
z.add('san')
z.update(set(['p', 'q']))
z
```

- a) {'a', 'c', 'c', 'p', 'q', 's', 'a', 'n'}
- b) {'abc', 'p', 'q', 'san'}
- c) {'a', 'b', 'c', 'p', 'q', 'san'}
- d) {'a', 'b', 'c', ['p', 'q'], 'san'}

[View Answer](#)

Answer: c

Explanation: The code shown first adds the element 'san' to the set z. The set z is then updated and two more elements, namely, 'p' and 'q' are added to it. Hence the output is: {'a', 'b', 'c', 'p', 'q', 'san'}

39. What arithmetic operators cannot be used with strings in Python?

- a) *
- b) -
- c) +
- d) All of the mentioned

[View Answer](#)

Answer: b

Explanation: + is used to concatenate and * is used to multiply strings.

40. What will be the output of the following Python code?

```
print("abc. DEF".capitalize())
```

- a) Abc. def
- b) abc. def
- c) Abc. Def
- d) ABC. DEF

[View Answer](#)

Answer: a

Explanation: The first letter of the string is converted to uppercase and the others are converted to lowercase.

41. Which of the following statements is used to create an empty set in Python?

- a) ()
- b) []
- c) {}
- d) set()

[View Answer](#)

Answer: d

Explanation: {} creates a dictionary not a set. Only set() creates an empty set.

42. What will be the value of 'result' in following Python program?

```
list1 = [1,2,3,4]
list2 = [2,4,5,6]
list3 = [2,6,7,8]
result = list()
result.extend(i for i in list1 if i not in (list2+list3) and i not in result)
result.extend(i for i in list2 if i not in (list1+list3) and i not in result)
result.extend(i for i in list3 if i not in (list1+list2) and i not in result)
```

- a) [1, 3, 5, 7, 8]
- b) [1, 7, 8]
- c) [1, 2, 4, 7, 8]
- d) error

[View Answer](#)

Answer: a

Explanation: Here, 'result' is a list which is extending three times. When first time 'extend' function is called for 'result', the inner code generates a generator object, which is further used in 'extend' function. This generator object contains the values which are in 'list1' only (not in 'list2' and 'list3'). Same is happening in second and third call of 'extend' function in these generator object contains values only in 'list2' and 'list3' respectively.

So, 'result' variable will contain elements which are only in one list (not more than 1 list).

43. To add a new element to a list we use which Python command?

- a) list1.addEnd(5)
- b) list1.addLast(5)
- c) list1.append(5)
- d) list1.add(5)

[View Answer](#)

Answer: c

Explanation: We use the function append to add an element to the list.

44. What will be the output of the following Python code?

```
print('*', "abcde".center(6), '*', sep='')
```

- a) * abcde *
- b) *abcde *
- c) * abcde*
- d) * abcde *

[View Answer](#)

Answer: b

Explanation: Padding is done towards the right-hand-side first when the final string is of even length.

45. What will be the output of the following Python code?

```
1. >>>list1 = [1, 3]
2. >>>list2 = list1
3. >>>list1[0] = 4
4. >>>print(list2)
```

- a) [1, 4]
- b) [1, 3, 4]
- c) [4, 3]

d) [1, 3]

[View Answer](#)

Answer: c

Explanation: Lists should be copied by executing [:] operation.

46. Which one of the following is the use of function in python?

- a) Functions don't provide better modularity for your application
- b) you can't also create your own functions
- c) Functions are reusable pieces of programs
- d) All of the mentioned

[View Answer](#)

Answer: c

Explanation: Functions are reusable pieces of programs. They allow you to give a name to a block of statements, allowing you to run that block using the specified name anywhere in your program and any number of times.

47. Which of the following Python statements will result in the output: 6?

```
A = [[1, 2, 3],  
     [4, 5, 6],  
     [7, 8, 9]]
```

- a) A[2][1]
- b) A[1][2]
- c) A[3][2]
- d) A[2][3]

[View Answer](#)

Answer: b

Explanation: The output that is required is 6, that is, row 2, item 3. This position is represented by the statement: A[1][2].

48. What is the maximum possible length of an identifier in Python?

- a) 79 characters
- b) 31 characters
- c) 63 characters
- d) none of the mentioned

[View Answer](#)

Answer: d

Explanation: Identifiers can be of any length.

49. What will be the output of the following Python program?

```
i = 0
while i < 5:
    print(i)
    i += 1
    if i == 3:
        break
else:
    print(0)
```

- a) error
- b) 0 1 2 0
- c) 0 1 2
- d) none of the mentioned

[View Answer](#)

Answer: c

Explanation: The else part is not executed if control breaks out of the loop.

50. What will be the output of the following Python code?

```
x = 'abcd'
for i in range(len(x)):
    print(i)
```

- a) error
- b) 1 2 3 4
- c) a b c d
- d) 0 1 2 3

[View Answer](#)

Answer: d

Explanation: i takes values 0, 1, 2 and 3.

51. What are the two main types of functions in Python?

- a) System function
- b) Custom function
- c) Built-in function & User defined function

d) User function

[View Answer](#)

Answer: c

Explanation: Built-in functions and user defined ones. The built-in functions are part of the Python language. Examples are: dir(), len() or abs(). The user defined functions are functions created with the def keyword.

52. What will be the output of the following Python program?

```
1. def addItem(listParam):  
2.     listParam += [1]  
3.  
4. mylist = [1, 2, 3, 4]  
5. addItem(mylist)  
6. print(len(mylist))
```

- a) 5
- b) 8
- c) 2
- d) 1

[View Answer](#)

Answer: a

Explanation: + will append the element to the list.

53. Which of the following is a Python tuple?

- a) {1, 2, 3}
- b) {}
- c) [1, 2, 3]
- d) (1, 2, 3)

[View Answer](#)

Answer: d

Explanation: Tuples are represented with round brackets.

54. What will be the output of the following Python code snippet?

```
z=set('abc$de')  
'a' in z
```

- a) Error
- b) True
- c) False
- d) No output

[View Answer](#)

Answer: b

Explanation: The code shown above is used to check whether a particular item is a part of a given set or not. Since 'a' is a part of the set z, the output is true. Note that this code would result in an error in the absence of the quotes.

55. What will be the output of the following Python expression?

`round(4.576)`

- a) 4
- b) 4.6
- c) 5
- d) 4.5

[View Answer](#)

Answer: c

Explanation: This is a built-in function which rounds a number to give precision in decimal digits. In the above case, since the number of decimal places has not been specified, the decimal number is rounded off to a whole number. Hence the output will be 5.

56. Which of the following is a feature of Python DocString?

- a) In Python all functions should have a docstring
- b) Docstrings can be accessed by the `__doc__` attribute on objects
- c) It provides a convenient way of associating documentation with Python modules, functions, classes, and methods
- d) All of the mentioned

[View Answer](#)

Answer: d

Explanation: Python has a nifty feature called documentation strings, usually referred to by its shorter name docstrings. DocStrings are an important tool that you should make use of since it helps to document the program better and makes it easier to understand.

57. What will be the output of the following Python code?

```
print("Hello {0[0]} and {0[1]}".format(('foo', 'bin')))
```

- a) Hello ('foo', 'bin') and ('foo', 'bin')
- b) Error
- c) Hello foo and bin
- d) None of the mentioned

[View Answer](#)

Answer: c

Explanation: The elements of the tuple are accessed by their indices.

58. What is output of print(math.pow(3, 2))?

- a) 9.0
- b) None
- c) 9
- d) None of the mentioned

[View Answer](#)

Answer: a

Explanation: math.pow() returns a floating point number.

59. Which of the following is the use of id() function in python?

- a) Every object in Python doesn't have a unique id
- b) In Python Id function returns the identity of the object
- c) None of the mentioned
- d) All of the mentioned

[View Answer](#)

Answer: b

Explanation: Each object in Python has a unique id. The id() function returns the object's id.

60. What will be the output of the following Python code?

```
x = [[0], [1]]  
print(' '.join(list(map(str, x))))
```

- a) 01
- b) [0] [1]
- c) ('01')

d) ('[0] [1]',)

[View Answer](#)

Answer: d

Explanation: (element,) is not the same as element. It is a tuple with one item.

61. The process of pickling in Python includes _____
- a) conversion of a Python object hierarchy into byte stream
 - b) conversion of a datatable into a list
 - c) conversion of a byte stream into Python object hierarchy
 - d) conversion of a list into a datatable

[View Answer](#)

Answer: a

Explanation: Pickling is the process of serializing a Python object, that is, conversion of a Python object hierarchy into a byte stream. The reverse of this process is known as unpickling.

62. What will be the output of the following Python code?

```
def foo():
    try:
        return 1
    finally:
        return 2
k = foo()
print(k)
```

- a) error, there is more than one return statement in a single try-finally block
- b) 3
- c) 2
- d) 1

[View Answer](#)

Answer: c

Explanation: The finally block is executed even there is a return statement in the try block.

Chapterwise Multiple Choice Questions on Python



Our 1000+ MCQs focus on all topics of the Python Programming subject, covering 100+ topics. This will help you to prepare for exams, contests, online tests, quizzes, viva-voce, interviews, and certifications. You can also download the PDF of Python MCQs by applying below.

1. [Python Variable Names & Operators](#)
2. [Python Data Types & Numeric Types](#)
3. [Python Precedence and Associativity](#)
4. [Python Bitwise & Boolean](#)
5. [Python Formatting & Decorators](#)
6. [Python While and For Loops](#)
7. [Python Strings](#)
8. [Python Lists](#)
9. [Python List Comprehension](#)
10. [Python Tuples](#)
11. [Python Sets](#)
12. [Python Dictionary](#)
13. [Python Built-in Functions](#)
14. [Python Functions](#)
15. [Python Argument Passing, Variables and Recursion](#)
16. [Python Mapping Functions](#)
17. [Python Modules](#)
18. [Python Regular Expressions](#)
19. [Python Files](#)
20. [Python Classes and Objects](#)
21. [Python OOPs](#)
22. [Python Exception Handling](#)

1. Python MCQ on Variable Names & Operators

The section contains multiple choice questions and answers on python variables and basic operators.

- [Python Variable Names](#)
- [Python Basic Operators](#)

2. Python MCQ on Data Types & Numeric Types

The section contains questions and answers on python core data types and numeric types.

- [Python Core Data Types](#)
- [Python Numeric Types](#)

3. MCQ on Precedence and Associativity in Python

The section contains Python MCQ on precedence and associativity.

- [Python Precedence and Associativity-1](#)
- [Python Precedence and Associativity-2](#)

4. Python MCQ on Bitwise & Boolean

The section contains multiple choice questions and answers on python bitwise and boolean.

- [Python Bitwise – 1](#)
- [Python Bitwise – 2](#)
- [Python Boolean](#)

5. Multiple Choice Questions on Formatting & Decorators in Python

The section contains questions and answers on python formatting, advanced formatting and decorators.

- [Python Formatting – 1](#)
- [Python Formatting – 2](#)
- [Python Advanced Formatting](#)
- [Python Decorators](#)

6. Python MCQ on While and For Loops

The section contains Python MCQ on different types of loops (while and for loops).

- [Python While & For Loops – 1](#)
- [Python While & For Loops – 2](#)
- [Python While & For Loops – 3](#)
- [Python While & For Loops – 4](#)
- [Python While & For Loops – 5](#)
- [Python While & For Loops – 6](#)

7. Python MCQ on Strings

The section contains multiple choice questions and answers on different types of python strings.

- [Python Strings – 1](#)
- [Python Strings – 2](#)
- [Python Strings – 3](#)
- [Python Strings – 4](#)
- [Python Strings – 5](#)
- [Python Strings – 6](#)
- [Python Strings – 7](#)
- [Python Strings – 8](#)
- [Python Strings – 9](#)
- [Python Strings – 10](#)
- [Python Strings – 11](#)
- [Python Strings – 12](#)
- [Python Strings – 13](#)

8. Python Multiple Choice Questions on Lists

The section contains questions and answers on different types of lists.

- [Python Lists – 1](#)
- [Python Lists – 2](#)
- [Python Lists – 3](#)
- [Python Lists – 4](#)
- [Python Lists – 5](#)
- [Python Lists – 6](#)
- [Python Lists – 7](#)

9. Python MCQ on List Comprehension

The section contains MCQ on different types of list comprehension.

- [Python List Comprehension](#)
- [Python List Comprehension – 1](#)
- [Python List Comprehension – 2](#)
- [Python Matrix List Comprehension](#)

10. MCQ on Python Tuples

The section contains multiple choice questions and answers on various types of tuples.

- [Python Tuples – 1](#)
- [Python Tuples – 2](#)
- [Python Tuples – 3](#)

11. MCQ on Python Sets

The section contains questions and answers on various types of python sets.

- [Python Sets – 1](#)
- [Python Sets – 2](#)
- [Python Sets – 3](#)
- [Python Sets – 4](#)
- [Python Sets – 5](#)

12. Multiple Choice Questions on Python Dictionary

The section contains multiple choice questions and answers on python dictionary.

- [Python Dictionary – 1](#)
- [Python Dictionary – 2](#)
- [Python Dictionary – 3](#)
- [Python Dictionary – 4](#)

13. Python MCQ on Built-in Functions

The section contains multiple choice questions and answers on different types of python built in functions.

- [Python Built-in Functions – 1](#)
- [Python Built-in Functions – 2](#)
- [Python Built-in Functions – 3](#)

14. Multiple Choice Questions on Python Functions

The section contains questions and answers on different types of python functions.

- | | |
|---|---|
| <ul style="list-style-type: none">Python Function – 1Python Function – 2 | <ul style="list-style-type: none">Python Function – 3Python Function – 4 |
|---|---|

15. Python MCQ on Argument Passing, Variables and Recursion

The section contains Python MCQ on argument parsing, global and local variables, recursion, differences between shallow and deep copy.

- | | |
|--|--|
| <ul style="list-style-type: none">Python Argument Passing – 1Python Argument Passing – 2Python Global vs Local Variables – 1 | <ul style="list-style-type: none">Python Global vs Local Variables – 2Python RecursionPython Shallow Copy vs Deep Copy |
|--|--|

16. Python MCQ on Mapping Functions

The section contains Python multiple choice questions and answers on functional programming tools and mapping functions.

- | | |
|--|---|
| <ul style="list-style-type: none">Python Functional Programming ToolsPython Mapping Functions – 1 | <ul style="list-style-type: none">Python Mapping Functions – 2Python Mapping Functions – 3 |
|--|---|

17. Python MCQ on Modules

The section contains questions and answers on Python modules like math module, random module, sys module, datetime module, pickle and turtlre modules.

- | | |
|--|---|
| <ul style="list-style-type: none">Python ModulesPython Math – 1Python Math – 2Python Math – 3 | <ul style="list-style-type: none">Python Random Module – 2Python Sys ModulePython Operating SystemPython Turtle Module – 1 |
|--|---|

- [Python Datetime Module – 1](#)
- [Python Datetime Module – 2](#)
- [Python Random Module – 1](#)
- [Python Turtle Module – 2](#)
- [Python Turtle Module – 3](#)
- [Python Pickle Module](#)

18. Python MCQ on Regular Expressions

The section contains Python MCQ on different types of regular expressions (regex).

- | | |
|--|--|
| <ul style="list-style-type: none">• Python Regular Expressions• Python Regular Expressions – 1• Python Regular Expressions – 2 | <ul style="list-style-type: none">• Python Regular Expressions – 3• Python Regular Expressions – 4• Python Regular Expressions – 5 |
|--|--|

19. Python Multiple Choice Questions on Files

The section contains multiple choice questions and answers on python files.

- | | |
|--|---|
| <ul style="list-style-type: none">• Python Files – 1• Python Files – 2• Python Files – 3 | <ul style="list-style-type: none">• Python Files – 4• Python Files – 5 |
|--|---|

20. Python MCQ on Classes and Objects

The section contains questions and answers on operator overloading, classes and objects.

- | | |
|--|--|
| <ul style="list-style-type: none">• Python Operator Overloading• Python Classes and Objects – 1 | <ul style="list-style-type: none">• Python Classes and Objects – 2 |
|--|--|

21. Python MCQ on OOPs

The section contains MCQ on oops concepts like inheritance, polymorphism and encapsulation.

- | | |
|--|---|
| <ul style="list-style-type: none">• Python Inheritance – 1 | <ul style="list-style-type: none">• Python Polymorphism |
|--|---|

- [Python Inheritance – 2](#)

- [Python Encapsulation](#)

22. Python MCQ on Exception Handling

The section contains multiple choice questions and answers on exception handling.

- | | |
|---|---|
| <ul style="list-style-type: none">• Python Exception Handling – 1• Python Exception Handling – 2 | <ul style="list-style-type: none">• Python Exception Handling – 3 |
|---|---|

If you would like to learn "Python" thoroughly, you should attempt to work on the complete set of 1000+ MCQs - multiple choice questions and answers mentioned above. It will immensely help anyone trying to crack an exam or an interview.

Note: We are working on pdf download for Python MCQs and will publish the download link here. Fill this [Python mcq pdf download](#) request form for download notification.

[Wish you the best in your endeavor to learn and master Python!](#)

Important Links:

- [Java Multiple Choice Questions](#)
- [C++ Multiple Choice Questions](#)
- [C Multiple Choice Questions](#)
- [C# Multiple Choice Questions](#)
- [Data Structures Multiple Choice Questions](#)
- [Computer Science Multiple Choice Questions](#)

advertisement

Additional Resources:

- [C# MCQ PDF Download](#)
- [PHP MCQ PDF Download](#)
- [Ruby MCQ PDF Download](#)
- [Linked List Programs in Python](#)
- [C++ Programming MCQ PDF Download](#)

Popular Pages:

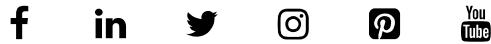
- [JavaScript MCQ Questions](#)
- [Searching and Sorting in Python](#)
- [Data Structure MCQ Questions](#)
- [Python Programs on Graphs](#)
- [Java Programming MCQ Questions](#)



[Manish Bhojwasi](#), a technology veteran with 20+ years @ Cisco & Wipro, is Founder and CTO at **Sanfoundry**. He lives in Bangalore, and focuses on development of Linux Kernel, SAN Technologies, Advanced C, Data Structures & Algorithms. Stay connected with him at [LinkedIn](#).

Subscribe to his free Masterclasses at [Youtube](#) & technical discussions at [Telegram SanfoundryClasses](#).

[About](#) | [Certifications](#) | [Internships](#) | [Jobs](#) | [Privacy Policy](#) | [Terms](#) | [Copyright](#) | [Contact](#)



© 2011-2023 Sanfoundry. All Rights Reserved.