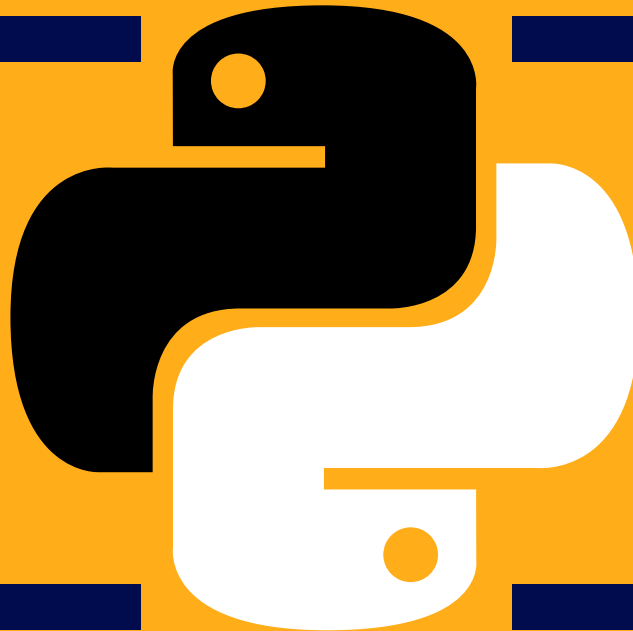


— CHIP SIZED'S —

250

A
B
C
D

MULTIPLE CHOICES



ALL IMPORTANT TOPICS COVERED
ANSWERS INCLUDED

Kindly Read This !!!

Thanks a lot for purchasing our book. We are thrilled to present this compilation of **250 engaging and thought-provoking Python quizzes** that are certain to challenge and enrich your understanding of this versatile programming language.

As **full-time freelancers** and ardent Python enthusiasts, we were clear on meticulously crafting the quizzes from scratch. Unlike **generic internet questions**, each quiz on these pages reflects our commitment to offering you a unique and enlightening learning experience.

Countless hours were invested in **researching, designing, and refining** each question to ensure that they test your knowledge and inspire a deeper exploration of **Python's concepts**. We wanted to provide a resource that caters to both beginners seeking to solidify their basics and seasoned developers aiming to challenge their expertise.

Your support gives us a lot of confidence. As **freelancers**, we've poured our passion into this book, and your appreciation fuels our drive to create more valuable content. If you find this book beneficial and wish to **extend your support**, we kindly invite you to consider supporting this **project financially** through our **Buy Me A Coffee** page. Every contribution, no matter how small, goes a long way in helping us continue our mission of spreading knowledge.

For any doubts and clarifications, we encourage you to connect with us on **LinkedIn**. We create content on different platforms like **Medium**, **Gumroad**, and **LinkedIn**. You can follow us through these links to consume more Python content. You can get my free **Python Handwritten Notes** from **Gumroad** as well which has over **2900 sales**. Get to know all our work through our **LinkTree** account.

We believe that learning should be an **immersive and enjoyable** experience, and we hope that this philosophy shines through in every quiz we've crafted. **Happy Coding !!!**

Thanks and Regards

Chip Sized

Table of Contents

01	Introduction to Python
02	The Print Function
03	Variables and Keywords - 01
04	Variables and Keywords - 02
05	Data Types - 01
06	Data Types - 02
07	Operators and Expressions
08	Decision-Making Statements
09	Looping Statements
10	Strings - 01
11	Strings - 02
12	Strings - 03
13	Lists - 01
14	Lists - 02
15	Lists - 03
16	Tuples
17	Sets
18	Dictionaries
19	User Defined Functions
20	Errors and Exceptions
21	Built-in Functions - 01
22	Built-in Functions - 02
23	Modules and Packages
24	Object Oriented Programming
25	File Handling
26	Correct Answers



Introduction to Python

01. Who is the founder of Python?

- A) Dennis Ritchie
- B) Bjarne Stroustrup
- C) Guido van Rossum
- D) James Gosling

02. Which of the following extensions is used in a Python file?

- A) .pyth
- B) .p
- C) .python
- D) .py

03. In which year was Python initially released?

- A) 1989
- B) 1991
- C) 1987
- D) 1994

04. Is Python compiled or interpreted?

- A) Interpreted
- B) Compiled
- C) Both Compiled and Interpreted
- D) None of the Above



05. There are ____ keywords in Python 3.11 (currently the latest version).

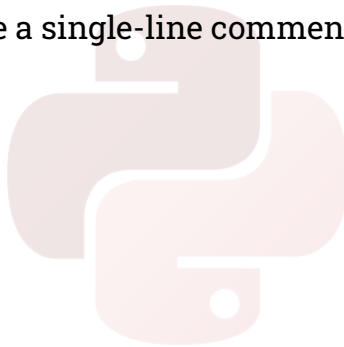
- A) 33
- B) 35
- C) 30
- D) 38

06. Which among the following is the full form of PIP?

- A) Python Index Packaging
- B) Preferred Installer Program
- C) Python Installer Program
- D) PIP Installs Python

07. Which symbol is used to write a single-line comment in Python?

- A) #
- B) !
- C) *
- D) /



08. Which of the following is not a built-in function in Python?

- A) `print()`
- B) `sqrt()`
- C) `ascii()`
- D) `sorted()`

09. Which of the following is an application of Python?

- A) Machine Learning
- B) Data Science
- C) Web Development
- D) All of the Above

10. The default IDE for Python is _____.

- A) PyCharm
- B) Visual Studio Code
- C) IDLE
- D) Jupyter Notebook



The Print Function

01. What is the output of the following code?

```
print('pyt.', 'hon')
```

- A) python
- B) pyt. hon
- C) pyt hon
- D) pyt.hon

02. What is the output of the following code?

```
print('10' + '20')
```

- A) 10 20
- B) 1020
- C) 30
- D) 10 + 20



03. Choose the correct code that would display the following output.

ABBBC

- A) `print('A' + 'B' + 'C', sep='B')`
- B) `print('ABB', 'BC')`
- C) `print('A', 'B', 'C', sep='B')`
- D) `print('A' + 'C', sep='BBB')`

04. What is the default value of the 'end' argument used in the `print()` function?

- A) `end= ' '`
- B) `end= '\n'`
- C) `end= ', '`
- D) `end= '\t'`

05. What is the output of the following code?

```
print("Hello World", sep='%')
```

- A) Hello World
- B) Hello World%
- C) Hello%World
- D) Error

06. What is the output of the following code?

```
print('code", 'blocks', sep='*', end='*')
```

- A) `code*blocks*`
- B) `code*blocks`
- C) `code blocks*`
- D) Error

07. Choose the correct code that would display the following output.

```
lemon+cake+
```

- A) `print('lemon' + 'cake', sep='+', end='+')`
- B) `print('lemon' + 'cake', end='+')`
- C) `print('lemon', 'cake', sep='+', end='+')`
- D) `print('lemon+', 'cake', end='+')`

08. Which of the following arguments is not a part of the `print()` function?

- A) `key`
- B) `flush`
- C) `end`
- D) `file`

09. Choose the correct code that would display the following output.

`c a t`

- A) `print('cat', sep=' ')`
- B) `print('c', 'a', 't', sep=',')`
- C) `print('c', 'a', 't', sep='')`
- D) `print('c', 'a', end=" t")`

10. What is the default value of the `'sep'` argument used in the `print()` function?

- A) `sep='\t'`
- B) `sep=','`
- C) `sep=' '`
- D) `sep='\n'`

Variables and Keywords - 01

01. Which of the following is an invalid variable name in Python?

- A) yield
- B) true
- C) C31
- D) a_b_c

02. What is the correct Python syntax to store the value 347 in a variable named n?

- A) n == 347
- B) 347 = n
- C) int n = 347
- D) n = 347

03. Which of the following is a valid variable name in Python?

- A) var%
- B) none
- C) pass
- D) 8num

04. What is the output of the following code?

```
a = 100
b = 200
b = a
a = b
print(b, a)
```

- A) 100 200
- B) 200 100
- C) 100 100
- D) 200 200

05. What is the output of the following code?

```
p = q  
q = 'hello'  
print(p)
```

- A) hello
- B) q
- C) Blank Output
- D) Error

06. Which of the following is not a keyword in Python?

- A) except
- B) final
- C) from
- D) global

07. What is the output of the following code?

```
r = 5 + 8  
print('r')
```

- A) 5 + 8
- B) r
- C) 13
- D) Error

08. How many keywords (as of Python 3.11) are not completely lowercase?

- A) 2
- B) 5
- C) 3
- D) 4

09. An _____ is a unique user-defined name assigned to a variable, function, or class.

- A) Iterator
- B) Identifier
- C) Literal
- D) Operator

10. What must be the input given to `m` in order to get `8` as the output?

```
m = int(input())  
n = m * 3  
p = n - 4  
q = p + m  
print(q)
```

- A) 3
- B) 5
- C) 8
- D) 0



Variables and Keywords - 02

01. Which of the following keywords is not used as an operator?

- A) is
- B) in
- C) or
- D) as

02. What is the syntax to assign the number 76 to variables `r`, `s`, and `t` using a single line?

- A) `r + s + t = 76`
- B) `r = s = t = 76`
- C) `r, s, t = 76`
- D) `r, s, t = 76 * 3`

03. What is the output of the following code?

```
x, y, z = 'y', 'z', 'x'
print(z, 'x', y)
```

- A) y z x
- B) x y z
- C) x x z
- D) y x z

04. Which of the following is a keyword in Python?

- A) default
- B) del
- C) false
- D) local

05. What is the output of the following code?

```
a, b, c = 3, 8, 5
d = b - c
b = d + a
a = b - c
print(d, a, b, c)
```

- A) 1 6 5 3
- B) 3 1 6 5
- C) 3 3 8 5
- D) 3 1 8 5

06. On executing the following code, how many objects has Python created?

```
val = 'code'
var = val
```



- A) 0
- B) 3
- C) 2
- D) 1

07. On executing the following code, how many references has Python created?

```
val = 758
var = val
```

- A) 2
- B) 0
- C) 3
- D) 1

08. What is the output of the following code?

```
c = 'choco'
d = 'bar'
c, d = d, c
print(c, d)
```

- A) bar choco
- B) choco choco
- C) bar bar
- D) choco bar

09. What is the output of the following code?

```
g = 'goat'
print(*g)
```

- A) g o a t
- B) g
- C) goat
- D) Error



10. Which of the following is a soft keyword as of Python 3.11?

- A) break
- B) pass
- C) case
- D) async

Data Types - 01

01. How is the dictionary data type represented in Python?

- A) dict
- B) dictionary
- C) dic
- D) dicts

02. Which of the following values is not a tuple?

- A) ('A')
- B) 10, 20
- C) ([10], [20])
- D) (10,)

03. Which of the following values displays an error if printed?

- A) {10, 20, 10}
- B) {1: 10, 2: 20, 1: 30}
- C) {6, [2, 3], 5}
- D) [1, ['Hello'], 1]

04. What is the output of the following code?

```
a = 00  
print(list(str(a)))
```

- A) ['00']
- B) ['0']
- C) ['0', '0']
- D) Error

05. What is the output of the following code?

```
d = {1: 'A', 2: 'B', 1: 'C'}  
print(d)
```

- A) {1: 'A', 2: 'B'}
- B) {1: 'A', 2: 'B', 1: 'C'}
- C) {1: 'C', 2: 'B'}
- D) Error

06. What is the output of the following code?

```
data = '20.0',  
Print(type(data))
```

- A) <class 'int'>
- B) <class 'float'>
- C) <class 'str'>
- D) <class 'tuple'>



07. Which of the following is a scientific notation of 0.095 in Python?

- A) 9.5e-3
- B) 9.5e3
- C) 95e-2
- D) 0.95e-1

08. Which of the following is an example of an implicit type conversion?

- A) String to List
- B) Integer to Float
- C) Tuple to List
- D) Float to Integer

09. What is the output of the following code?

```
r = 3  
s = 5  
print(str(s + r))
```

- A) 35
- B) 8
- C) 53
- D) Error

10. What is the output of the following code?

```
print(isinstance('true', bool))
```

- A) True
- B) False
- C) true
- D) Error



Data Types - 02

01. Which of the following is a mutable data type in Python?

- A) Integer
- B) Dictionary
- C) Complex
- D) String

02. Which of the following represents an empty set in Python?

- A) []
- B) set()
- C) {}
- D) ()

03. What is the output of the following code?

```
a = '01.01'  
print(int(a))
```

- A) 01
- B) 1
- C) 1.01
- D) Error

04. Which of the following data type conversion is not possible in Python?

- A) String to Integer
- B) List to Tuple
- C) List to Integer
- D) Complex to String

05. What is the output of the following code?

```
num = 6  
print(complex(num))
```

- A) 6.0
- B) 6j
- C) 6+0j
- D) Error

06. What is the output of the following code?

```
s = '123'  
print(list(s))
```

- A) [1, 2, 3]
- B) [123]
- C) ['1', '2', '3']
- D) ['123']



07. What is the output of the following code?

```
var = '100'  
print(int(var, 2))
```

- A) 100
- B) 2
- C) 4
- D) Error

08. Which value would become **False** when converted to a boolean value?

- A) [0]
- B) 1.0e0
- C) 1+0j
- D) (0)

09. What is the output of the following code?

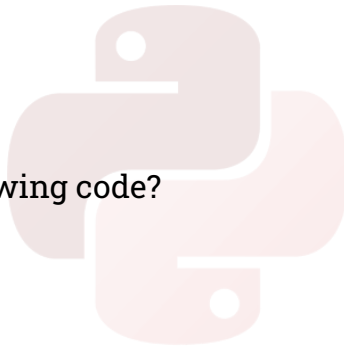
```
dic = {5: 10, 6: 12}  
print(tuple(dic))
```

- A) (10, 12)
- B) (5, 6)
- C) (5, 10, 6, 12)
- D) Error

10. What is the output of the following code?

```
b = True + True  
print(b)
```

- A) 2
- B) TrueTrue
- C) 1
- D) True



Operators and Expressions

01. Which of the following is not a comparison operator?

- A) !=
- B) <<
- C) >=
- D) ==

02. What is the output of the following code?

```
s = 'HdaPKwTb'
a = 'ab' not in s
b = 'p' in s
print(a, b)
```

- A) True False
- B) True True
- C) False False
- D) False True



03. What is the output of the following code?

```
x = 35
y = 40
x += y
y *= 2
print(x, y)
```

- A) 35 70
- B) 75 150
- C) 35 40
- D) 75 80

04. Which of the following denotes the **bitwise OR** operator?

- A) &
- B) or
- C) |
- D) =

05. Which of the following operators has right-to-left associativity in Python?

- A) **
- B) <
- C) not
- D) is

06. What is the output of the following code?

```
print(not 5<=3 or 4>4 and 8==8)
```

- A) 1
- B) True
- C) False
- D) 0

07. What is the output of the following code?

```
print(7 - 5 * (3 > 4) + 2)
```

- A) 0
- B) 2
- C) 9
- D) 5

08. Which among the following operators has the lowest precedence?

- A) **
- B) +
- C) and
- D) in

09. What is the output of the following code?

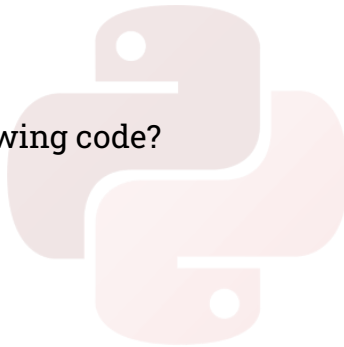
```
print(2 ** 3 ** 2)
```

- A) 512
- B) 64
- C) 256
- D) 16

10. What is the output of the following code?

```
m = [1, 2]
n = [1, 2]
o = 10
p = 10
print(m is n, o is p)
```

- A) True False
- B) True True
- C) False False
- D) False True



Decision-Making Statements

01. What is the recommended number of spaces used for indentation in Python?

- A) 2
- B) 4
- C) 1
- D) 8

02. What is the output of the following code if the input of `x` is 9?

```
x = input()
if x != 9:
    print(x, end=' ')
print(7)
```

- A) 7 9
- B) 9
- C) 7
- D) 9 7



03. What is the output of the following code?

```
k = 15
if k = 13:
    print(k + 2)
else:
    print(k - 2)
```

- A) 13
- B) 11
- C) 15
- D) Error

04. Which block is an optional alternative to the `if` block that must also have a condition?

- A) `else`
- B) `elif`
- C) `if`
- D) `case`

05. What is the output of the following code?

```
print(1 if 3 <= 1 else 3)
```

- A) `False`
- B) `3`
- C) `1`
- D) `Error`

06. What is the output of the following code?

```
a = 3
if a - 3:
    print(a * a)
else:
    print(a + a)
```

- A) `6`
- B) `0`
- C) `9`
- D) `Error`

07. Which character is used as a wildcard pattern in a `match-case` statement?

- A) Underscore (`_`)
- B) Asterisk (`*`)
- C) Pipeline (`|`)
- D) Percentage (`%`)

08. What is the output of the following code if the input of **m** is **2**?

```
m = int(input())
match m:
    case 1:
        print(3, end=' ')
    case 2:
        print(1, end=' ')
    case 3:
        print(2, end=' ')
print(m)
```

- A) 1 2
- B) 3 2
- C) 2 1
- D) 2 2

09. Fill the question mark with the correct condition to get the given output.

```
r = 25
if ?:
    print(r, end=' ')
print(r * 2)
```

Output:

25 50

- A) `r % 2 == 0`
- B) `r % 5 != 0`
- C) `r % 2 == 1`
- D) `r % 5 == 1`

10. What is the output of the following code?

```
b = int('D', 16)
match b - 1:
    case 11 | 14:
        print('Gold', end=' ')
    case 13 | 10:
        print('Silver', end=' ')
    case 12 | 15:
        print('Bronze', end=' ')
print(b)
```

- A) Bronze 13
- B) Gold 14
- C) Silver 13
- D) Bronze 12



Looping Statements

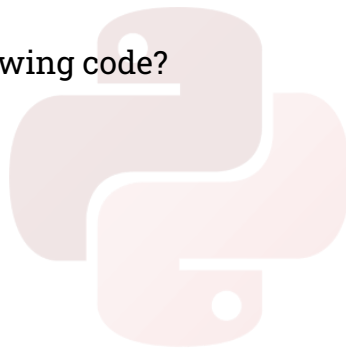
01. What is the output of the following code?

```
s = 4
while s < 9:
    s = s + 1
    print(s, end='-')
```

- A) 4-5-6-7-8-
- B) 5-6-7-8-
- C) 4-5-6-7-8-9-
- D) 5-6-7-8-9-

02. What is the output of the following code?

```
lis = [[8, 7], [6, 5]]
for p, q in lis:
    print(p + q, end='&')
```



- A) 15&11&
- B) 26&
- C) 14&12&
- D) Error

03. How many times does the given loop get executed?

```
for k in range(3, 9, 2):
    print(k, end=' ')
```

- A) 5
- B) 3
- C) 6
- D) 4

04. What is the output of the following code?

```
for i in range(1):  
    print(i, end=' ')
```

- A) 1
- B) 0 1
- C) Blank Output
- D) 0

05. Fill the question mark with the suitable code to get the given output.

```
for z in ?:  
    print(z, end=' ')
```

Output:

7 5

- A) range(7, 5, -2)
- B) range(4, 7, -2)
- C) range(5, 7, -2)
- D) range(7, 3, -2)



06. What is the output of the following code?

```
a = 6  
while a > 8:  
    print(a, end=' ')  
    a = a - 1
```

- A) Infinite Loop
- B) 6 7
- C) Blank Output
- D) Error

07. What is the output of the following code?

```
for x in range(3):  
    print(x, end=' ')  
    x = 3
```

- A) 0 3 3
- B) 0 1 2
- C) 0
- D) Error

08. What is the output of the following code?

```
lis = [10, 20, 30, 40]  
for m in lis:  
    print(m, end=' ')  
    if m >= 30:  
        break
```

- A) 10 20 30
- B) 10 20 30 40
- C) 10 20
- D) 10 20 40



09. Fill the question mark with the condition to make the loop run infinitely.

```
r = 0  
while ?:  
    print(r)  
    r = r + 2
```

- A) `r <= 0`
- B) `r > 0`
- C) `r >= 0`
- D) `r < 0`

10. Which keyword, when encountered, skips the current iteration in a loop?

- A) `continue`
- B) `else`
- C) `break`
- D) `pass`



Strings - 01

01. Which method is used to convert a string to uppercase?

- A) capital()
- B) caps()
- C) uppercase()
- D) upper()

02. Choose the correct code that would display the following output.

ZZZ

- A) print('Z', 'Z', 'Z')
- B) print(ZZZ)
- C) print('Z' * 'Z' * 'Z')
- D) print('Z' * 3)

03. What is the output of the following code?

```
m = 'chipsized'
print(m[-5], m[5])
```

- A) p i
- B) s i
- C) p s
- D) s s

04. Which of the following string methods does not return an integer?

- A) count()
- B) find()
- C) replace()
- D) index()

05. What is the output of the following code?

```
k = 'internet'
print(k[5:9])
```

- A) net
- B) netne
- C) rnet
- D) Error

06. What is the output of the following code?

```
i = 'ice'
print(f'{i} coffee')
```

- A) {i} coffee
- B) i coffee
- C) {ice} coffee
- D) ice coffee



07. Choose the correct code that would display the following output.

533

- A) `print('353533535'[6::-2])`
- B) `print('353533535'[::-4])`
- C) `print('353533535'[8:3:-2])`
- D) `print('353533535'[3:5])`

08. Which of the following code doesn't produce **ppl** as the output?

- A) `print('apple'[1:4])`
- B) `print('apple'[::-1][3:0:-1])`
- C) `print('apple'[-4:-2])`
- D) `print('apple'[1:7][:3])`

09. What is the output of the following code?

```
r = 'peace'  
print('y'.join(r))
```

- A) pyeyaycyey
- B) peacey
- C) pyeyaycye
- D) ypeace

10. What is the output of the following code?

```
m = 'code'  
print(m[1:3:2])
```

- A) ode
- B) o
- C) code
- D) oe



Strings - 02

01. What is the output of the following code?

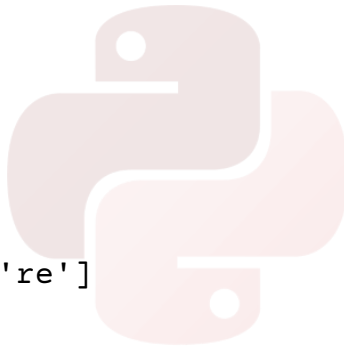
```
y = 'chocolate'
print(y[::-3][-3])
```

- A) o
- B) l
- C) c
- D) e

02. What is the output of the following code?

```
h = 'hardware'
print(h.split('a'))
```

- A) ['h', 'rdware']
- B) ['h', 'a', 'rdw', 'a', 're']
- C) ['h', 'rdw', 're']
- D) ['h', 'ardw', 'are']



03. What is the output of the following code?

```
t = 'programmers'
print(t.replace('r', 'l', 2))
```

- A) ploplammels
- B) ploplammers
- C) proplammers
- D) proplammels

04. Which of the following statements would print two backslashes to the output?

- A) `print('\2b')`
- B) `print('\\\\\\')`
- C) `print('\\\\')`
- D) `print('\\\\\\')`

05. What is the output of the following code?

```
s = 'welcome'
print(s.index('e', 3))
```

- A) 1
- B) 3
- C) 2
- D) 6

06. What is the output of the following code?

```
b = 'pythonista'
print(b[2:5:0])
```

- A) thon
- B) tho
- C) Blank Output
- D) Error

07. Which of the following code slices `iwd` from the following string?

```
j = 'sandwich'
```

- A) `j[-3:2:-1]`
- B) `j[-2:2:-1]`
- C) `j[-3:3:-1]`
- D) `j[-2:3:-1]`

08. What is the output of the following code?

```
a = 'bOokKEepEr'  
print(a.swapcase().count('e'))
```

- A) 1
- B) 3
- C) 0
- D) 2

09. What is the output of the following code?

```
n = 'python quiz'  
print(n.title())
```

- A) PYTHON QUIZ
- B) Python Quiz
- C) PYTHON quiz
- D) Python quiz



10. What is the output of the following code?

```
p = 'google'  
print(p[-7::2])
```

- A) log
- B) oge
- C) gol
- D) ego

Strings - 03

01. What is the output of the following code?

```
e = 'origin'
print(e.split('i', 1))
```

- A) ['orig', 'n']
- B) ['or', 'g', 'n']
- C) ['or', 'gin']
- D) Error

02. What is the output of the following code?

```
x = 'coding'
print(x[-4:4:-1])
```

- A) nid
- B) di
- C) ni
- D) Blank Output



03. What is the output of the following code?

```
y = 'start'
print(y.startswith('t', 1))
```

- A) True
- B) t
- C) 1
- D) False

04. Which of the following is not a string method?

- A) casefold()
- B) find()
- C) remove()
- D) center()

05. What is the output of the following code?

```
f = 'download'
f.replace('d', 'x')
print(f)
```

- A) download
- B) xownload
- C) xownloax
- D) Error

06. What is the output of the following code?

```
print('s'.join('run').count('s'))
```

- A) 1
- B) 2
- C) 3
- D) 4

07. What is the output of the following code?

```
print(r'chip\'s question')
```

- A) chip's question
- B) r'chip\'s question'
- C) chip\'s question
- D) Error

08. What is the output of the following code?

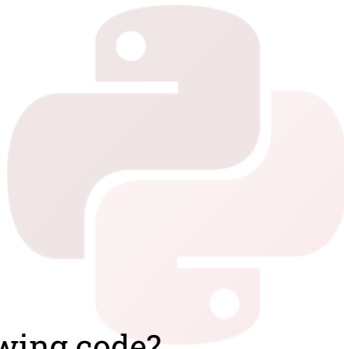
```
v = "650"  
print(v.zfill(4))
```

- A) 0065000
- B) 6500
- C) 0650
- D) 6500000

09. What is the output of the following code?

```
c = "cracked"  
print(c[1:5][1:3])
```

- A) rac
- B) ac
- C) rack
- D) ack



10. What is the output of the following code?

```
x = "developer"  
print(x.partition('o'))
```

- A) ('devel', 'o', 'per')
- B) ('devel', 'oper')
- C) ('devel', 'per')
- D) ('develo', 'per')

Lists - 01

01. What is the output of the following code?

```
k = ['A', 'B', ['C', 'D']]
print(k[:3][:2])
```

- A) ['C', 'D']
- B) ['A', 'B', ['C', 'D']]
- C) ['A', 'B', 'C']
- D) ['A', 'B']

02. What is the output of the following code?

```
m = [1, 2, 3, 4]
m.insert(2, 3)
print(m)
```

- A) [1, 2, 3, 3, 4]
- B) [1, 2, 3, 4]
- C) [1, 2, 3, 2, 4]
- D) [1, 2, 2, 3, 4]



03. What is the output of the following code?

```
c = [5, 9, 4]
c[1:2] = [6, 8, 3]
print(c)
```

- A) [5, 6, 4]
- B) [5, 6, 8, 3, 4]
- C) [5, 6, 8]
- D) Error

04. What is the output of the following code?

```
z = [2, 7, 6]
z.append([8, 9])
print(z)
```

- A) [2, 7, 6, 8, 9]
- B) [2, 8, 9]
- C) [2, 7, 6, [8, 9]]
- D) Error

05. What is the output of the following code?

```
r = [1, 3, 4, 3, 2]
print(r.index(3, 4))
```

- A) 3
- B) 2
- C) 1
- D) Error



06. What is the output of the following code?

```
i = [30, 40]
print(i + i)
```

- A) [30, 30, 40, 40]
- B) [70]
- C) [30, 40, 30, 40]
- D) Error

07. What is the output of the following code?

```
t = [13, 14, 15]
t.extend([16])
print(t)
```

- A) [13, 14, 15, [16]]
- B) [13, 14, 15, 16]
- C) [16]
- D) [13, 14, 16]

08. What is the output of the following code?

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

- A) [3, 1, 4]
- B) [0, 3, 1, 4, 0]
- C) [0, 3, 4, 0]
- D) Error



09. What is the output of the following code?

```
x = [43, 21, 92, 66, 80]
x.sort(reverse=True)
print(x)
```

- A) [92, 80, 66, 43, 21]
- B) [80, 66, 92, 21, 43]
- C) [21, 43, 66, 80, 92]
- D) [43, 21, 92, 66, 80]

10. Which concept is used in the first line of the given code?

```
p = [x * x for x in range(1, 6)]  
print(p)
```

- A) List Unpacking
- B) List Comprehension
- C) List Generation
- D) List Iteration



Lists - 02

01. Which of the following code accesses the value 3 from the given list?

```
h = [1, 2, [3, 4], 5]
```

- A) `h[2][0]`
- B) `h[3][1]`
- C) `h[3][2]`
- D) `h[2][1]`

02. What is the output of the following code?

```
g = [1, 2, 3, 2, 5]  
g.remove(2)  
print(g)
```

- A) `[1, 3, 2, 5]`
- B) `[1, 2, 2, 5]`
- C) `[1, 3, 5]`
- D) `[1, 2, 3, 5]`



03. What is the output of the following code?

```
n = [76, 24]  
p = n.copy()  
n.pop()  
print(p, n)
```

- A) `[76] [76, 24]`
- B) `[76, 24] [76, 24]`
- C) `[76] [76]`
- D) `[76, 24] [76]`

04. What is the output of the following code?

```
q = [47, 28, 33, 54, 15]
q.reverse()
print(q[:3])
```

- A) [33, 54, 15]
- B) [47, 28, 33]
- C) [33, 28, 47]
- D) [15, 54, 33]

05. What is the output of the following code?

```
k = [2, 1, 0, 3, 0, 2, 1]
print(k.count(k.index(0)))
```

- A) 2
- B) 1
- C) 0
- D) 3



06. What is the output of the following code?

```
a = [1, 2, 3, 4, 5]
print(a[:4].pop())
```

- A) [1, 2, 3, 4]
- B) 4
- C) [1, 2, 3, 5]
- D) 5

07. Fill the question mark with the suitable code to get the given output.

```
i = [10, 20, 30]
?  
print(i)
```

Output:

```
[10, 20, 30, [40, 50]]
```

- A) `i.append([40, 50])`
- B) `i.extend(40, 50)`
- C) `i.append(40, 50)`
- D) `i.extend([40, 50])`

08. Fill the question mark with the suitable code for the output to be 24.

```
d = [17, 14, 28, 24]  
print(?)
```

- A) `d.pop(24)`
- B) `d.remove(24)`
- C) `d.pop()`
- D) `d.remove(3)`

09. What is the output of the following code?

```
r = [20, 40, 60, 80]  
r[1:4] = []  
print(r)
```

- A) `[20, []]`
- B) `[20]`
- C) `[20, [], 60, 80]`
- D) Error

10. Which of the following list methods does not return **None**?

- A) `insert()`
- B) `remove()`
- C) `pop()`
- D) `extend()`



Lists - 03

01. What is the output of the following code?

```
b = [10, 20, 30]
print(b.append(40))
```

- A) [10, 20, 30]
- B) None
- C) [10, 20, 30, 40]
- D) Error

02. What is the output of the following code?

```
e = ['3']
e.extend('456')
print(e)
```

- A) ['3', '456']
- B) ['3456']
- C) ['3', '4', '5', '6']
- D) Error



03. What is the output of the following code?

```
j = [45]
k = j
k.append(32)
print(j, k)
```

- A) [45] [45, 32]
- B) [45, 32] [45, 32]
- C) [45] [45]
- D) [45, 32] [45]

04. What is the output of the following code?

```
m = ['a', 'C', 'B', 'b']  
m.sort()  
print(m)
```

- A) ['a', 'B', 'b', 'C']
- B) ['a', 'b', 'B', 'C']
- C) ['a', 'b', 'B', 'C']
- D) ['B', 'C', 'a', 'b']

05. What is the output of the following code?

```
y = [[1, 2]]  
print(y * 2)
```

- A) [1, 2, 1, 2]
- B) [[1, 2], [1, 2]]
- C) [1, 1, 2, 2]
- D) [[1, 1], [2, 2]]



06. Fill the question mark with the suitable code to get the given output.

```
c = [12, 14, 16, 18, 20]  
?  
print(c)
```

Output:

```
[12, 14, [24], 18, 20]
```

- A) `c[2:3] = [24]`
- B) `c[-3] = [24]`
- C) `c[2] = 24`
- D) `c[2:2] = [24]`

07. What is the output of the following code?

```
g = [8, 3, 9, 0, 7]
g.clear()
print(g)
```

- A) [8, 3, 9, 0]
- B) []
- C) Blank Output
- D) Error

08. What is the output of the following code?

```
s = [3, 2, 4, 5, 1]
x = s.pop(2)
print(s[-x])
```

- A) 4
- B) 2
- C) 5
- D) 3



09. Fill the question mark with the suitable code to get the given output.

```
a = [54, 65, 76]
?
print(a)
```

Output:

```
[54, 65, 76, 87]
```

- A) a.insert(5, 87)
- B) a.extend(87)
- C) a.insert(2, 87)
- D) a.append([87])

10. What is the output of the following code?

```
v = [10, 20, 30, 40]
v = [v[i]*i for i in range(4)]
print(v)
```

- A) [10, 40, 90, 160]
- B) [10, 20, 30, 40]
- C) [0, 20, 60, 120]
- D) [10, 40, 90, 120]



Tuples

01. What is the output of the following code?

```
b = 'awesome',  
print(b[:-2])
```

- A) aweso
- B) ()
- C) ('awesome',)
- D) me

02. What is the output of the following code?

```
f = (23, 34, 45, 56, 67)  
del f[1:4]  
print(f)
```

- A) (23, 67)
- B) (23)
- C) (56, 67)
- D) Error



03. What is the output of the following code?

```
r, *s, t = (1, 2, 3, 4, 5)  
print(s)
```

- A) [2, 3, 4]
- B) 2
- C) 4
- D) (2, 3, 4)

04. Fill the question mark with the suitable code to get the given output.

```
n = (4, 2, 3, 2, 4, 3)
print(?)
```

Output:

2

- A) `n.index(2, 2)`
- B) `n.index(2, 3)`
- C) `n.index(3, 3)`
- D) `n.index(3, 2)`

05. What is the output of the following code?

```
a = (4,)
print((a + a) * 2)
```



- A) `(4, 4, 4)`
- B) `(16)`
- C) `(4, 4, 4, 4)`
- D) `(4, 4)`

06. What is the output of the following code?

```
k = ([2], '2', (2))
print(k.count(2))
```

- A) 2
- B) 3
- C) 1
- D) 0

07. What is the output of the following code?

```
m = (2, 3, 4, 3, 1, 3)
print(m.index(3, 2, 4))
```

- A) 3
- B) 5
- C) 4
- D) 1

08. Fill the question mark with the suitable code to get the given output.

```
g = ([20, 40], [40, 20])
print(?)
```

Output:

[40]

- A) g[1][1:]
- B) g[0][1]
- C) g[:1][1]
- D) g[1][:1]



09. What is the output of the following code?

```
p = (10, 20, 30)
q = (10,)
print(p + q)
```

- A) (10, 20, 30, 10)
- B) (20, 30, 40)
- C) (10, 10, 20, 30)
- D) (20, 20, 30)

10. Which of the following is not a property of a tuple?

- A) Tuples can be Indexed
- B) Tuples are Mutable
- C) Tuples Allow Duplicate Values
- D) Tuples are Ordered



Sets

01. What is the output of the following code?

```
e = {'7', 5, '6', 5, '5'}  
print(e)
```

- A) {'7', '6', '5'}
- B) {'7', 5, '6'}
- C) {'7', 5, '6', 5, '5'}
- D) {'7', 5, '6', '5'}

02. What is the output of the following code?

```
g = {'A', 'B', 'C'}  
h = {'C', 'D', 'E'}  
print(h.difference(g))
```

- A) {'A', 'B', 'D', 'E'}
- B) {'A', 'B'}
- C) {'A', 'B', 'C', 'D', 'E'}
- D) {'D', 'E'}



03. What is the output of the following code?

```
k = {11, 22, 33, 22, 11}  
k.remove(22)  
print(k)
```

- A) {11, 33, 22}
- B) {11, 33, 22, 11}
- C) {11, 33}
- D) {11, 33, 11}

04. Fill the question mark with the suitable code to get the given output.

```
z = {'A', 'B', 'C'}  
?  
print(z)
```

Output:

```
{'A', 'C'}
```

- A) `z.symmetric_difference({'B'})`
- B) `z.intersection_update({'A', 'B', 'C'})`
- C) `z.symmetric_difference_update({'B'})`
- D) `z.difference_update({'A', 'C'})`

05. What is the output of the following code?

```
b = {'A', 'B'}  
c = {'A', 'C'}  
print(c.issuperset(b))
```

- A) False
- B) {'A', 'B'}
- C) True
- D) {'A', 'C'}

06. What is the output of the following code?

```
m = {3, 4, 5}  
m.discard('3')  
print(m)
```

- A) {3, 4, 5}
- B) {3}
- C) {4, 5}
- D) Error

07. Fill the question mark with the suitable code to get the given output.

```
a = {10, 20, 30}
?  
print(a)
```

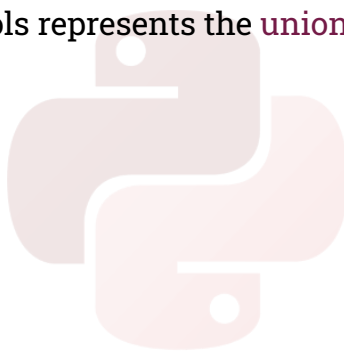
Output:

```
{10, 20, 30, 40, 50}
```

- A) `a.add({40, 50})`
- B) `a.update({40, 50})`
- C) `a.intersection({40, 50})`
- D) `a.union({40, 50})`

08. Which of the following symbols represents the **union** operation in a set?

- A) `^`
- B) `&`
- C) `|`
- D) `-`



09. Fill the question mark with the suitable code to get the given output.

```
r = {10, 20, 30}
s = {40, 50, 60}
print(?)
```

Output:

```
set()
```

- A) `r & s`
- B) `r - s`
- C) `r | s`
- D) `s - r`

10. What is the output of the following code?

```
f = {56, 98, 43, 67}
print(f.add(98))
```

- A) {56, 98, 43, 67}
- B) None
- C) {56, 98, 43, 67, 98}
- D) Error



Dictionaries

01. What is the output of the following code?

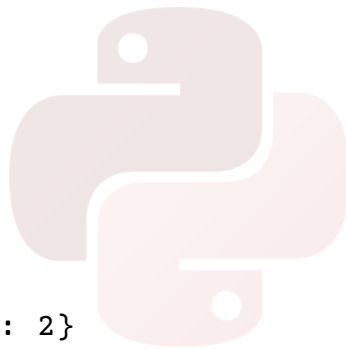
```
a = {1: 10, 2: 20, 3: 30, 2: 40}
print(a)
```

- A) {1: 10, 2: 40, 3: 30}
- B) {1: 10, 2: 20, 3: 30, 2: 40}
- C) {1: 10, 2: 20, 3: 30}
- D) {1: 10, 3: 30, 2: 40}

02. What is the output of the following code?

```
b = {[10]: 1, [20]: 2}
b.update({[30]: 2})
print(b)
```

- A) {[10]: 1, [20]: 2}
- B) {[10]: 1, [20]: 2, [30]: 2}
- C) {[10]: 1, [30]: 2}
- D) Error



03. What is the output of the following code?

```
c = {1: 2, 0: 4, 2: 8}
c[1] = 0
print(c)
```

- A) {1: 2, 0: 0, 2: 8}
- B) {0: 2, 2: 8}
- C) {1: 0, 0: 4, 2: 8}
- D) {1: 2, 0: 1, 2: 8}

04. Fill the question mark with the suitable code to get the given output.

```
d = {'A': 'C', 'C': 'B'}  
d.update({'B': 'A'})  
print(?)
```

Output:

B

- A) d.get('A')
- B) d[0]
- C) d['B']
- D) d.get('C')

05. What is the output of the following code?

```
e = {1: 23, 2: 89, 2: 66}  
print(e.pop(2))
```

- A) {1: 23}
- B) 66
- C) None
- D) 89

06. What is the output of the following code?

```
f = {4: 12, 2: 24, 3: 36}  
f.update({1: 24, 2: 48})  
print(f)
```

- A) {4: 12, 2: 48, 3: 36, 1: 24}
- B) {4: 12, 3: 36, 1: 24, 2: 48}
- C) {4: 12, 2: 24, 3: 36, 1: 24}
- D) Error

07. What is the output of the following code?

```
g = {1: 10, 2: 20, 3: 30}
h = g.copy()
h[3] = 40, 50
print(g)
```

- A) {1: 10, 2: 20, 3: 50}
- B) {1: 10, 2: 20, 3: (40, 50)}
- C) {1: 10, 2: 20, 3: 30}
- D) Error

08. Fill the question mark with the suitable code to get the given output.

```
k = {1: 1, 2: 4, 3: 9}
print(?)
```

Output:
(3, 9)

- A) k.pop(3)
- B) k.popitem()
- C) k.pop()
- D) k.popitem(3)



09. Fill the question mark with the suitable code for the output to be 3.

```
m = {1: {3: 2}, 1: {2: 3}}
print(?)
```

- A) m[1][3]
- B) m[1]
- C) m[2]
- D) m[1][2]

10. Which of the following is not a dictionary method?

- A) `iskey()`
- B) `values()`
- C) `keys()`
- D) `items()`



User Defined Functions

01. What is the output of the following code?

```
def a():  
    print(200, end=' ')  
  
print(400, end=' ')  
a()
```

- A) 400
- B) 200 400
- C) 400 200
- D) 200

02. Which of the following represents an arbitrary keyword argument?

- A) `**args`
- B) `**kwargs`
- C) `*args`
- D) `*kwargs`

03. What is the output of the following code?

```
def hi(name='Friend'):  
    print('Hi', name)  
  
hi('Jack')
```

- A) Hi Friend
- B) Hi Jack
- C) Hi Jack Friend
- D) Error

04. Which function call prints **10 20 30** as the output for the given function?

```
def num(x, y, z):  
    print(x, y, z)
```

- A) num(x=10, 20, 30)
- B) num(20, z=30, x=10)
- C) num(z=20, x=10, y=30)
- D) num(10, 20, z=30)

05. Fill the question mark with the suitable code to get the given output.

```
m = lambda x, y: y ** x  
print(?)
```

Output:

9

- A) m(x=3, y=2)
- B) m(3, 2)
- C) m(2, y=3)
- D) m(x=2, 3)



06. What is the output of the following code?

```
r('5', '3')  
  
def r(a, b):  
    print(int(b + a), end=' ')
```

- A) 53
- B) 8
- C) 35
- D) Error

07. What is the output of the following code?

```
def fn(b):  
    b.append(2)  
    print(b, end=' ')  
  
a = [1]  
print(a, end=' ')  
fn(a)  
print(a, end=' ')
```

- A) [1] [1, 2] [1, 2]
- B) [1] [1, 2] [1]
- C) [1] [2] [1]
- D) [1] [1] [1, 2]

08. A variable that can be accessed throughout a program is said to be a _____.

- A) default variable
- B) non-local variable
- C) local variable
- D) global variable

09. What is the output of the following code?

```
def func(p, q):  
    return p + q  
    print(p - q, end=' ')  
  
print(func(6, 2), end=' ')
```

- A) 8 4
- B) 8
- C) 4 8
- D) Error

10. What is the output of the following code?

```
def arb(*k):  
    print(k, end=' ')
```

```
arb(1, 7, 9, 5)
```

A) (1, 7, 9, 5)

B) 1

C) 5

D) [1, 7, 9, 5]



Errors and Exceptions

01. Which error would be displayed for the given code?

```
print(5 + 4 / 0)
```

- A) TypeError
- B) ZeroDivisionError
- C) ValueError
- D) NameError

02. Which block gets executed irrespective of whether an exception occurs or not?

- A) except
- B) else
- C) finally
- D) raise

03. Which of the following is not a **built-in error** in Python?

- A) StringError
- B) FileExistsError
- C) KeyError
- D) SystemError

04. Which error would be displayed for the given code?

```
print(g)  
g = 10
```

- A) ValueError
- B) TypeError
- C) SyntaxError
- D) NameError

05. What is the output of the following code if the input of **s** is **A**?

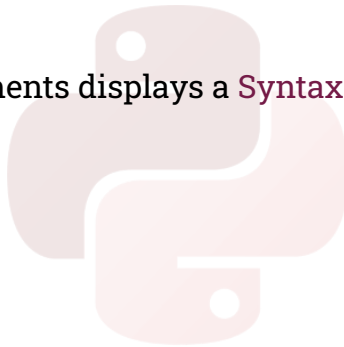
```
try:
    s = int(input())
    print(s, end=' ')
except:
    print('B', end=' ')

print('C', end=' ')
```

- A) B C
- B) C
- C) A B C
- D) A C

06. Which of the following statements displays a **SyntaxError**?

- A) `print(12B)`
- B) `print(list(8, 7, 6))`
- C) `print()`
- D) `print('2' - '3')`



07. Which error would be displayed for the given code?

```
for i in range(10):
    print(float(i))
```

- A) `FloatingPointError`
- B) `TypeError`
- C) `IndentationError`
- D) `SyntaxError`

08. Which of the following statements displays a **ValueError**?

- A) `print(a + b)`
- B) `print(int('4.5'))`
- C) `print(z*)`
- D) `print(list(10))`

09. What is the output of the following code?

```
try:
    print('5/0', end=' ')
except:
    print('Wrong', end=' ')
else:
    print('Done', end=' ')
```

- A) 5/0 Done
- B) 5/0
- C) Wrong
- D) Wrong Done



10. Which error would be displayed for the given code?

```
k = [10, 20, 30, 40, 50]
print(k[5])
```

- A) `ValueError`
- B) `TypeError`
- C) `KeyError`
- D) `IndexError`

Built-in Functions - 01

01. What is the output of the following code if the input of **m** is 5.63?

```
m = input()
print(type(m))
```

- A) <class 'tuple'>
- B) <class 'int'>
- C) <class 'str'>
- D) <class 'float'>

02. What is the output of the following code?

```
b = (2345)
print(len(b))
```

- A) 4
- B) 1
- C) 0
- D) Error



03. What is the output of the following code?

```
k = 'D'
print(chr(ord(k) + 30))
```

- A) b
- B) D
- C) d
- D) Error

04. What is the output of the following code?

```
p = [(), [], {}]  
print(all(p))
```

- A) True
- B) [(), [], {}]
- C) False
- D) []

05. What is the output of the following code?

```
r = 'quiz'  
print(sorted(r))
```

- A) ('i', 'q', 'u', 'z')
- B) ['i', 'q', 'u', 'z']
- C) iquz
- D) ('z', 'u', 'q', 'i')



06. What is the output of the following code?

```
a = '5678'  
print(sum(a, 4))
```

- A) 30
- B) 5682
- C) 26
- D) Error

07. What is the output of the following code?

```
print(pow(pow(2, 3), pow(0, 2)))
```

- A) 8
- B) 64
- C) 1
- D) 0

08. What is the output of the following code?

```
s = 'coding'  
x = slice(1, 4)  
print(s[x])
```

- A) odi
- B) odin
- C) din
- D) Error



09. What is the output of the following code?

```
y = [1, 2, 3]  
z = [4, 5]  
for i in zip(y, z):  
    print(i, end=' ')
```

- A) (1, 4) (2, 5)
- B) (1, 4) (2, 5) (3, 0)
- C) (1, 2, 3, 4, 5)
- D) (1, 4, 2, 5, 3)

10. What is the output of the following code?

```
c = ['01', '1', '2', '03']  
print(max(c[1:]))
```

- A) 01
- B) 2
- C) 1
- D) 03



Built-in Functions - 02

01. What is the output of the following code?

```
r = 154.953  
print(round(r, 4))
```

- A) 154.953
- B) 154.0000
- C) 154.9
- D) 154.9530

02. What is the output of the following code?

```
k = divmod(6, 3)  
print(k)
```

- A) (3, 2)
- B) (0, 2)
- C) (6, 0)
- D) (2, 0)



03. What is the output of the following code?

```
x = [10, 20, 30]  
print(sum(x, 10))
```

- A) 50
- B) 70
- C) 60
- D) 10

04. What is the output of the following code?

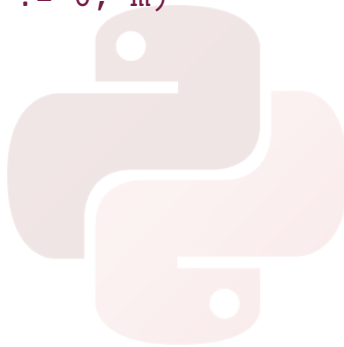
```
a = (10, '20', '30', 40)
print(min(a))
```

- A) 10
- B) 20
- C) 40
- D) Error

05. What is the output of the following code?

```
m = [78, 91, 83, 64, 55]
n = filter(lambda i: i % 2 != 0, m)
print(list(n))
```

- A) [91, 83, 64, 55]
- B) [78, 64]
- C) [91, 83, 55]
- D) [78, 83, 64, 55]



06. What is the output of the following code?

```
f = [45, 32, 95, 84]
g = sorted(reversed(f))
print(g)
```

- A) [32, 45, 84, 95]
- B) [84, 95, 32, 45]
- C) [45, 32, 95, 84]
- D) [95, 84, 45, 32]

07. What is the output of the following code?

```
z = ('0', 0+0j, 0)
print(any(z))
```

- A) False
- B) ('0',)
- C) True
- D) (0+0j, 0)

08. What is the output of the following code?

```
p = ['A', 'B', 'C']
q = map(ord, p)
print(list(q))
```

- A) [97, 98, 99]
- B) [1, 2, 3]
- C) ['A', 'B', 'C']
- D) [65, 66, 67]



09. What is the output of the following code?

```
s = [10, 20]
for x in enumerate(s, 1):
    print(x, end=' ')
```

- A) (0, 10) (1, 20)
- B) (1, 10) (2, 20)
- C) (0, 10)
- D) (0, 20)

10. Which of the following is not a built-in function?

- A) `hex()`
- B) `reduce()`
- C) `id()`
- D) `eval()`



Modules and Packages

01. Which of the following keywords is used for renaming a module?

- A) from
- B) or
- C) as
- D) else

02. Which statement imports the **function** named **add** from the **module** named **calc**?

- A) `import add from calc`
- B) `import calc.add`
- C) `from calc import add`
- D) `import from calc.add`

03. Fill the question mark with the suitable code to get **16.0** as the output.

```
import math
```

```
print(?)
```

- A) `pow(2, 4)`
- B) `math.pow(2, 4)`
- C) `math(pow(2, 4))`
- D) `sqrt(256)`

04. Which of the following extensions does a module file contain?

- A) `.pym`
- B) `.module`
- C) `.py`
- D) `.mod`

05. Which of the following modules is used for working with regular expressions?

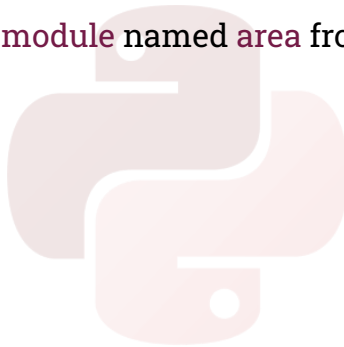
- A) `reg`
- B) `regex`
- C) `expr`
- D) `re`

06. A package folder must contain the _____ file to be recognized as a package.

- A) `__main__.py`
- B) `__init__.py`
- C) `__pack__.py`
- D) `__list__.py`

07. Which statement imports the **module** named **area** from the **package** named **rect**?

- A) `import area from rect`
- B) `from rect import area`
- C) `from area import rect`
- D) `import from rect.area`



08. Which command installs the third-party package named **requests**?

- A) `pip run requests`
- B) `install requests from pip`
- C) `pip install requests`
- D) `pip download requests`

09. Which is the official third-party software repository for Python?

- A) `PyPy`
- B) `PyLib`
- C) `PyPkg`
- D) `PyPI`

10. Which command updates the package named **numpy** to its latest version?

- A) `pip install --upgrade numpy`
- B) `pip --upgrade numpy`
- C) `pip --update numpy`
- D) `pip install --update numpy`



Object Oriented Programming

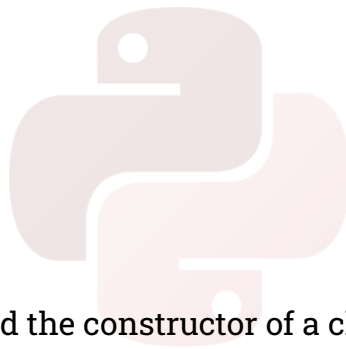
01. Which of the following is a blueprint from which **objects** are created?

- A) module
- B) class
- C) function
- D) method

02. What is the syntax for creating an object named **c** for the given class?

```
class Car:  
    pass
```

- A) `c = Car()`
- B) `Car c = Car()`
- C) `Car() = c`
- D) `c = Car(c)`



03. The _____ method is called the constructor of a class in Python.

- A) `__main__()`
- B) `__cons__()`
- C) `__self__()`
- D) `__init__()`

04. A method name prefixed with a single underscore is a _____ method.

- A) protected
- B) public
- C) private
- D) None of the Above

05. The default access modifier of a class is _____.

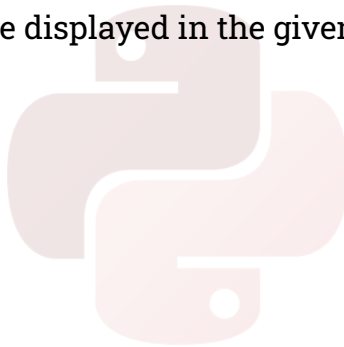
- A) public
- B) protected
- C) private
- D) None of the Above

06. Which decorator name is used to define a static method within a class?

- A) classmethod
- B) static
- C) staticmethod
- D) classdef

07. What is the type of inheritance displayed in the given code?

```
class M:  
    pass  
  
class N(M):  
    pass
```



- A) Multilevel Inheritance
- B) Single Inheritance
- C) Hierarchical Inheritance
- D) Multiple Inheritance

08. A class that inherits another class is called as _____.

- A) main class
- B) derived class
- C) base class
- D) abstract class

09. What is the type of inheritance displayed in the given code?

```
class A:  
    pass  
  
class B:  
    pass  
  
class C(A, B):  
    pass
```

- A) Multilevel Inheritance
- B) Single Inheritance
- C) Hierarchical Inheritance
- D) Multiple Inheritance

10. What is the type of inheritance displayed in the given code?

```
class R(T):  
    pass  
  
class S(R):  
    pass  
  
class T(S):  
    pass
```

- A) Multiple Inheritance
- B) Hierarchical Inheritance
- C) Multilevel Inheritance
- D) Error

File Handling

01. Which statement opens the file named `sum.txt` for writing?

- A) `f = open('w', 'sum.txt')`
- B) `f = open('a', 'sum.txt')`
- C) `f = open('sum.txt', 'w')`
- D) `f = open('sum.txt', 'r')`

02. Which error is displayed when a file that doesn't exist is opened in `read` mode?

- A) `FileNotExistsError`
- B) `FileExistsError`
- C) `FileError`
- D) `FileNotFoundError`

03. Which of the following is not a file method?

- A) `readable()`
- B) `append()`
- C) `flush()`
- D) `truncate()`

04. Which of the following methods is used to close a file?

- A) `exit()`
- B) `close()`
- C) `closefile()`
- D) `end()`



05. Which of the following represents the **create** file mode?

- A) x
- B) a
- C) r
- D) c

06. Which file mode displays an error when a file that already exists is opened?

- A) Create
- B) Read
- C) Write
- D) Append

07. Which statement reads the given file completely up to the end as a string?

```
f = open('a.txt')
```

- A) f.readline()
- B) f.seek()
- C) f.read()
- D) f.readlines()

08. Which keyword is used as a context manager for opening files?

- A) assert
- B) yield
- C) with
- D) import

09. Which file mode places the file pointer at the end of the file?

- A) Create
- B) Write
- C) Read
- D) Append

10. Which method is used to set the position of a file pointer?

- A) `tell()`
- B) `set()`
- C) `position()`
- D) `seek()`



Correct Answers

Introduction to Python

- 01. (C) Guido van Rossum
- 02. (D) .py
- 03. (B) 1991
- 04. (C) Both Compiled and Interpreted
- 05. (B) 35
- 06. (B) Preferred Installer Program
- 07. (A) #
- 08. (B) sqrt()
- 09. (D) All of the Above
- 10. (C) IDLE

The Print Function

- 01. (B) pyt. hon
- 02. (B) 1020
- 03. (C) print('A', 'B', 'C', sep='B')
- 04. (B) end='\n'
- 05. (A) Hello World
- 06. (D) Error
- 07. (C) print('lemon', 'cake', sep='+', end='+')
- 08. (A) key
- 09. (D) print('c', 'a', end=" t")
- 10. (C) sep=' '

Variables and Keywords - 01

- 01. (A) yield
- 02. (D) n = 347
- 03. (B) none
- 04. (C) 100 100
- 05. (D) Error
- 06. (B) final
- 07. (B) r
- 08. (C) 3

09. (B) Identifier

10. (A) 3

Variables and Keywords - 02

01. (D) as

02. (B) r = s = t = 76

03. (C) x x z

04. (B) del

05. (B) 3 1 6 5

06. (D) 1

07. (A) 2

08. (A) bar choco

09. (A) g o a t

10. (C) case

Data Types - 01

01. (A) dict

02. (A) ('A')

03. (C) {6, [2, 3], 5}

04. (B) ['0']

05. (C) {1: 'C', 2: 'B'}

06. (D) <class 'tuple'>

07. (D) 0.95e-1

08. (B) Integer to Float

09. (B) 8

10. (B) False



Data Types - 02

01. (B) Dictionary

02. (B) set()

03. (D) Error

04. (C) List to Integer

05. (C) 6+0j

06. (C) ['1', '2', '3']

07. (C) 4

08. (D) (0)

09. (B) (5, 6)

10. (A) 2

Operators and Expressions

01. (B) <<

02. (A) True False

03. (D) 75 80

04. (C) |

05. (A) **

06. (B) True

07. (C) 9

08. (C) and

09. (A) 512

10. (D) False True

Decision-Making Statements

01. (B) 4

02. (D) 9 7

03. (D) Error

04. (B) elif

05. (B) 3

06. (A) 6

07. (A) Underscore (_)

08. (A) 1 2

09. (C) $r \% 2 == 1$

10. (A) Bronze 13



Looping Statements

01. (D) 5-6-7-8-9-

02. (A) 15&11&

03. (B) 3

04. (D) 0

05. (D) `range(7, 3, -2)`

06. (C) Blank Output

07. (B) 0 1 2

08. (A) 10 20 30

- 09. (C) `r >= 0`
- 10. (A) `continue`

Strings - 01

- 01. (D) `upper()`
- 02. (D) `print('Z' * 3)`
- 03. (B) `s i`
- 04. (C) `replace()`
- 05. (A) `net`
- 06. (D) `ice coffee`
- 07. (B) `print('353533535'[::-4])`
- 08. (C) `print('apple'[-4:-2])`
- 09. (C) `pyeyaycye`
- 10. (B) `o`

Strings - 02

- 01. (D) `e`
- 02. (C) `['h', 'rdw', 're']`
- 03. (B) `ploglammers`
- 04. (B) `print('\\\\\')`
- 05. (D) `6`
- 06. (D) `Error`
- 07. (A) `j[-3:2:-1]`
- 08. (D) `2`
- 09. (B) `Python Quiz`
- 10. (C) `gol`

**Strings - 03**

- 01. (C) `['or', 'gin']`
- 02. (D) `Blank Output`
- 03. (A) `True`
- 04. (C) `remove()`
- 05. (A) `download`
- 06. (B) `2`
- 07. (C) `chip\'s question`
- 08. (C) `0650`

- 09. (B) ac
- 10. (A) ('devel', 'o', 'per')

Lists - 01

- 01. (D) ['A', 'B']
- 02. (A) [1, 2, 3, 3, 4]
- 03. (B) [5, 6, 8, 3, 4]
- 04. (C) [2, 7, 6, [8, 9]]
- 05. (D) Error
- 06. (C) [30, 40, 30, 40]
- 07. (B) [13, 14, 15, 16]
- 08. (D) Error
- 09. (A) [92, 80, 66, 43, 21]
- 10. (B) List Comprehension

Lists - 02

- 01. (A) h[2][0]
- 02. (A) [1, 3, 2, 5]
- 03. (D) [76, 24] [76]
- 04. (D) [15, 54, 33]
- 05. (A) 2
- 06. (B) 4
- 07. (A) i.append([40, 50])
- 08. (C) d.pop()
- 09. (B) [20]
- 10. (C) pop()

**Lists - 03**

- 01. (B) None
- 02. (C) ['3', '4', '5', '6']
- 03. (B) [45, 32] [45, 32]
- 04. (D) ['B', 'C', 'a', 'b']
- 05. (B) [[1, 2], [1, 2]]
- 06. (B) c[-3] = [24]
- 07. (B) []
- 08. (D) 3

- 09. (A) `a.insert(5, 87)`
- 10. (C) `[0, 20, 60, 120]`

Tuples

- 01. (B) `()`
- 02. (D) Error
- 03. (A) `[2, 3, 4]`
- 04. (D) `n.index(3, 2)`
- 05. (C) `(4, 4, 4, 4)`
- 06. (C) 1
- 07. (A) 3
- 08. (D) `g[1][:1]`
- 09. (A) `(10, 20, 30, 10)`
- 10. (B) Tuples are Mutable

Sets

- 01. (D) `{'7', 5, '6', '5'}`
- 02. (D) `{'D', 'E'}`
- 03. (C) `{11, 33}`
- 04. (C) `z.symmetric_difference_update({'B'})`
- 05. (A) False
- 06. (A) `{3, 4, 5}`
- 07. (B) `a.update({40, 50})`
- 08. (C) `|`
- 09. (A) `r & s`
- 10. (B) None

Dictionaries

- 01. (A) `{1: 10, 2: 40, 3: 30}`
- 02. (D) Error
- 03. (C) `{1: 0, 0: 4, 2: 8}`
- 04. (D) `d.get('C')`
- 05. (B) 66
- 06. (A) `{4: 12, 2: 48, 3: 36, 1: 24}`
- 07. (C) `{1: 10, 2: 20, 3: 30}`
- 08. (B) `k.popitem()`

09. (D) `m[1][2]`

10. (A) `iskey()`

User Defined Functions

01. (C) `400 200`

02. (B) `**kwargs`

03. (B) `Hi Jack`

04. (D) `num(10, 20, z=30)`

05. (C) `m(2, y=3)`

06. (D) `Error`

07. (A) `[1] [1, 2] [1, 2]`

08. (D) `global variable`

09. (B) `8`

10. (A) `(1, 7, 9, 5)`

Errors and Exceptions

01. (B) `ZeroDivisionError`

02. (C) `finally`

03. (A) `StringError`

04. (D) `NameError`

05. (A) `B C`

06. (A) `print(12B)`

07. (C) `IndentationError`

08. (B) `print(int('4.5'))`

09. (A) `5/0 Done`

10. (D) `IndexError`



Built-in Functions - 01

01. (C) `<class 'str'>`

02. (D) `Error`

03. (A) `b`

04. (C) `False`

05. (B) `['i', 'q', 'u', 'z']`

06. (D) `Error`

07. (C) `1`

08. (A) `odi`

09. (A) (1, 4) (2, 5)

10. (B) 2

Built-in Functions - 02

01. (A) 154.953

02. (D) (2, 0)

03. (B) 70

04. (D) Error

05. (C) [91, 83, 55]

06. (A) [32, 45, 84, 95]

07. (C) True

08. (D) [65, 66, 67]

09. (B) (1, 10) (2, 20)

10. (B) reduce()

Modules and Packages

01. (C) as

02. (C) from calc import add

03. (B) math.pow(2, 4)

04. (C) .py

05. (D) re

06. (B) __init__.py

07. (B) from rect import area

08. (C) pip install requests

09. (D) PyPI

10. (A) pip install --upgrade numpy



Object Oriented Programming

01. (B) class

02. (A) c = Car()

03. (D) __init__()

04. (A) protected

05. (A) public

06. (C) staticmethod

07. (B) Single Inheritance

08. (B) derived class

09. (D) Multiple Inheritance

10. (D) Error

File Handling

01. (C) `f = open('sum.txt', 'w')`

02. (D) `FileNotFoundError`

03. (B) `append()`

04. (B) `close()`

05. (A) `x`

06. (A) Create

07. (C) `f.read()`

08. (C) `with`

09. (D) Append

10. (D) `seek()`





