PythonInstitute.PCEP-30-02.v2025-12-04.q27

□□□□:	PCEP-30-02
□□□□:	PCEP - Certified Entry-Level Python Programmer
□□□:	Python Institute
□□ □□ □□□:	27
□□:	v2025-12-04
# □□ □:	105
# □□ □□□:	270
https://www.krdump.com/PythonInstitute.PCEP-30-02.v2025-12-04.q27.html	

NEW QUESTION: 1
□□□□ □□ □□□ □□□□ □□ □□ □□□□□?
A. □□□ 3□ □□□□□.
B. □□□ 2□ □□□□□.
C. □□□ □□ □□□ □□□□□.
D. □□□ 1□ □□□□□.
Answer: B (LEAVE A REPLY)
□□□□ □□ □□□ 0~3 □□□ □□ □□□□ "total" □□□ □□ □□□□□. □□□ □□□ □ □□□.
total = 0 for i in range(0, 3): if i % 2 == 0: total = total + 1 else: total = total + 2 print(total) □ □□ □ □□ "total"□ □ 0□ □□□□ □□□ □□□□□. □□ □□ □ 0, 1, 2□ □□□□ for □□□ □□□□□(range □□□ □□□ □□□□□). □□ □□□□ □□□ □□□ □□□(%)□ □□□ □ "i"□ □□ □□ □□□□ □□□□ □□□□□. "i"□ □□□□ □□□ "total" □□ 1□ □□□□. "i"□ □□□□ □□□ "total" □□ 2□ □□□□. "i"□ 3□ □□□□ □□□ □□□□ □□□ "total"□ □□ □□ □□□ □□□□□.
"total"□ □□ □□□ □□ □□□□□ □□□ □□□ 2□ □□□□□.
* i = 0□ □, total = 0 + 1 = 1
* i = 1□ □, □□ = 1 + 2 = 3
* i = 2□ □, □□ = 3 + 1 = 4
* i = 3□ □ □□□ □□□□ total = 4□ □□□□□.
□□□ □□□ B□□□. □□□ 2□ □□□□□.
□□: [Python Institute - □□ Python □□□□□ □□□]

NEW QUESTION: 2
□□ □□□ □□ □□□ □□□□□?
A. 5
B. 2

C. 1
D. □ □□□ □□□□ □□ □□□ □□□□□□.
Answer: (SHOW ANSWER)
□□
□□□□ □□ □□□ □□□ □□□□□□ □□□□ □□ □□□□ □ □□□ □□□□□ □□ □. □□□ □□□ □□□□.
my_list = [1, 2, 3, 4, 5] new_list = [x > 5□□ my_list□ □□ x□ □□ x]
□ □□□ □□ 1, 2, 3, 4, 5□ □□□□ "my_list"□□ □□□□ □□□□ □□□ □□□□□. □□ □□ □□□ □□□□□□ □□□□ "new_list"□□ □ □□□□ □□□□□ □□□. □□□ □□ □□□□ □□ □□□□ □ □□□ □□□□□ □□□ □□□□ □ □□□□ □□□□ □□□ □ □□□□. □□□ □□□□□□ □□□ □□□ □□□□.
new_list = [□□□ □□ old_list□ □□□ □□ □□□]
□□□□ □ □□□ □□□ □□□, □□□ □□□□□ □□□ □□□ □ □□□□. □□□ □□ □ □□□ □ □□ □□□□ □□□□□. □□□ □ □□□ □□□ □□□ □□□□ □□□ □□□□ □. □□ □□, □□ □□ □□□□ □□ □□□□ □□□ □□□ □□□□ □ □□□ □□□□.
old_list = [1, 2, 3, 4, 5, 6] new_list = [x ** 2 for x in old_list if x % 2 == 0] new_list = [4, 16, 36]□□ □□ □□□ 5□□ □ □□ □□□□ □□□ □□□□ □ □□□□ □□□□ □□□. □□□ □ □ □□□ □□□ □□□□. □□□ □□ □□□□ □□□ 5□□ □□ □□ □□□ □□□ □□ □□ □□□ □□□□. □, □ □□□□ □□ □□ □□ □□□□ □□□□ □□□□. □□□ □□□□ □□ x□ □□□□ □□ □□□□ □□□ □□□□ □□□□. □□ □□ NameError □□□ □□□ □□, □□ □□ □□□□ □□ □□□ □□ □ □□ □ □□□□ □□□□□. □□□ □□□ □□ □□ □□□□ □□ □□□□ □□ □□□□□.
□ □□□ □□ □□□ □□□□ □□ □□□□□. □□□ □□□□ □□ □□□ □□□□ □□ □ □□ □□□□□ □□ □□□□□. □□□ □□□ D□□□. □ □□□ □□□□ □□ □□□ □□ □□□□.

NEW QUESTION: 3
□□ □□ □ □□ □□ □□□□□? (□□ □ □□ □□□□□.)
A. □□□ □□ □□□□ □□□.
B. □□□ □□ □□□ □□□ □ □□□□.
C. □□ □□□ □□□□ □□ □□□□□ □□□.
D. □□ □□□ □□□ □ □ □□ □□□ □ □□□□.
Answer: B,C (LEAVE A REPLY)

NEW QUESTION: 4
□□ □□□ □□ □□□ □□□□□?
A. □□□ □□□ □□□□ □□□□.
B. * * *
C. * *
{<□>}: *

Answer: C (LEAVE A REPLY)

□□

□□□□ □□ □□□ □□ "counter"□ 0□□ □□□, 42□□ □□□ □□□, □□ □ □ □□□ □ □□□ □□□□□□. □□□ □□□ □□□□.

counter < 0: print("") elif counter >= 42: print("") else: print("") □ □□□ "counter"□ □□ 0□□ □ □□ □□□□ □□□ □□□□□. 0□□ □□□ □□□ □□() □□□ □□□□ □□□□ □□□ □□. 42□□ □□□ □□□ "counter"□ □□ 42□□ □□□ □□□ □□□□□. 42□□ □□□ □□□ □□□ □□() □ □□ □□□□ □□□□ □□□□□. 42□□ □□□ □□□ □□□ □□() □ □□ □□□□ □□□□ □□□□□.

□□□ □□ □□□ "counter" □□ □□ □□□□□. □□□ □□ "counter" □□ 10□□, □□□ □□□ □□ □ □(**)□ □□□□□. 10□ 0□□ □□□ □□ 42□□ □□□ □□□ □□ □□□□ □. □□□ □□□ C□□□. **

NEW QUESTION: 5

□□□□ □□ □□□ □□□□ □□ □□ □□□□□?

A. □□□□□ □ □□ □□(***)□ □□□ □□□□□.

B. □□□□□ □□□ □□(*) □□□ □□□□□.

C. □□□□□ □□ □□□ □□□□□.

D. □□□□□ □□□ □□ 5□(*****)□ □□□□□.

Answer: B (LEAVE A REPLY)

NEW QUESTION: 6

□□ □□□ 50.0□□ □□ □ □□ □□□□ □□□□□ □□□□ □□□ □□□□ □□□□ □ □ □□ □□□ □□□ □□□ □□□□□.

Answer:

□□

□□ □□□ 50.0□□ □□ □ □□ □□□□ □□□□□ □□□□ □□□ □□□ □□□□ □□ □□ □□□ □□□ □□□ □□□□ □ □□ □□□ □□□ □□□ □□□□.

□□□ 50.0 □□□ □□:

print("□□□ □□□□.")

□ □□□ if □□□□ □□□□ □□ □□ □□ □□□□ □□□□ □□□□□. □□ 50.0□□ □ □□ "The speed is low."□□ □□□□ □□□ □□□□□. print □□□ □□□ □□□□ □ □□ □□□. □ □□□ if □□□ □□□□ □□ □□□ □□□□ □□ □□□□□□ □□□□.

Python□ if □□ print □□□ □□ □□□ □□□ □□ □□ □□□□ □□□ □ □□□□.

□□□ If ... Else

□□□ □□ □□

NEW QUESTION: 7

□□ □□□ □□ □□□ □□□□□?

A. 2

B. 0
C. 3
D. 1
Answer: (SHOW ANSWER)
□□
□□□□ □□ □□□ count □□□□ □□□□ □□□□ □□ □□ □□ □□□ □□ □□□□□.
□□□ □□□ □□□□.
□_□□□ = [1, 2, 3, 4, 5] print(□_□□□.count(1))
□ □□□ □□ 1, 2, 3, 4, 5□ □□□□ "my_list"□□ □□□ □□□□ □□□ □□□□□. □□ □
□ print □□□ □□□□ □□ 1□ □□□□ □□□ count □□□□ □□□ □□□ □□□□□.
count □□□□ □□□ □□ □□□□ □□□ □□□□ □ □□□□□. □□ □□, my_list.count(1)
□ □□□ 1□ □ □ □□□□□ 1□ □□□□□.
□ □□□ □□ □□□ 1□□□. □ □□□ □□□□ 1□ □□□ □□□ □□□□ □□□□□.
□□□ □□□ D. 1□□□.

NEW QUESTION: 8
□□□□□ □□ □□ □□□□□ □□□□ □□ depth □□□ □□□□ □□ □□ □□□□ □□ □□ □□□ □□□ □□□ □□□□□.
(□□: □□ □□ □□□ □□□□ □□□□.)
Answer:
□□:
□□□□□ □□ □□ □□□□□ □□□□ □□ depth □□□ □□□□ □□ □□ □□□□ □□ □□ □□□ □□□ □□□ □□□□ □ □□ □□□ □□□ □□□ □□□□.
depth = int(input("□□ □□□ □□□□□: "))
□ □□ □□ input □□□ □□□□ □□□□□ □□□ □□ □□□□□ □□□ □□, int □□□ □□□□ □□ □□□ □□ □□□ □□□□□. □ □□□ depth □□□ □□□□□.
Python□ □□ □ int □□□ □□ □□□ □□□ □□ □□ □□□□ □□□ □ □□□□.
* [□□□ input() □□]
* [□□□ int() □□]

NEW QUESTION: 9
□□ □□□ □□ □□□ □□□□□?
A. □□□ □□□□ □□□ □ □□□□.
B. ppt
C. 213
D. □□□□□□□□
Answer: (SHOW ANSWER)
□□□□ □□ □□□ □□□□ □□□ □□□□ □□□□ □□□ □□□ □□□□ □□□□□.
□□□ □□□ □□□□.

pizza = "pizza" pasta = "pasta" folpetti = "folpetti" print(pizza[0] + pasta[0] + folpetti[0]) □ □□□ □□□ "pizza", "pasta", "folpetti"□ □□ pizza, pasta, folpetti □□□ □□□□ □□□ □□□□□. □□ □□ print □□□ □□□□ □ □□□□ □ □□□ □□□ □□□ □□□□□. □□□□ □ □ □□ □□□ □□ □□□ 0□ □□□□ □□□ □ □□□□. □□ □□, pizza[0]□ "p"□ □□□□ □. □□ □□□ + □□□□ □□□□ □ □ □□□ □□□□ □□□□ □ □□□□□. □□ □□, "a" + "b"□ "ab"□ □□□□□. □ □□□ pizza[0] + pasta[0] + folpetti[0]□ □□□ "p" + "p" + "f", □ "ppt"□ □□□□□.
□ □□□ □□ □□□ ppt□□□. □ □□□□ □ □□□ □□□□ □□□□□. □□□ □□□ B. ppt□□□.
□□: Python □□□ □□□□ - W3SchoolsPython □□□ □□ - W3Schools

NEW QUESTION: 10
□□□□□ □□ □□ □□□□□ □□□□ □□ depth □□□ □□□□ □□ □□ □□□□ □□ □□ □□□ □□□ □□□ □□□□□.
(□□: □□ □□ □□□ □□□□ □□□□.)
Answer:
□□
□□□□□ □□ □□ □□□□□ □□□□ □□ depth □□□ □□□□ □□ □□ □□□□ □□ □□ □□□ □□□ □□□ □□□□ □ □□ □□□ □□□ □□□ □□□□.
depth = int(input("□□ □□□ □□□□□: "))
□ □□ □□ input □□□ □□□□ □□□□□ □□□ □□ □□□□□ □□□ □□, int □□□ □□□□ □□ □□□ □□ □□□ □□□□□. □ □□□ depth □□□ □□□□□.
Python□ □□ □ int □□□ □□ □□□ □□□ □□ □□ □□□□ □□□ □ □□□□.
[□□□ input() □□]
[□□□ int() □□]

NEW QUESTION: 11
□□ □□□ □□□ □□ □□□ '□□ □□'□ □□□□ □□□□□ □□□ □□□.
(□□: □□ □□ □□□ □□□□ □□□□.)
Answer:

NEW QUESTION: 12
□□□□ □□ □□□ □□□□ □□ □□ □□□□□?
A. □□□ 3□ □□□□□.
B. □□□ 2□ □□□□□.
C. □□□ □□ □□□ □□□□□.
D. □□□ 1□ □□□□□.
Answer: B (LEAVE A REPLY)
□□

□□□□ □□ □□□ 0~3 □□□ □□ □□□□ "total" □□□ □□ □□□□□. □□□ □□□ □ □□□.

total = 0 for i in range(0, 3): if i % 2 == 0: total = total + 1 else: total = total + 2 print(total) □ □□ □ □□ "total"□ □ 0□ □□□□ □□□ □□□□□. □□ □□ □ 0, 1, 2□ □□□□ for □□□ □□□□□(range □□□ □□□□ □□□□□). □□ □□□□ □□□ □□□ □□□(%)□ □□ □□ "i"□ □□ □□ □□□□ □□□□ □□□□□. "i"□ □□□□ □□□ □□ 1□ □□□□. "total". "i"□ □□□□ □□□ "total" □□ 2□ □□□□. "i"□ 3□ □□□□ □□□ □□□□, □□ □ "total"□ □□ □□ □□□ □□□□□.

"total"□ □□ □□□ □□ □□□□□ □□□ □□□ 2□ □□□□□.

i = 0□ □, total = 0 + 1 = 1

i = 1□ □, total = 1 + 2 = 3

i = 2□ □, □□ = 3 + 1 = 4

i = 3□ □ □□□ □□□□ total = 4□ □□□□□.

□□□ □□□ B□□□. □□□ 2□ □□□□□.

NEW QUESTION: 13

□□ □□□ □ 0□ □□ □□□ □□□□ □□□□ □□□□□? (□□ □ □□ □□□□□.)

A. 2 ** 3 / A - 2

B. 4 / 2 * * 3 - 2

C. 1 * * 3 / 4 - 1

D. 1 * 4 // 2 ** 3

Answer: A,B (LEAVE A REPLY)

□□

Python□□□ ** □□□□ □□ □□□ □□□□ / □□□□ □□ □□□ □□□□ □□□□ // □□□□ □□ □□□□ □□□□□. □□ □□□ □□, □□, □□/□□□, □□/□□□□□. □ □□ □□□□ □□□ □□ □□□ □ □□□□.

A). 2 ** 3 / A - 2 = 8 / A - 2 (A□ 0□ □□□□ □□□□ □□ □□□□ □□)

B). 4 / 2 * * 3 - 2 = 4 / 8 - 2 = 0.5 - 2 = -1.5 C. 1 * * 3 / 4 - 1 = 1 / 4 - 1 = 0.25 - 1 = -0.75 D. 1 * 4 // 2 ** 3 = 4 // 8 = 0 □□□ A□ B□ 0□ □□ □□□ □□□□□.

NEW QUESTION: 14

□□ □□□ □□ □□□ □□□□□?

A. 5

B. 4

C. 6

D. □ □□□ □□□ □□□□□ □□□□ □□□□ □□□□.

Answer: D (LEAVE A REPLY)

□□□ □□ □□ □□□ "□□□"□ "□□□" □ □□□ □□□ □□□ □□□□□ □□□.

"□□". □□□ □□□ □□□□.

collection = [] collection.append(1) collection.insert(0, 2) duplicate = collection
duplicate.append(3) print(len(collection) + len(duplicate)) □□□ "collection"□□□ □ □□□□
□□□□ □□ 1□ □□□□ □□□ □□□□□. □□ □□□□□ [1]□ □□□□□. □□ □□, □
□□□ □□ □□□ □□ 2□ □□□□□. □□ □□□□□ [2, 1]□ □□□□□.
□□ □□ □□□ "duplicate"□□ □ □□□ □□□ "collection" □□ □□□□□. □□□ □ □□□
□□□ □□□□ □□□□ □□ □□□ □□□ □□ □□□ □□ □□□ □□□□□. □□□ □□
□ □□□ □□ □□ □□□
"duplicate"□ "collection"□□ □□□ □□□, □ □□□ □□□ □□□□□□□. □□ □□ □□□
"duplicate"□ □□ 3□ □□□□□.
□□ □□□□ [2, 1, 3]□ □□□□ "collection"□ □□□□□. □□□□□ □□□ "collection"□
"duplicate"□ □□□ □□ □□□□□ □□□□□. □□□ len □□□ □ □□ □□ □□□ □□□
□□□□ □□□ □□□ □□□□□. □ □□□ □□□ □□□□ □□□□ □□□□ □□□□ □
□□□.
□ □□□ □□ □□□ □□□□ □□□ □□□□. □□□ □□□ □□□□□ □□□□ □□□□
□. □□□ □□□ D□□□. □ □□□ □□□ □□□□□ □□□□ □□□□ □□□□.
□□: [Python Institute - □□ Python □□□□□ □□□]

NEW QUESTION: 15
□□□□□ □□ □□ □□□□□ □□□□ □□ □□ □□□ □□□ □□□□ □□ □□ □□□
□ □□ □□ □□□ □□□ □□□ □□□□□.
(□□: □□ □□ □□□ □□□□ □□□□.)
Answer:

NEW QUESTION: 16
□□□□□ □□ □□□ □□ □□□□ □□ □□ □□□ □□□□ □□ □□ □□□□ □□ □□
□□□ □□□ □□□ □□□□□.
(□□: □□ □□ □□□ □□□□ □□□□.)
Answer:
□□
□□□□□ □□ □□□ □□ □□□□ □□ □□ □□□ □□□□ □□ □□ □□□□ □□ □□
□□□ □□□ □□□ □□□□ □ □□ □□□ □□□ □□□ □□□□.
□□ = float(input("□□□ □□□□□:
□ □□ □□ input □□□ □□□□ □□□□□ □□□ □□ □□□□□ □□□ □□, float □□□
□□□□ □□ □□□ □□ □□ □□□ □□□ □□□□□. □□ □□ □ □□□ mass □□□ □
□□□□.
Python□ □□ □ float □□□ □□ □□□ □□□ □□ □□ □□□□ □□□ □ □□□□.
[□□□ input() □□]
[□□□ float() □□]

NEW QUESTION: 17

□□ □□□ □□□□ □□□□ □□□ □□□□□?

A. □□□ 1□ □□□□□.

B. □□□ 2□ □□□□□.

C. □ □□□ □□□□ □□ □□□ □□□□□□.

D. □□□ 0□ □□□□□.

Answer: C (LEAVE A REPLY)

□□□□ □□□□ □□ □□□ index □□□□ □□□□ □□□□ □□ □□□ □□□□ □□□.

□□□ □□□ □□□□.

the_list = [1, 2, 3, 4, 5] print(the_list.index(6))

□ □□□ □□ 1, 2, 3, 4, 5□ □□□□ "the_list"□□ □□□□ □□□□ □□□ □□□□□. □□ □□, □□ 6□ □□□□ □□□□ index □□□□ □□□ □□□ □□□□□. index □□□□ □□ □□□ □ □□□ □□□□ □□ □□□□ □ □□□□□. □□ □□, the_list.index(1)□ 1□ □□ □□ □ □□ □□□□ 0□ □□□□□.

□□□ □ □□□□ □□□ □□□□. □□□ □ 6□ □□□ □□□ index □□□□ □□ □□ □□ □ □□□ □□□□. □□ □□ ValueError □□□ □□□□□, □□ □□□ □□□ □□□ □□□ □□ □□□□ □□ □□ □□□ □□ □ □□□□ □□□□□. □ □□□ □ □□□ □□□□ □□ □□ □□□□ □□ □□□□□.

□ □□□ □□ □□□ □□□□ □□ □□□□□. □□□ □□□ □□ □□ □□□□ □□ □□□ □□. □□□ □□□ C□□□. □ □□□ □□□□ □□ □□□ □□□□□□.

□□: Python List index() □□□ - W3SchoolsPython □□: □□ - □□ Python

NEW QUESTION: 18

□□□ □□ □□□ □□ □□ □□□□□? (□□ □ □□ □□□□□.)

A. □□□ □□ □□□□□□. □, □□□ □□□ □□ □□ □□□ □ □□□□.

B. len { } □□□ □□□ □□□ □ □□□□.

C. □ □□□ { }□ □□□□□.

D. □□□ □□□□□ □□□ □ □□□□□ □□□□□.

Answer: A,D (LEAVE A REPLY)

□□

□□□ □□□□ □□□ □□□ □□ □ □□□, □□□ □□□ □□□□ □ □□□□□. □□□ □ □□, □□, □□□□□ □□ □□ □□□ □□□ □□□□ □ □□ □□□ □□□ □□□□. □□ □ □□□ □□□ □□□□.

□□□ □□□□□□. □, □□ □□ □□□ □□□ □ □□□□. □□□ □□□□ □□, □□ □□ □□□ □ □□□□. □ □□□ □□□ □□ □□□ □□□□ □□□□□ □□□ □ □□□□. □□ □ □□ □□□ □□ □□□ □□□□ □□□□ □□□ □□□□□ □□ □□□□□ □□□. □□ □□ □□□ □□□ □□□□□ □□□ □□□ □ □□□ □□□ □□ □□□ □□□□ □□□. 12 □□□ □□□□ □□□□. □, □□□ □□□□ □□□ □□□ □□□ □□□□ □□□□ □□□ □ □ □□□□. □□□ □□□□ □ □□ □□□ □□ 0□□ □□□□ □□□ □□□ □□ □□ □ □□□ 1□ □ □□□□□□. □□□□ □□□ □□ □□□, □ □□ □□□ □□□□□ □□□□ □. □□ □□, □□ t = ("a", "b", "c")□ □□ □□, t[0]□ "a"□ □□□□ t[-1]□ "c"□ □□□□□. □ □□ □□□□□ □□□ □ □□□□□ □□□□□. □, □□□□ □□□□ □□ □ □ □□□□ □ □□□ □□□ □□ □□ □□ □□ □□□□ □□□ □ □□□□. □□ □□, □□ t□ □□ □□ = ("a", "b", "c", "d", "e")□□ t[2]□ "c"□ □□□□ t[1:4]□ ("b", "c", "d")□ □□□□□. □□ □□ □ □□□□ □□□ □□□□□□ □□□□□ □□□ □□□□□ □□□□. □ □□□□ □□ □□□ □□ □□□□ □□□□□.12 □□□ □□□, □□, □□, □□□, □□, □□ □□ □□ □□ □ □ □ □□□ □□□ □□□ □ □□□□. □□□ □□ □□ □□ □□ □□□, □□ □□□ □□□ □ □□ □ □ □□ □□□ □ □□□ □□□□□. □□ □□, □ □□ 1□ □ □□ 2□12

□□□ □□ □□□ □□□□□. □, □□□ □ □□□ □□□ □□□ □□□. □□ □□, □□ □□ □ □□□□ □□ t = ("a", "b", "c")□ □□ □ □□□□. □□□ □□ □□□ □□□□ □□ □□□ □□□ □□□□ □□□ □□ □□ □□□□. □□ □□, □□□ □□□□ □□ □□ t = "a", "b", "c"□ □□ □ □□□□. □□ □□ □□□□□ □□, □□□ □□□ □□ □□ □□□ □ □□□ □.12 len() □□□ □□□□ □□□ □ □□□□. □, len() □□□ □□□□ □□□ □□ □□ □□ □ □□□□. □□ □□, □□ t = ("a", "b", "c")□ □□□ len(t)□ 312□ □□□□□. □ □□□ ()□ □□□□□, □□ □□□ □□ □□□ □□□□□ □ □□ □□ □□□□ □□□ □□ □□□□□. □□ □□, □ □□□ □□□□ □ □□ t = ()□ □□□ □ □□□□.

□□□ □□□ □□□ □□ □□□ □□□□ □□ □□ □□□ □□□□ □□□. □□□ □□□ Python□ □□ □□□ □□□□ □□□□. □□ □□, □□□ □□□□ □□□ □□□ □□ □□ t = ("a",)□ □□ □ □□□□. □□□ □□□ A□□□.

□□□ □□ □□□□□□. □, □□□ □□□ □□ □□ □□□ □ □□□□. D. □□□ □□□□ □□□ □ □□□□□ □□□□□.

NEW QUESTION: 19

□□□ □□□□ □□□□ □□□ □□□□ □□ □□□ □□□ □□ □□□□.

A. □□

B. □□

C. □□□

D. □□

Answer: (SHOW ANSWER)

□□□□ □□□ □□□ □□□ □□□ □□□□ □□□□ □ □□□□□. □□□□ □□□ □□ □, □□□□ □□□ □□□ □□□□□. □□□ □□□ □ □□, □□□, □□ □□ □□□□ □□ □□.

□□: [Python Institute - □□ Python □□□□□ □□□]

NEW QUESTION: 20
□□□ □□□ □ 1, 3, 5(□□□ □□)□ □□□□ □□□ □□□□ □□□ □□ □□ □□ □□□ □□□ □□□ □□□□□.
Answer:
□□ □ □□□(1, 7, 2):
□□:
* □ □□
* □□□
* □□
* □□
* (
* 1
* ,
* 7
* ,
* 2
*)
□□ □□□ □□ □□□□ □□□□□.
□□□ □□ □□□□ 1 # 3 # 5□ □□□□□.

NEW QUESTION: 21
□□ □ Python□ □□ □□ □□□□ □□□ □□□□□?
(□□ □ □□ □□□□□.)
A. □□□
B. □□
C. □□□
D. □□
Answer: A,D (LEAVE A REPLY)
□□
□□□ □□□ □□□□ □□ □□ □□(=)□ □□ □□ □□□□ □□□□ □□□□□. □□ □□, print (sep='-', end='!')□ □□□ □□□ □□□□ □□ □□□□□. □□□ □□□ □□□ □□□ □□□□ □□□□ □□ □□□ □□□□ □□□□ □ □□□ □ □□□□.
□□ □□□ □□ □□□ □□□□□ □□ □□□ □□□□ □□□□□. □□ □□, print ('Hello', 'World')□ □□ □□□ □□□□ □□ □□□□□. □□ □□□ □□□ □□□□ □□ □□□□□ □□, □□ □□□□□ □□ □ □□□ □□□□ □□□.
□□□□: 1: □□□ □□ □□□□ □□□□ □□□ 5□□ □□ | □□ 2: □□□ - □□ □□ □□ □ □□□□ □□□□□ □□□ □□□□□?

NEW QUESTION: 22

□□ □□□ □□□□□ □□□□□□ □□□□□.

□□ □□□ □ □□ □□ □□□ □□□□□? (□ □□ □□□□ □□□□□.)

A. the_List.index {"1"} in the_list

B. 1.1 in the_list |1:3 |

C. len (□□ [0:2]} <3

D. the_list.index {'1'} -- 0

Answer: C,D (LEAVE A REPLY)

□□

□□□□ □□ □□□ □ □□ □ □□□ "the_list"□□ □□□ □□□□ □□□□□. □□□ □□ □ □□□□.

the_list = ['1', 1, 1, 1]

□ □□□ '1', 1, 1, 1 □□ □□□□ □□ □□□ □□□□ □□ □□ "the_list"□ □□□□□.

□□ □□□□ □□ □□□□ □□□□ □□□ □□□□ □ □□□□. □□□□ □□□□ □□□ □□.

□ □□ □□ 0□□, □□□ □□ □□ □□□□ 1□ □ □□□□□□. □□□□ □□□ □□ □□ □, □ □□ □□ □□□□□ □□□□□. □□ □□, the_list[0]□ '1'□ □□□□, the_list[-1]□ 1□ □□□□□.

□□□□ □□□□ □□□ □□ □□□ □□□□ True □□ False□□ □□ □□ □□□□□ □□ □. □ □ □□□ □□□□□, □□□ □□□□ □□ □□□ □□□□□□.

□□□ □□□ □□□ □□□□ □□ □ □□□□ □□□□□. □□□□ □□□ □□□□.

A). the_List.index {"1"} in the_list: □ □□□□ □□□□ □ '1'□ □□□□ □□□□ □□□ □□□ □□ □□□. □□□ □ □□□□ index □□□□ □□□ □ □□ □□ □□□□ □□□□□ □□ □□ □□□□. index □□□□ □□□□ □ □□□ □□□□ □□ □□□□ □ □□□□□. □□ □□, the_list.index('1')□ '1'□ □□□ □ □□ □□□ □□□ 0□ □□□□□. □□□ the_list.index

{"1"}□ SyntaxError □□□ □□□□□ □□□□ □□□□ □□□□.

B). 1.1 in the_list |1:3 |: □ □□□□ □ 1.1□ □□□ □□ □□□ □□□ □□□□□ □□□.

□□□ □ □□□□ □□ □□□ □□ □ □ □□□□ □□□□ □ □□ □□ □□ □□□ □□□□ □ □□□□ □□□□. □□ □□□ □□□□ □□□ □□□□ □□□ □□□ □□□□ □□□□ □□□ □□ □□□□. □□ □□, the_list[1:3]□ □□□ 1□□ □□□ 3□□(□ □□□ □□)□ □ □ □□□ [1, 1]□ □□□□□. □□□ the_list |1:3 |□ SyntaxError □□□ □□□□□ □□□□ □ □□□ □□□□.

C). len (□□ [0:2]} <3: □ □□□□ □□□ □□ □□□ □□□ 3□□ □□□ □□□□□ □□□.

□ □□□□ len □□□ □□□□ □□□ □□□□ □□□□□ □□□□□. len □□□ □□□□ □□ □□□ □□ □□ □□□ □□□□ □ □□□□□. □□ □□, len(the_list)□ □□□ □□ 4□ □□□□ 4□ □□□□□. □□□□ □□□ □□□□ □□□ □□□□ □□□ □□□ □□□□ □ □□□□□. □□ □□, the_list[0:2]□ □□□ 0□□ □□□ 2□□□ □□□ □□ □□□ ['1', 1]□ □□□□□(□ □□□ □□). □□□ len (the list [0:2]} <3□ □□ □□ ['1', 1]□ □□□ 2□□ 3□□ □□□□ True□ □□□□□.

D). the_list.index {'1'} - 0: □ □□□□ □□□□ □ '1'□ □□□□ 0□ □□□ □□□□□ □□□.
□ □□□□ index □□□□ □□ □□□□ □□□□ □□□□□ □□□□□. index □□□□ □□ □□ □□ □ □□ □□□ □□□□ □ □□□□□. □□ □□, the_list.index('1')□ '1'□ □□□□ □ □□ □□□ □□□ 0□ □□□□□. □□ □□□□ □ □□ □□□□ □□□ True□ □□□□ □□□ □□□ False□ □□□□ □ □□□□□. □□ □□, 0 == 0□ True□ □□□□ 0 == 1□ False□ □□□□□. □□□ the_list.index {'1'} - 0□ □□□□ '1'□ □□□□ 0□□ 0□ 0□ □□ □□□ True□ □□□□□.
□□□ □□□ C. len (□□□ [0:2]} <3 □ D. the_list. index {'1'} - 0□□□.

NEW QUESTION: 23
□□ □□□ □□ □□□ □□□□□?
A. □□□ □□□ □□□□ □□□□.
B. * * *
C. * *
{<□>}: *
Answer: C (LEAVE A REPLY)
□□□□ □□□□ □□ □□□ □□ "counter"□ □□□□ □□□ □□□□ □□□□□□.
0, 42 □□, □□ □ □ □□. □□□ □□□ □□□□.
counter < 0: print("") elif counter >= 42: print("") else: print("") □ □□□ "counter"□ □□ 0□□ □ □□ □□□□ □□□ □□□□□. 0□□ □□□ □□□ □□() □□□ □□□□ □□□□ □□□ □□. 42□□ □□□ □□□ "counter"□ □□ 42□□ □□□ □□□ □□□□□. 42□□ □□□ □□□ □□□ □□() □ □□ □□□□ □□□□ □□□□□. 42□□ □□□ □□□ □□□ □□() □ □□ □□□□ □□□□ □□□□□.
□□□ □□ □□□ "counter" □□ □□ □□□□□. □□□ □□ "counter" □□ 10□□, 10□ 0□ □ □□□ □□ 42□□ □□□ □□□ □□ □□□ □□□ □□ □ □(**)□ □□□□□. □□□ □ □□ C□□□. ** □□: [Python Institute - □□ Python □□□□□ □□□]

NEW QUESTION: 24
□□□ □□ □□□ □□ □□ □□□□□? (□□ □ □□ □□□□□.)
A. □□□ □□ □□□□□□. □, □□□ □□□ □□ □□ □□□ □ □□□□.
B. len { } □□□ □□□ □□□ □ □□□□.
C. □ □□□ { }□ □□□□□.
D. □□□ □□□□□ □□□ □ □□□□□ □□□□□.
Answer: A,D (LEAVE A REPLY)
□□□ □□□□ □□□ □□□ □□ □ □□□, □□□ □□□ □□□□ □ □□□□□. □□□ □ □□, □□, □□□□□ □□ □□ □□□ □□□ □□□□ □ □□ □□□ □□□ □□□□. □□ □ □□□ □□□ □□□□.
* □□□ □□□(immutable)□ □□□ □□□□. □, □□□ □□□ □□ □□ □□□ □ □□□□.
□□□ □□ □□□□ □□, □□ □□ □□□ □ □□□□. □□□ □ □□□ □□□ □□ □□□ □□□□ □□□□ □□□□ □□□□□. □□□ □□ □□□ □□ □□□ □□□□ □□□□ □

□□ □□□□□ □□ □□□□□. □□ □□, □□□ □□□ □□□□□ □□□ □□□ □□ □ □ □□ □□□□ □□□ □□□ □□□□ □□□.
* □□□ □□□ □□□□. □, □□□ □□□□ □□□ □□□ □□□ □□□□ □□□□ □□□ □ □□□□. □□□ □□□□ □ □□ □□□ □□ 0□□ □□□□ □□□ □□□ □□ □□ □□ □□ 1□ □ □□□□□□. □□□□ □□□ □□ □□□, □□□ □□ □□ □□□□□ □□□□ □. □□ □□, □□ t = ("a", "b", "c")□ □□ □ t[0]□ "a"□ □□□□ t□
[-1]□ "c"12□ □□□□□.
* □□□ □□□□□ □□□ □ □□□□□ □□□□□. □, □□□□ □□□□ □□ □ □ □□□ □ □□□□ □□□ □□ □□□□ □□ □□□□ □□□ □ □□□□. □□ □□, □□ t□ □□ □ □
= ("a", "b", "c", "d", "e")□□ t[2]□ "c"□ □□□□ t[1:4]□ ("b", "c", "d")□ □□□□□. □□□□□ □□ □□ □ □□□□ □□□ □□□□□□ □□□ □□□□□ □□□□. □ □□□□ □□□ □ □ □□□ □□ □□□□ □□□□□.12
* □□□ □□□, □□, □□, □□□, □□, □□ □□ □□ □□ □ □□ □□□ □□□ □□□ □ □□□□. □□□ □□ □□ □□ □□ □□□, □□ □□□ □□□ □□□ □ □ □□ □□□ □ □ □□ □□□□□. □□ □□, □ □□ 1□ □ □□ 1□ □□□□ □□ t = (1, 2, 3, 1, 2)□ □□ □ □ □□□.
2□12
* □□□ □□ □□□ □□□□□. □, □□ □□ □□□ □□□ □□□ □□□. □□ □□, □□ □ □□ □□□□ □□ t = ("a", "b", "c")□ □□ □ □□□□. □□□ □□ □□□ □□□□ □□ □□ □ □□□ □□□□ □□□ □□ □□ □□□□. □□ □□, □□□ □□□□ □□ □□ t = "a", "b", "c"□ □□ □ □□□□. □□ □□ □□□□□ □□, □□□ □□□ □□ □□ □□□ □ □□□□.
* len() □□□ □□□□ □□□ □ □□□□. □, len() □□□ □□□□ □□□ □□ □□□ □□ □ □□□□. □□ □□, □□ t = ("a", "b", "c")□ □□ □ len(t)□ 312□ □□□□□.
* □ □□□ ()□ □□□□□. □, □□□ □□ □□□ □□□□ □ □□ □□ □□□□ □□□. □□ □□, □ □□□ □□□□ □ □□ t = ()□ □□ □ □□□□.
□□□ □□□ □□□ □□ □□□ □□□□ □□ □□ □□□ □□□□ □□□. □□□ □□□ Python□ □□□ □□□□ □□□□. □□ □□, □□□ □□□□ □□□ □□□ □□ t = ("a",)□ □□ □ □□□□. □□□ □□□ A□□□. □□□ □□ □□□□□□ □□ □□ □□□ □□□ □ □□□□. D□□□. □□□ □□□□□ □□□□□ □□□□□ □ □□□□.
□□: Python □□ - W3SchoolsPython□ □□ - GeeksforGeeks

NEW QUESTION: 25
□□ □□□ □□ □□□ □□□□□?
A. □□□ □□□□ □□□□ □□□□.
B. 3
C. 1
D. 4
Answer: (SHOW ANSWER)
□□

□□□□ □□ □□□ □ □□□ □□□ □□□□ □□□ □□□□ □□□□□. □□□ □□□ □ □□□.

num1 = 1 num2 = 2 num1 == num2□□: print(4) □□□ □□□: print(1)

□ □□□ □□ "num1"□ "num2"□ □□ □ 1□ 2□ □□□□ □□□ □□□□□. □□ □□, □□ □□□(==)□ □□□□ "num1"□ "num2"□ □□ □□□□ if □□ □□□□□. □□ □□□ 4□ □ □□ □□□□, □□ □□ □□□ 1□ □□□ □□□□□.

□ □□□ □□ □□□ 1□□□. "num1"□ "num2"□ □□ □□ □□ □□□□□. □□□ □□□ C □□□. 1.

NEW QUESTION: 26

□□ □□□ □□□□□ □□□□□□ □□□□□.

□□ □□□ □ □□ □□ □□□ □□□□□? (□ □□ □□□□ □□□□□.)

A. the_List.index {"1"} in the_list

B. 1.1 in the_list |1:3 |

C. len (□□ [0:2]} <3

D. the_list.index {'1'} -- 0

Answer: C,D (LEAVE A REPLY)

□□□□ □□ □□□ □ □□ □ □□□ "the_list"□□ □□□ □□□□ □□□□□. □□□ □□ □ □□□□.

the_list = ['1', 1, 1, 1]

□ □□□ '1', 1, 1, 1 □□ □□□□ □□ □□□ □□□□ □□ □□ "the_list"□ □□□□□.

□□ □□□□ □□ □□□□ □□□□ □□□ □□□□ □ □□□□. □□□□ □□□□ □□□ □□.

□ □□ □□ 0□□, □□□ □□ □□ □□□□ 1□ □ □□□□□□. □□□□ □□□ □□ □□ □, □ □□ □□ □□□□□ □□□□□. □□ □□, the_list[0]□ '1'□ □□□□, the_list[-1]□ 1□ □□□□□.

□□□□ □□□□ □□□ □□ □□□ □□□□ True □□ False□□ □□ □□ □□□□□ □□ □. □ □ □□□ □□□□□, □□□ □□□□ □□ □□□ □□□□□□.

□□□ □□□ □□□ □□□□ □□ □ □□□□ □□□□□. □□□□ □□□ □□□□.

A). the_List.index {"1"} in the_list: □ □□□□ □□□□ □ '1'□ □□□□ □□□□ □□□ □□□ □□ □□□. □□□ □ □□□□ index □□□□ □□□ □ □□ □□ □□□□ □□□□□ □□ □□ □□□□. index □□□□ □□□□ □ □□□ □□□□ □□ □□□□ □ □□□□□. □□ □□, the_list.index('1')□ '1'□ □□□ □ □□ □□□ □□□ 0□ □□□□□. □□□ the_list.index

{"1"}□ SyntaxError □□□ □□□□□ □□□□ □□□□ □□□□.

B). 1.1 in the_list |1:3 |: □ □□□□ □ 1.1□ □□□ □□ □□□ □□□ □□□□□ □□□.

□□□ □ □□□□ □□ □□□ □□ □ □ □□□□ □□□□ □ □□ □□ □□ □□□ □□□□ □ □□□□ □□□□. □□ □□□ □□□□ □□□ □□□□ □□□ □□□ □□□□ □□□□ □□□ □□ □□□□. □□ □□, the_list[1:3]□ □□□ 1□□ □□□ 3□□(□ □□□ □□)□ □

□ □□□ [1, 1]□ □□□□□. □□□ the_list |1:3 |□ SyntaxError □□□ □□□□□ □□□□ □ □□□ □□□□.
C). len (□□ [0:2]} <3: □ □□□□ □□□ □□ □□□ □□□ 3□□ □□□ □□□□□ □□□.
□ □□□□ len □□□ □□□□ □□□ □□□□ □□□□□ □□□□□. len □□□ □□□□ □□ □□□ □□ □□ □□□ □□□□ □ □□□□□. □□ □□, len(the_list)□ □□□ □□ 4□ □□□□ 4□ □□□□□. □□□□ □□□ □□□□ □□□ □□□□ □□□ □□□ □□□□ □ □□□□□. □□ □□, the_list[0:2]□ □□□ 0□□ □□□ 2□□□ □□□ □□ □□□ ['1', 1]□ □□□□□(□ □□□ □□). □□□ len (the list [0:2]} <3□ □□ □□ ['1', 1]□ □□□ 2□□ 3□□ □□□□ True□ □□□□□.
D). the_list.index {'1'} - 0: □ □□□□ □□□□ □ '1'□ □□□□ 0□ □□□ □□□□□ □□□.
□ □□□□ index □□□□ □□ □□□□ □□□□ □□□□□ □□□□□. index □□□□ □□ □□ □□ □ □□ □□□ □□□□ □ □□□□□. □□ □□, the_list.index('1')□ '1'□ □□□□ □ □□ □□□ □□□ 0□ □□□□□. □□ □□□□ □ □□ □□□□ □□□ True□ □□□□ □□□ □□□ False□ □□□□ □ □□□□□. □□ □□, 0 == 0□ True□ □□□□ 0 == 1□ False□ □□□□□. □□□ the_list.index {'1'} - 0□ □□□□ '1'□ □□□□ 0□□ 0□ 0□ □□ □□□ True□ □□□□□.
□□□ □□□ C. len (□□□ [0:2]} <3 □ D. the_list. index {'1'} - 0□□□.
□□: Python □□□ □□□ - W3Schools5. □□ □□ - Python 3.11.5 □□. Python □□□ □□□ - GeeksforGeeks

NEW QUESTION: 27
□□ □□ □ □ □□ □□□ □□□ □ □□ □□□ □□□□□?
A.
B.
C.
{<□>}:
Answer: B (LEAVE A REPLY)
□□□□ □□ □□□ Python□□ □ □□ □□ □□□ □□□□ □□□□. □□□ □□ □□□ □ □□□ □□ □□□□ □□□□□□ □□□□ □ □□□□. □□□ 0□ □□□ □□□ □□ □ □ □□, □□□ □□□ □□□ □ □□□ □□□□ □□□□. □□□ □□ □□□□□ Python□ □□ □□ □□ None□ □□□ □□ □□□□.
□□□□□ □□□ □□□□□ def □□□ □□□ □□ □□□ □□□ □□□□□. □□ □□□ □ □□ □□ □□□□□ □□□ □□□ □ □□□□.
□□ □□ □□□ □□□□ □□ □□□□ □□□ □□ □□□ □□□□□. □:
def function_name(parameter1, parameter2): # □□ □□ □□ □□ Python□□ □□□ □□□□ □ □□ □□ □□ □□□ □□□□□. □□ □□□ □□□ □□□□ □□ □□ □□□ □ □□□ □. □□□ □□□ □□□ □□□ □□□ □□□□ □□□□ □□ □ □□ □□□ □□ □□□□□ □□ □ □□□ □□□□ □□□. □:
□□_□□(□□1, □□2)
□□□□ □□□□ □□ □□□ □□□ □□□□.

A) def my_function(): print("Hello")
B) def my_function(a, b): a + b□ □□□□□.
C) def my_function(a, b, c): a * b * c□ □□□□□.
D) def my_function(a, b=0): a - b□ □□□□□.
□□□ □ □□ □□□ □□□ □ □□ □□□ □□□□ □□ □□□□. □, □□ □□ □ □□□□ □ □ □□□ □□□, □□□□ □□ □□□□ □□□ □□ □□□□ □□□ □□□ □□□.
□□□□ □□ □□ □ □□□ □□□□ □□ □□ □□□□□ □□□□ □□□□. □□ □□, □□ D□□ □□□□ b□ □□□□ 0□□□, □□□ □□ □□ □ □□ □□□ □□□ □ □□□□.
□ □□□ □□□□ □□□ □□□ □□ B□□□. □□ B□ □□□ a□ b□□ □ □□ □□□□□ □□□, □ □□□□□ □□ □□□□□. □ □□□ □ □□ □□□ □□□ □ □□□□. □□ □□, my_function(2, 3)□ 5□ □□□□□.
□□ □□□ □ □□ □□□ □□□ □ □□□□. □□ A□ □□□□□ □□□□ my_function()□ □ □□ □□□ □□□ □ □□□, □ □□ "Hello"□ □□□□□. □□ C□ a, b, c□□ □ □□ □□ □□□ □□□, □ □ □□ □□□□□ □□ □□□□□. □ □□□ my_function(2, 3, 4)□□ □ □ □ □□□□ □□□ □ □□□, □ □□ 24□ □□□□□. □□ D□ □□□□ □□ □□□□ b□ □ □□□ □□ □□ □□□□ a□ □□□, □ □ □□□□□ □□ □□□□□. □ □□□ my_function(2) □□ my_function(2, 3)□□ □□ □□ □ □□ □□□ □□□ □ □□□, □□ 2 □ □ -1□ □□□□□.
□□□ □□□ B□□□. □□ B.