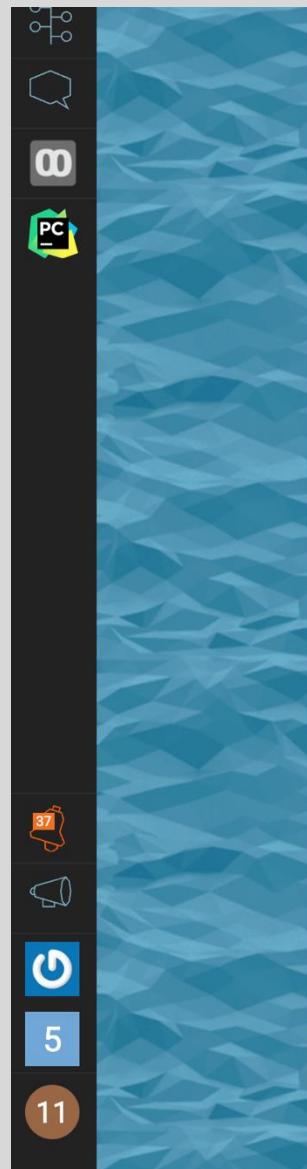


PY.CHECKIO
(LESSONS)





Multiply (intro)

Elementary

English

This is an intro mission, the purpose of which is to explain how to solve missions on CheckiO. If you want to know how to get the maximum out of using CheckiO, check out our blog post "[From Basic to Advance usage](#)".

This mission is the easiest one. Write a function that will receive 2 numbers as input and it should return the multiplication of these 2 numbers.

Input: Two arguments. Both are of type int.

Output: Int.

Examples:

```
1 assert mult_two(3, 2) == 6
2 assert mult_two(0, 1) == 0
```

A series of hints below will help you to understand how to solve the mission. Start the series by clicking on "I don't know how to solve that mission."

Become Awesome

- No Ads
- No Limits
- More Content



Solve

Become Awesome = No ADs + No Limits + More Content;

← Prev Next →

124 100 %

Story Discuss Best Solutions Solution Search Rand. Solution Next mission

Share: [f](#) [t](#)

[https://py.checkio.org/m](#)

Player of the Month

 17 sanddro

Current LeaderBoard

25 days ago	hide
Users attempted:	3534
Users succeeded:	2807
Users attempted (Overall):	403802
Users succeeded (Overall):	369794

Tablero | Presentacion | DeepSeek | Para qué val | (37) Multiply | Tinkering a | engranajes e | polea en ing | 3D design C | +

py.checkio.org/en/mission/multiply-intro/

I don't know how to solve that mission.

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

```
1 def mult_two(a: int, b: int) -> int:
2     mult=a*b
3     return mult
4
5
6 print("Example")
7 print(mult_two(1, 2))
8
9 assert mult_two(3, 2) == 6
10 assert mult_two(0, 1) == 0
11
12 print("The first mission is done! Click 'Check' to earn cool rewards!")
13
```

>>>

Best "Clear" Solution  40 [oduvan](#) [Follow] Operator Nov 17, 2018 All "Clear" Solutions

```
1 from operator import mul as mult_two
```

25 days ago hide
Users attempted: 3534
Users succeeded: 2807
Users attempted (Overall): 403802
Users succeeded (Overall): 260784

 PyCharm
The Python IDE for Professional Developers.
Enjoy productive Python, web and scientific development with PyCharm. Download it now!

37

py.checkio.org/en/mission/first-word-simplified/

The interface shows a sidebar with various icons (file, code, terminal, etc.) and a central content area. The content area has a decorative background of a hand holding a smartphone over a landscape. It contains mission instructions, examples, and a code editor.

You are given a string and you have to find its first word.

- The input string consists of only English letters and spaces.
- There aren't any spaces at the beginning and the end of the string.

greeting from CheckiO Planet

1 2 3 4

greeting

Input: A string (str).

Output: A string (str).

Examples:

```
1 assert first_word("Hello world") == "Hello"
2 assert first_word("a word") == "a"
3 assert first_word("greeting from CheckiO Planet") == "greeting"
4 assert first_word("hi") == "hi"
```

How it is used: The first word is a command in a command line.

Precondition: The text can contain a-z, A-Z and spaces.

Rand. Solution

Next mission

Share: [f](#) [t](#)

[https://py.checkio.org/m](#)

Player of the Month

17 sanddro

Current LeaderBoard

1 month ago	hide
Users attempted:	2164
Users succeeded:	1904
Users attempted (Overall):	176118
Users succeeded (Overall):	131448

You

ClassRoom

Demo is Ready

With the new tool CheckiO ClassRoom, you will be able

py.checkio.org/en/mission/first-word-simplified/

I have no idea how to start solving this mission.

oduvan 40 [Follow]

Your general strategy here should be to split the string into words and then get the first one.

- 0 +

OK. Maybe you can give me some doc links?

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

```
1 def first_word(text: str) -> str:
2     separate=text.split(" ")
3     first=separate[0]
4     return first
5
6
7 print("Example:")
8 print(first_word("Hello world"))
9
10 # These "asserts" are used for self-checking
11 assert first_word("Hello world") == "Hello"
12 assert first_word("a word") == "a"
13 assert first_word("greeting from Checkio Planet") == "greeting"
14 assert first_word("hi") == "hi"
15
```

1 month ago hide
Users attempted: 2164
Users succeeded: 1904
Users attempted (Overall): 176118
Users succeeded (Overall): 131448

You ClassRoom Demo is Ready
With the new tool CheckIO
ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map, and so on.

py.checkio.org/en/mission/is-even/

Some hints are available. Click the button below

I have no idea how to start solving this mission

Even numbers are evenly divisible by 2 (divided by 2 with remainder 0). So to solve this mission you should divide input number by two, find the remainder and compare it to 0.

40 freeman... [Follow]

0

What instruments of Python should I use?

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

```
1 def is_even(num: int) -> bool:
2     if (num % 2)==0:
3         return True
4     else:
5         return False
6
7
8 print("Example:")
9 print(is_even(2))
10
11 # These "asserts" are used for self-checking
12 assert is_even(2) == True
13 assert is_even(5) == False
14 assert is_even(0) == True
15
16 print("The mission is done! Click 'Check Solution' to earn rewards!")
17
```

>>> exec show

11

10

9

8

7

6

5

4

3

2

1

PyCharm

The Python IDE for Professional Developers.

Enjoy productive Python, web and scientific development with PyCharm. Download it now!

1 month ago hide

Users attempted: 2835

Users succeeded: 2642

Users attempted (Overall): 70898

Users succeeded (Overall): 64831

py.checkio.org/en/mission/acceptable-password-i/

Acceptable Password I

Elementary

English



You are at the beginning of a password series. Every mission is based on the previous one. The missions that follow will become slightly more complex.

In this mission, you need to create a password verification function.

The verification condition is:

- the length should be bigger than 6.

Input: A string (str).

Output: A logic value (bool).

Examples:

```
1 assert is_acceptable_password("short") == False
2 assert is_acceptable_password("muchlonger") == True
3 assert is_acceptable_password("ashort") == False
```

Solve

Become Awesome = No Ads + No Limits + More Content;

INITIATION
Prev Next
124 100 %

Story Discuss Best Solutions Solution Search Rand. Solution Next mission

Share: [f](#) [t](#)
[m](#) <https://py.checkio.org/m>

Player of the Month
17 sandro
[Current LeaderBoard](#)

You ClassRoom Demo is Ready

With the new tool CheckIO ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map, and so on.

23 5 11

py.checkio.org/en/mission/acceptable-password-i/

Become AWESOME

- No Ads
- No Limits
- More Content

How to improve this mission? <https://github.com/CheckIO-Missions/checkio-mission-acceptable-password-1>

by:  40 oduvan [Follow] 193

Some hints are available. Click the button below

I have no idea how to start solving this mission.

Run Code (ctrl + /) Check Solution (ctrl + ,)

```
Become Awesome = No ADs + No Limits + More Content;
1 def is_acceptable_password(password: str) -> bool:
2
3     return len(password)> 6
4
5
6 print("Example:")
7 print(is_acceptable_password("short"))
8
9 # These "asserts" are used for self-checking
10 assert is_acceptable_password("short") == False
11 assert is_acceptable_password("muchlonger") == True
12 assert is_acceptable_password("ashort") == False
13
14 print("The mission is done! Click 'Check Solution' to earn rewards!")
15
```

1 month ago hide
Users attempted: 740
Users succeeded: 695
Users attempted (Overall): 122210
Users succeeded (Overall): 99242

You ClassRoom Demo is Ready
With the new tool CheckIO ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map, and so on.



23 5 11

py.checkio.org/en/mission/number-length/

Finish update :

You have a non-negative integer. Try to find out how many digits it has.

378

1 2 3

14521

1 2 3 4 5

Input: A non-negative integer (int).

Output: An integer (int).

Examples:

```
1 assert number_length(10) == 2
2 assert number_length(0) == 1
3 assert number_length(4) == 1
4 assert number_length(44) == 2
```

Solve

Become Awesome = No ADs + No Limits + More Content;

Story

Discuss

Best Solutions

Solution Search

Rand. Solution

Next mission

Share: [f](#) [t](#)

<https://py.checkio.org/m>

Player of the Month

17 sandro

Current LeaderBoard

1 month ago [hide](#)

Users attempted: 1588

Users succeeded: 1532

Users attempted (Overall): 92317

Users succeeded (Overall): 83391

You

ClassRoom

Demo is Ready

With the new tool CheckIO ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map

py.checkio.org/en/mission/number-length/

Some hints are available. Click the button below

I have no idea how to start solving this mission.

oduvan 40 (Follow)

rodri... 5

String in Python is a list of characters or symbols. So the string "Hi" is a list of two elements "H" and "i". So almost all list functions work correctly with string. Your goal is to convert integer value into a string, and then find the length of that string (or find the length of list of chars).

Run Code (ctrl + /) Check Solution (ctrl + ,)

Which function am I looking for?

Become Awesome = No ADs + No Limits + More Content!

```
1 def number_length(value: int) -> int:
2     lol=len(str(value))
3     return lol
4
5
6 print("Example:")
7 print(number_length(10))
8
9 # These "asserts" are used for self-checking
10 assert number_length(10) == 2
11 assert number_length(0) == 1
12 assert number_length(4) == 1
13 assert number_length(44) == 2
14
15 print("The mission is done! Click 'Check' to earn cool rewards!")
16
```

1 month ago

Users attempted: 1588

Users succeeded: 1532

Users attempted (Overall): 92317

Users succeeded (Overall): 83391

You ClassRoom Demo is Ready

With the new tool CheckiO ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map, and so on.

py.checkio.org/en/mission/backward-string/

Backward String

Elementary

English

You should return a given string in reverse order.



Input: A string (str).

Output: A string (str).

Examples:

```
1 assert backward_string("val") == "lav"
2 assert backward_string("") == ""
3 assert backward_string("ohho") == "ohho"
4 assert backward_string("123456789") == "987654321"
```

Solve

Become Awesome = No Ads + No Limits + More Content;

124 100 %

Story Discuss Best Solutions Solution Search Rand. Solution Next mission

Share: [f](#) [t](#)

<https://py.checkio.org/m>

Fresh Awesome

 2	MR_ki
 17	hao.daniel
 6	medea447

[Become Awesome](#)

 PyCharm

The Python IDE for Professional Developers.

Enjoy productive Python, web and scientific development with PyCharm. [Download it now!](#)

23 5 11

py.checkio.org/en/mission/backward-string/

Some hints are available. Click the button below

I have no idea how to start solving this mission...

18 Doppelok [Follow] First thing first - [google "python slicing"](#)

0 +

I can't understand...

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

```
1 def backward_string(val: str) -> str:
2     return val[::-1]
3
4
5 print("Example:")
6 print(backward_string("val"))
7
8 # These "asserts" are used for self-checking
9 assert backward_string("val") == "lav"
10 assert backward_string("") == ""
11 assert backward_string("ohho") == "ohho"
12 assert backward_string("123456789") == "987654321"
13
14 print("The mission is done! Click 'Check Solution' to earn rewards!")
15
```

>>> exec show

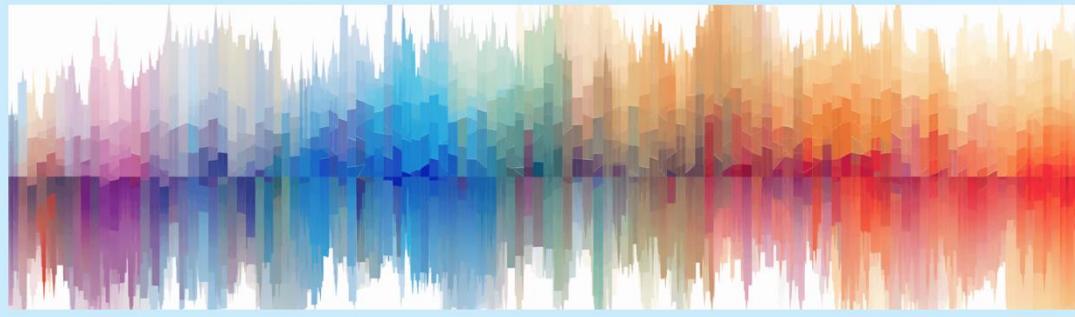
Best "Clear" Solution 31 Olpag [Follow] val[::-1] Feb 27, 2020 All "Clear" Solutions

1 month ago hide
Users attempted: 1921
Users succeeded: 1793
Users attempted (Overall): 89535
Users succeeded (Overall): 84257

PyCharm The Python IDE for Professional Developers. Enjoy productive Python, web and scientific development with PyCharm. Download it now!

py.checkio.org/en/mission/the-most-frequent/

Finish update :



You have a sequence of strings, and you'd like to determine the most frequently occurring string in the sequence. It can be the only one.

`["a", "b", "c", "a", "b", "a"]`

1 1 2 3 \Rightarrow "a"

1 2

1

Input: Non-empty list of string (str).

Output: A string (str).

Examples:

```
1 assert most_frequent(["a", "b", "c", "a", "b", "a"]) == "a"
2 assert most_frequent(["a", "a", "bi", "bi", "bi"]) == "bi"
```

Solve

Become Awesome = No ADs + No Limits + More Content;

Story
Discuss
Best Solutions
Solution Search
Rand. Solution
Next mission

Share: [f](#) [t](#)

<https://py.checkio.org/m>

Player of the Month
17 sandro

Current LeaderBoard

1 month ago [hide](#)
Users attempted: 492
Users succeeded: 419
Users attempted (Overall): 28152
Users succeeded (Overall): 24110

PyCharm
The Python IDE for Professional Developers.

py.checkio.org/en/mission/the-most-frequent/

Finish update

this task.

40 freeman... [Follow]

You need to find an element one of the parameters of which is **maximal** (in this case it's frequency). For these types of cases Python has a great built-in [max\(\)](#) function.

0

How exactly does this function work? Can you give an example?

Run Code (ctrl + /) Check Solution (ctrl + ,)

1 month ago

Users attempted: 492

Users succeeded: 419

Users attempted (Overall): 28152

Users succeeded (Overall): 24110

PyCharm

The Python IDE for Professional Developers.

Enjoy productive Python, web and scientific development with PyCharm. Download it now!

```
from collections import Counter
def most_frequent(data: list[str]) -> str:
    return Counter (data).most_common(1)[0][0]

print("Example:")
print(most_frequent(["a", "b", "c", "a", "b", "a"]))

# These "asserts" are used for self-checking
assert most_frequent(["a", "b", "c", "a", "b", "a"]) == "a"
assert most_frequent(["a", "a", "bi", "bi", "bi"]) == "bi"

print("The mission is done! Click 'Check Solution' to earn rewards!")
```

>>> exec show

py.checkio.org/en/mission/sum-numbers/

In a given text you need to sum the numbers while excluding any digits that form part of a word.

The text consists of numbers, spaces and letters from the English alphabet.

"This is the 1st string with numbers 2 and 11"

Input: A string (str).
Output: An integer (int).

Examples:

```
1 assert sum_numbers("hi") == 0
2 assert sum_numbers("who is 1st here") == 0
3 assert sum_numbers("my numbers is 2") == 2
4 assert (
5     sum_numbers(
6         "This picture is an oil on canvas painting by Danish artist Anna Petersen between 1845 and 1910
7     )
8     == 3755
9 )
```

Solve

Become Awesome = No Ads + No Limits + More Content;

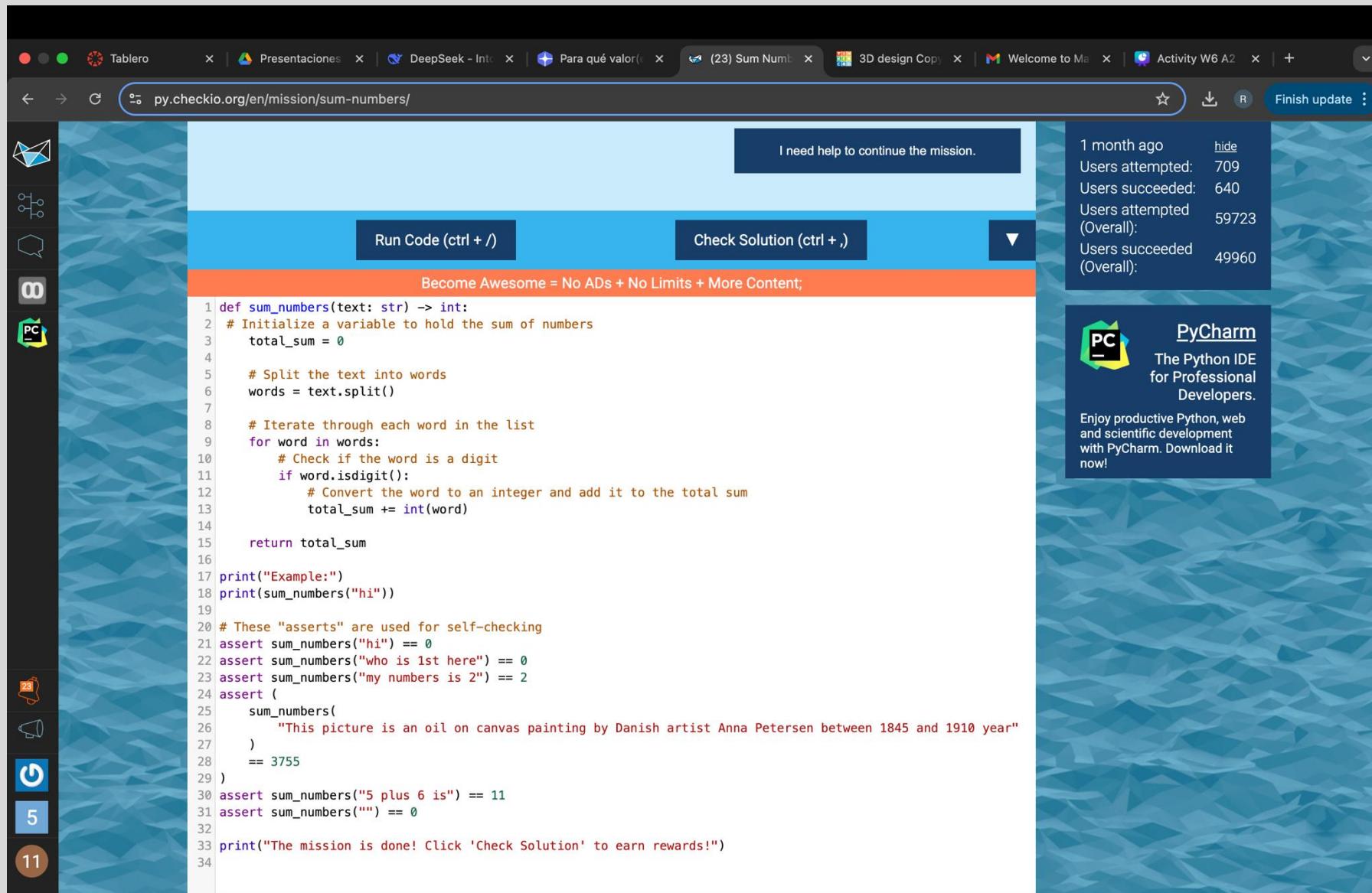
Rand. Solution
Next mission

Share: [f](#) [t](#)
<https://py.checkio.org/m>

Player of the Month
17 sandro

Current LeaderBoard

PyCharm
The Python IDE for Professional Developers.
Enjoy productive Python, web and scientific development with PyCharm.
Download it now!



py.checkio.org/en/mission/sum-numbers/

I need help to continue the mission.

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

```
1 def sum_numbers(text: str) -> int:
2     # Initialize a variable to hold the sum of numbers
3     total_sum = 0
4
5     # Split the text into words
6     words = text.split()
7
8     # Iterate through each word in the list
9     for word in words:
10         # Check if the word is a digit
11         if word.isdigit():
12             # Convert the word to an integer and add it to the total sum
13             total_sum += int(word)
14
15     return total_sum
16
17 print("Example:")
18 print(sum_numbers("hi"))
19
20 # These "asserts" are used for self-checking
21 assert sum_numbers("hi") == 0
22 assert sum_numbers("who is 1st here") == 0
23 assert sum_numbers("my numbers is 2") == 2
24 assert (
25     sum_numbers(
26         "This picture is an oil on canvas painting by Danish artist Anna Petersen between 1845 and 1910 year"
27     )
28     == 3755
29 )
30 assert sum_numbers("5 plus 6 is") == 11
31 assert sum_numbers("") == 0
32
33 print("The mission is done! Click 'Check Solution' to earn rewards!")
34
```

1 month ago [hide](#)
Users attempted: 709
Users succeeded: 640
Users attempted (Overall): 59723
Users succeeded (Overall): 49960

 PyCharm
The Python IDE
for Professional
Developers.
Enjoy productive Python, web
and scientific development
with PyCharm. Download it
now!

py.checkio.org/en/mission/end-zeros/



Story Discuss Best Solutions Solution Search Rand. Solution Next mission

Share: [f](#) [t](#)

<https://py.checkio.org/m>

Fresh Awesome

	2	MR_ki
	17	hao.daniel
	6	medea447

[Become Awesome](#)

PyCharm
The Python IDE for Professional Developers.

Enjoy productive Python, web and scientific development with PyCharm. Download it now!

Try to find out how many zeros a given number has at the end.

134226 \Rightarrow 0

422600 \Rightarrow 2

... 3 2 1

Input: A non-negative integer (int).

Output: An integer (int).

Examples:

```
1 assert end_zeros(0) == 1
2 assert end_zeros(1) == 0
3 assert end_zeros(10) == 1
4 assert end_zeros(101) == 0
```

Solve

Become Awesome = No ADS + No Limits + More Content;

py.checkio.org/en/mission/end-zeros/

I have no idea how to start solving this mission

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

```
1 def end_zeros(a: int) -> int:
2     # Initialize a counter for trailing zeros
3     count = 0
4
5     # Convert the integer to a string to iterate over its characters from the end
6     str_a = str(a)
7
8     # Iterate over the string in reverse
9     for char in reversed(str_a):
10         # If the character is '0', increment the counter
11         if char == '0':
12             count += 1
13         else:
14             # Break the loop if a non-zero character is encountered
15             break
16
17     return count
18
19
20
21 print("Example:")
22 print(end_zeros(10))
23
24 # These "asserts" are used for self-checking
25 assert end_zeros(0) == 1
26 assert end_zeros(1) == 0
27 assert end_zeros(10) == 1
28 assert end_zeros(101) == 0
29 assert end_zeros(245) == 0
30 assert end_zeros(100100) == 2
31
32 print("The mission is done! Click 'Check Solution' to earn rewards!")
33
```

>>> exec show

27 days ago hide
Users attempted: 355
Users succeeded: 317
Users attempted (Overall): 58213
Users succeeded (Overall): 47942

 PyCharm
The Python IDE
for Professional
Developers.
Enjoy productive Python, web
and scientific development
with PyCharm. Download it
now!

py.checkio.org/en/mission/all-the-same/

In this mission you should check if all elements in the given sequence are equal.

`[1, 1, 1, 2]`

$1 == 1 == 1 == 1 == 2 \Rightarrow \text{False}$

Input: List.

Output: Logic value (bool).

Examples:

```
1 assert all_the_same([1, 1, 1]) == True
2 assert all_the_same([1, 2, 1]) == False
3 assert all_the_same([1, "a", 1]) == False
4 assert all_the_same([1, 1, 1, 2]) == False
```

Solve

Become Awesome = No ADS + No Limits + More Content;

Story
Discuss
Best Solutions
Solution Search
Rand. Solution
Next mission

Share: [f](#) [t](#)
<https://py.checkio.org/m>

Fresh Awesome

	2	MR_ki
	17	hao.daniel
	6	medea447

[Become Awesome](#)

PyCharm
The Python IDE for Professional Developers.
Enjoy productive Python, web and scientific development with PyCharm. Download it now!

py.checkio.org/en/mission/all-the-same/

I have no idea how to solve this task.

Run Code (ctrl + /) Check Solution (ctrl + .)

Become Awesome = No ADs + No Limits + More Content;

```
1 from typing import List, Any
2
3
4 def all_the_same(elements: List[Any]) -> bool:
5     if len(elements) <= 1:
6         return True
7
8     # Compare each element with the first element
9     first_element = elements[0]
10    for element in elements:
11        if element != first_element:
12            return False
13
14    return True
15
16
17
18 print("Example:")
19 print(all_the_same([1, 1, 1]))
20
21 # These "asserts" are used for self-checking
22 assert all_the_same([1, 1, 1]) == True
23 assert all_the_same([1, 2, 1]) == False
24 assert all_the_same([1, "a", 1]) == False
25 assert all_the_same([1, 1, 1, 2]) == False
26 assert all_the_same([]) == True
27 assert all_the_same([1]) == True
28
29 print("The mission is done! Click 'Check Solution' to earn rewards!")
30
```

>>> exec show

Best "Clear" Solution 27 vlad.bez... First Feb 02, 2018 All "Clear" Solutions

27 days ago hide
Users attempted: 339
Users succeeded: 305
Users attempted (Overall): 81817
Users succeeded (Overall): 40788

PyCharm
The Python IDE for Professional Developers.
Enjoy productive Python, web and scientific development with PyCharm. Download it now!

11

py.checkio.org/en/mission/easy-unpack/

Your mission here is to create a function that gets a tuple and returns a tuple with only 3 elements - the first, third and second element from the last for the given tuple.

One important thing worth pointing out is that you need to use index in order to extract elements from the tuple. Pay attention, index counting starts from 0, not from 1. Which means that if you need to get the first element from the tuple `elements`, you should do `elements[0]`, and the second element is `elements[1]`.

(6, 2, 9, 4, 3, 9)
1 2 3 4 5 6
-6 -5 -4 -3 -2 -1
(6, 9, 3)

Input: A tuple, at least 3 elements long.

Output: A tuple.

Examples:

```
1 assert easy_unpack((1, 2, 3, 4, 5, 6, 7, 9)) == (1, 3, 7)
2 assert easy_unpack((1, 1, 1, 1)) == (1, 1, 1)
3 assert easy_unpack((6, 3, 7)) == (6, 7, 3)
```

Solve

Become Awesome = No ADS + No Limits + More Content;

Best Solutions
Solution Search
Rand. Solution
Next mission

Share: [f](#) [t](#)
<https://py.checkio.org/m>

Player of the Month
17 sandro

Current LeaderBoard

27 days ago [hide](#)
Users attempted: 272
Users succeeded: 259
Users attempted (Overall): 71288
Users succeeded (Overall): 57991

PyCharm
The Python IDE for Professional Developers.
Enjoy productive Python, web and scientific development with PyCharm. Download it now!

The screenshot shows a web browser window with the URL py.checkio.org/en/mission/easy-unpack/. The page is titled "easy-unpack" and features a banner with the text "No Ads", "No Limits", and "More Content". A user profile for "oduvan" (40 points) is displayed. A sidebar on the left contains various icons and a counter "11". A sidebar on the right displays statistics: 27 days ago, 272 users attempted, 259 succeeded, and overall 71288 attempted and 57991 succeeded. An advertisement for PyCharm is also present.

How to improve this mission? <https://github.com/CheckIO-Missions/checkio-mission-elementary-unpack>

by:  40 [oduvan](#) [Follow](#)

150

Some hints are available. Click the button below

I don't know how should I approach this problem.

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

```
1 def easy_unpack(elements: tuple) -> tuple:
2     # Extract the required elements from the tuple
3     return (elements[0], elements[2], elements[-2])
4
5
6 print("Example:")
7 print(easy_unpack((1, 2, 3, 4, 5, 6, 7, 9)))
8
9 # These "asserts" are used for self-checking
10 assert easy_unpack((1, 2, 3, 4, 5, 6, 7, 9)) == (1, 3, 7)
11 assert easy_unpack((1, 1, 1, 1)) == (1, 1, 1)
12 assert easy_unpack((6, 3, 7)) == (6, 7, 3)
13
14 print("The mission is done! Click 'Check Solution' to earn rewards!")
15
```

exec show

py.checkio.org/en/mission/count-digits/

Count Digits

Elementary

English



You need to count the number of digits in a given string.

Input: String.

Output: Integer.

Examples:

```
1 assert count_digits("hi") == 0
2 assert count_digits("who is 1st here") == 1
3 assert count_digits("my numbers is 2") == 1
4 assert (
5     count_digits(
6         "This picture is an oil on canvas painting by Danish artist Anna Petersen between 1845 and 1910
7     )
8     == 8
9 )
```

Solve

Become Awesome = No ADS + No Limits + More Content;

INITIATION

← Prev Next →

124 100 %

Story
Discuss
Best Solutions
Solution Search
Rand. Solution
Next mission

Share: [f](#) [t](#)

<https://py.checkio.org/m/>

Player of the Month

 17 sandro

[Current LeaderBoard](#)

19 days ago [hide](#)
Users attempted: 132
Users succeeded: 117
Users attempted (Overall): 21773
Users succeeded (Overall): 21078

The screenshot shows a web browser window with the URL py.checkio.org/en/mission/count-digits/. The page displays a mission titled "Count Digits" created by user "oduvan" (40 points). The mission description is: "Some hints are available. Click the button below". A button below the description says "I have no idea how to start solving this mission.". Below the description, there are two buttons: "Run Code (ctrl + /)" and "Check Solution (ctrl +)". A note at the bottom of the code area says "Become Awesome = No ADs + No Limits + More Content;". The code provided is:

```
1 def count_digits(text: str) -> int:
2     # Use a generator expression to count digits in the string
3     return sum(1 for char in text if char.isdigit())
4
5 print("Example:")
6 print(count_digits("hi"))
7
8 # These "asserts" are used for self-checking
9 assert count_digits("hi") == 0
10 assert count_digits("who is 1st here") == 1
11 assert count_digits("my numbers is 2") == 1
12 assert (
13     count_digits(
14         "This picture is an oil on canvas painting by Danish artist Anna Petersen between 1845 and 1910 year"
15     )
16     == 8
17 )
18 assert count_digits("5 plus 6 is") == 2
19 assert count_digits("") == 0
20
21 print("The mission is done! Click 'Check Solution' to earn rewards!")
22
```

At the bottom of the code area, there are "exec" and "show" buttons. The browser's sidebar on the left shows various mission icons with counts: 23, 11, 5, and 40. The top of the browser window shows a tab bar with multiple open tabs, including "Tablero", "Presentaciones", "DeepSeek - Int", "Para qué valor()", "(23) Count Digits", "3D design Copy", "Welcome to Ma", "Activity W6 A2", and a "+" tab.

py.checkio.org/en/mission/all-upper/

All Upper I

Elementary



Check if a given string has all symbols in upper case. If the string is empty or doesn't have any letter in it - function should return **True**.

UPPER and lower \Rightarrow False

✓ ✓ ✓ ✓ ✓ ✗ ✗ ✗ ✗ ✗ ✗

Input: A string (str).

Output: A logic value (bool).

Examples:

```
1 assert is_all_upper("ALL UPPER") == True
2 assert is_all_upper("all lower") == False
3 assert is_all_upper("mixed UPPER and lower") == False
4 assert is_all_upper("") == True
```

Solve

Become Awesome = No Ads + No Limits + More Content;

INITIATION
124 100 %

Story Discuss Best Solutions Solution Search Rand. Solution Next mission

Share: [f](#) [t](#) <https://py.checkio.org/m>

Player of the Month
17 sanddro

Current LeaderBoard

27 days ago [hide](#)
Users attempted: 267
Users succeeded: 256
Users attempted (Overall): 36369
Users succeeded (Overall): 32667

PyCharm

← → ⌂ py.checkio.org/en/mission/remove-all-before/ ⌂ R Finish update :



Story Discuss Best Solutions Solution Search Rand. Solution Next mission

Share: [f](#) [t](#)

https://py.checkio.org/m

Player of the Month

 17 sanddro

[Current LeaderBoard](#)

25 days ago [hide](#)
Users attempted: 202
Users succeeded: 189
Users attempted (Overall): 39708
Users succeeded (Overall): 34108

PyCharm
The Python IDE for Professional Developers

Not all of the elements are important. What you need to do here is to remove from the sequence all of the elements before the given one.

[1, 2, [3, 4, 5]]

For the illustration we have a sequence [1, 2, 3, 4, 5] and we need to remove all elements that go before 3 - which are 1 and 2.

We have two edge cases here: (1) if a cutting element cannot be found, then the sequence shouldn't be changed. (2) if the sequence is empty, then it should remain empty.

Input: List and the border element.

Output: List or another Iterable (tuple, iterator, generator).

Examples:

```
1 assert list(remove_all_before([1, 2, 3, 4, 5], 3)) == [3, 4, 5]
2 assert list(remove_all_before([1, 1, 2, 2, 3, 3], 2)) == [2, 2, 3, 3]
3 assert list(remove_all_before([1, 1, 2, 4, 2, 3, 4], 2)) == [2, 4, 2, 3, 4]
4 assert list(remove_all_before([1, 1, 5, 6, 7], 2)) == [1, 1, 5, 6, 7]
```

Solve

py.checkio.org/en/mission/remove-all-before/

I have no idea how to start solving this mission.

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

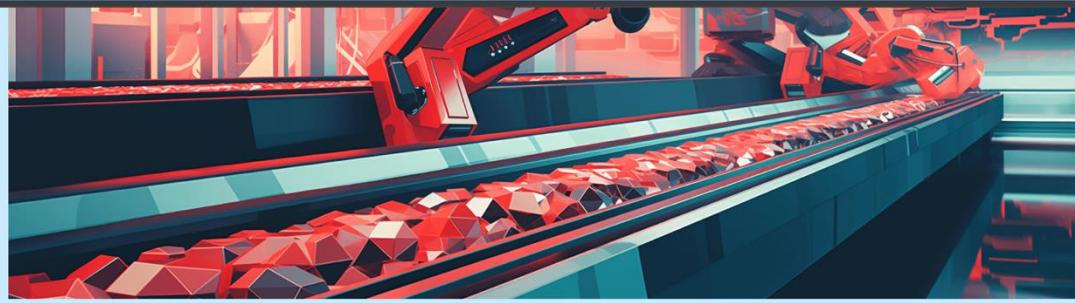
```
1 from collections.abc import Iterable
2
3
4 def remove_all_before(items: list, border: int) -> Iterable:
5     # Find the index of the first occurrence of the border
6     if border in items:
7         index = items.index(border)
8         return items[index:] # Return the sublist from the border to the end
9     return items # Return the entire list if border is not found
10
11
12
13 print("Example:")
14 print(list(remove_all_before([1, 2, 3, 4, 5], 3)))
15
16 # These "asserts" are used for self-checking
17 assert list(remove_all_before([1, 2, 3, 4, 5], 3)) == [3, 4, 5]
18 assert list(remove_all_before([1, 1, 2, 2, 3, 3], 2)) == [2, 2, 3, 3]
19 assert list(remove_all_before([1, 1, 2, 4, 2, 3, 4], 2)) == [2, 4, 2, 3, 4]
20 assert list(remove_all_before([1, 1, 5, 6, 7], 2)) == [1, 1, 5, 6, 7]
21 assert list(remove_all_before([], 0)) == []
22 assert list(remove_all_before([7, 7, 7, 7, 7, 7, 7, 7, 7], 7)) == [
23     7,
24     7,
25     7,
26     7,
27     7,
28     7,
29     7,
30     7,
31     7,
32 ]
33
34 print("The mission is done! Click 'Check Solution' to earn rewards!")
35
```

25 days ago hide
Users attempted: 202
Users succeeded: 189
Users attempted (Overall): 39708
Users succeeded (Overall): 34108

 PyCharm
The Python IDE
for Professional
Developers.
Enjoy productive Python, web
and scientific development
with PyCharm. Download it
now!

py.checkio.org/en/mission/replace-first/

Finish update :



In a given sequence the first element should become the last one. An empty sequence or with only one element should stay the same.

Input: List.

Output: List or another Iterable (tuple, iterator, generator).

Examples:

```
1 assert replace_first([1, 2, 3, 4]) == [2, 3, 4, 1]
2 assert replace_first([1]) == [1]
3 assert replace_first([]) == []
```

Solve

Become Awesome = No Ads + No Limits + More Content;

Story
Discuss
Best Solutions
Solution Search
Rand. Solution
Next mission

Share: [f](#) [t](#)
<https://py.checkio.org/m>

Player of the Month
 17 sanddro
[Current LeaderBoard](#)

25 days ago [hide](#)
Users attempted: 225
Users succeeded: 216
Users attempted (Overall): 47305
Users succeeded (Overall): 44823

You
ClassRoom
Demo is
Ready

With the new tool CheckIO
ClassRoom, you will be able
to analyze your students'
progress, have your own
leaderboard, share

The screenshot shows a mission page on the CheckiO website. The URL in the address bar is py.checkio.org/en/mission/replace-first/. The mission is titled "Replace First" and is categorized under "list" and "numbers". It was created by a user named "oduvan" with a rating of 40. The page indicates that 70 users have attempted the mission. A sidebar on the right provides statistics: 25 days ago, 225 users attempted and 216 succeeded; Overall, 47305 users attempted and 44823 succeeded. A message in the sidebar says "You ClassRoom Demo is Ready". The main content area shows the mission description and the code to solve it. The code uses the `collections.abc` module to create an iterator that replaces the first element of a list with the last one. It includes a self-checking mechanism with assertions. The code editor has "exec" and "show" buttons at the bottom. On the left, there is a sidebar with various icons and numbers (23, 11, 5) and a navigation bar with tabs like "Tablero", "Presentaciones", "DeepSeek - Int", "Para qué valor()", "(23) Replace Fi", "3D design Copy", "Welcome to Ma", "Activity W6 A2", and a "+" button.

How to improve this mission? <https://github.com/CheckiO-Missions/checkio-mission-replace-first>

list numbers by:  40 [oduvan](#) [Follow](#) - 70 +

Some hints are available. Click the button below

I have no idea how to start solving this mission.

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

```
1 from collections.abc import Iterable
2 def replace_first(items: list) -> Iterable:
3     # Check if the list has more than one element
4     if len(items) > 1:
5         # Move the first element to the end of the list
6         first_item = items.pop(0) # Remove the first item
7         items.append(first_item) # Append it to the end
8     return items
9
10
11 # These "asserts" are used for self-checking
12 print("Example:")
13 print(list(replace_first([1, 2, 3, 4])))
14
15 assert replace_first([1, 2, 3, 4]) == [2, 3, 4, 1]
16 assert replace_first([1]) == [1]
17 assert replace_first([]) == []
18
19 print("The mission is done! Click 'Check Solution' to earn rewards!")
20
```

>>> exec show

25 days ago hide
Users attempted: 225
Users succeeded: 216
Users attempted (Overall): 47305
Users succeeded (Overall): 44823

You ClassRoom Demo is Ready
With the new tool CheckiO ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map, and so on.

py.checkio.org/en/mission/max-digit/

Finish update :

The page features a decorative header with a mountain and sailboat illustration. The main content area contains the following text and visual elements:

You have a number and you need to determine which digit in this number is the biggest.

maximum

134226

Input: A positive integer (int).

Output: An integer 0-9 (int).

Examples:

```
1 assert max_digit(0) == 0
2 assert max_digit(52) == 5
3 assert max_digit(634) == 6
4 assert max_digit(1) == 1
```

Solve

Become Awesome = No Ads + No Limits + More Content;

Best Solutions
Solution Search
Rand. Solution
Next mission

Share: [f](#) [t](#)
<https://py.checkio.org/m>

Fresh Awesome

	2	MR_ki
	17	hao.daniel
	6	medea447

[Become Awesome](#)

You
ClassRoom
Demo is
Ready

With the new tool CheckIO ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map, and so on.

py.checkio.org/en/mission/max-digit/

• NO ADS
• No Limits
• More Content

How to improve this mission? <https://github.com/CheckIO-Missions/checkio-mission-max-digit>

numbers string by:  40 oduvan [Follow] - 103 +

Some hints are available. Click the button below

I have no idea how to start solving this mission.

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

```
1 def max_digit(value: int) -> int:
2     return max(int(digit) for digit in str(value))
3
4
5 print("Example:")
6 print(max_digit(10))
7
8 # These "asserts" are used for self-checking
9 assert max_digit(0) == 0
10 assert max_digit(52) == 5
11 assert max_digit(634) == 6
12 assert max_digit(1) == 1
13 assert max_digit(1000) == 1
14
15 print("The mission is done! Click 'Check Solution' to earn rewards!")
16
```

>>> exec show

25 days ago hide
Users attempted: 483
Users succeeded: 466
Users attempted (Overall): 40944
Users succeeded (Overall): 38438

You ClassRoom Demo is Ready
With the new tool CheckIO ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map, and so on.

py.checkio.org/en/mission/beginning-zeros/

The image shows a screenshot of a web browser displaying a Checkio mission titled 'beginning-zeros'. The mission is set in a stylized autumn landscape with orange and yellow trees and large orange numbers (0, 1, 2, 3) scattered across the ground. A large orange circle is in the background. On the left, there's a sidebar with various icons and a vertical list of numbers: 23, 5, 11. The main content area contains text and examples for the mission. On the right, there's a sidebar with links for 'Story', 'Discuss', 'Best Solutions', 'Solution Search', 'Rand. Solution', and 'Next mission'. It also includes social sharing buttons for Facebook and Twitter, a 'Player of the Month' section for 'sandro' (ID 17), and a statistics box showing user activity and success rates. A 'PyCharm' advertisement is at the bottom right.

You have a string that consist only of digits. You need to find how many zero digits ("0") are at the beginning of the given string.

Example:

“134226” \Rightarrow 0

“004226” \Rightarrow 2

1 2 3 ... num of 0

Input: A string (str), that consists of digits.

Output: An integer (int).

Examples:

```
1 assert beginning_zeros("100") == 0
2 assert beginning_zeros("001") == 2
3 assert beginning_zeros("100100") == 0
4 assert beginning_zeros("001001") == 2
```

Solve

Become Awesome = No ADS + No Limits + More Content;

Story
Discuss
Best Solutions
Solution Search
Rand. Solution
Next mission

Share: [Facebook](#) [Twitter](#)
<https://py.checkio.org/m/11/>

Player of the Month
17 sandro

Current LeaderBoard

19 days ago [hide](#)
Users attempted: 393
Users succeeded: 334
Users attempted (Overall): 48907
Users succeeded (Overall): 45507

PyCharm
The Python IDE for Professional Developers.
Enjoy productive Python, web and scientific

py.checkio.org/en/mission/beginning-zeros/

• No Limits
• More Content

How to improve this mission? <https://github.com/CheckIO-Missions/checkio-mission-beginning-zeros>

math string by:  40 oduvan [Follow] - 72 +

Some hints are available. Click the button below

I have no idea how to start solving this mission!

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

```
1 def beginning_zeros(a: str) -> int:
2     # Count the number of leading zeros using the lstrip method
3     return len(a) - len(a.lstrip('0'))
4
5 print("Example:")
6 print(beginning_zeros("10"))
7
8 # These "asserts" are used for self-checking
9 assert beginning_zeros("100") == 0
10 assert beginning_zeros("001") == 2
11 assert beginning_zeros("100100") == 0
12 assert beginning_zeros("001001") == 2
13 assert beginning_zeros("012345679") == 1
14 assert beginning_zeros("0000") == 4
15
16 print("The mission is done! Click 'Check Solution' to earn rewards!")
17
```

>>> exec show

19 days ago hide
Users attempted: 393
Users succeeded: 334
Users attempted (Overall): 48907
Users succeeded (Overall): 45507

 PyCharm
The Python IDE for Professional Developers.
Enjoy productive Python, web and scientific development with PyCharm. Download it now!

py.checkio.org/en/mission/between-markers-simplified/

Finish update :

You are given a string and two markers (the initial one and final). You have to find a substring enclosed between these two markers. But there are a few important conditions.

- The initial and final markers are always different.
- The initial and final markers are always 1 char size.
- The initial and final markers always exist in a string and go one after another.

What is >apple< > <

apple

Input: Three arguments. All of them are strings (str). The second and third arguments are the initial and final markers.

Output: A string (str).

Examples:

```
1 assert between_markers("What is >apple<", ">", "<") == "apple"
2 assert between_markers("What is [apple]", "[", "]") == "apple"
3 assert between_markers("What is ><", ">", "<") == ""
4 assert between_markers("[an apple]", "[", "]") == "an apple"
```

Solve

Become Awesome = No Ads + No Limits + More Content;

Best Solutions
Solution Search
Rand. Solution
Next mission

Share: [f](#) [t](#)
<https://py.checkio.org/m>

Player of the Month
17 sanddro

Current LeaderBoard

19 days ago [hide](#)
Users attempted: 464
Users succeeded: 442
Users attempted (Overall): 39109
Users succeeded (Overall): 36906

You
ClassRoom
Demo is Ready

With the new tool CheckIO ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map, and so on.

py.checkio.org/en/mission/between-markers-simplified/

How to improve this mission: <https://github.com/CheckiO-Missions/CheckiO-Mission-Between-Markers-Simplified>

by:  40 [oduvan](#) [\[Follow\]](#)

19 days ago [hide](#)
Users attempted: 464
Users succeeded: 442
Users attempted (Overall): 39109
Users succeeded (Overall): 36906

Some hints are available. Click the button below

I have no idea how to start solving this mission...

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

```
1 def between_markers(text: str, start: str, end: str) -> str:
2     # Find the positions of the start and end markers
3     start_index = text.find(start) + len(start) if start in text else 0
4     end_index = text.find(end) if end in text else len(text)
5
6     # If the end marker comes before the start marker, return an empty string
7     if end_index < start_index:
8         return ""
9
10    # Return the substring between the markers
11    return text[start_index:end_index]
12
13 print("Example:")
14 print(between_markers("What is >apple<", ">", "<"))
15
```

You ClassRoom Demo is Ready
With the new tool CheckiO
ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map, and so on.

py.checkio.org/en/mission/split-pairs/

Split Pairs

Elementary



Split the string into pairs of two characters. If the string contains an odd number of characters, then the missing second character of the final pair should be replaced with an underscore ('_').

Input: A string.

Output: A list or another Iterable of strings.

Example:

```
1 assert list(split_pairs("abcd")) == ["ab", "cd"]
2 assert list(split_pairs("abc")) == ["ab", "c_"]
3 assert list(split_pairs("abcdf")) == ["ab", "cd", "f_"]
4 assert list(split_pairs("a")) == ["a_"]
```

Precondition: $0 \leq \text{len}(\text{text}) \leq 100$

Solve

Become Awesome = No ADS + No Limits + More Content;

INITIATION
← Prev Next →
124 100 %

Story
Discuss
Best Solutions
Solution Search
Rand. Solution
Next mission

Share: [f](#) [t](#)
<https://py.checkio.org/m>

Player of the Month
17 sandro
Current LeaderBoard

You
ClassRoom
Demo is
Ready

With the new tool CheckIO ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map, and so on.

py.checkio.org/en/mission/split-pairs/

How to improve this mission? <https://github.com/CheckIO-Missions/checkio-mission-split-pairs>

list by:  40 oduvan - 124 +

Some hints are available. Click the button below

I have no idea how to start solving this mission.

Run Code (ctrl + /) Check Solution (ctrl +)

Become Awesome = No ADs + No Limits + More Content;

```
1 from typing import Iterable
2
3 def split_pairs(text: str) -> Iterable[str]:
4     # If the string length is odd, pad it with an underscore
5     if len(text) % 2 != 0:
6         text += "_"
7
8     # Use a list comprehension to create pairs of two characters
9     return [text[i:i+2] for i in range(0, len(text), 2)]
10
11 print("Example:")
12 print(list(split_pairs("abcd")))
13
14 # These "asserts" are used for self-checking
15 assert list(split_pairs("abcd")) == ["ab", "cd"]
16 assert list(split_pairs("abc")) == ["ab", "c_"]
17 assert list(split_pairs("abcdef")) == ["ab", "cd", "f_"]
18 assert list(split_pairs("a")) == ["a_"]
19 assert list(split_pairs("")) == []
20
21 print("The mission is done! Click 'Check Solution' to earn rewards!")
```

>>> exec show

19 days ago hide
Users attempted: 168
Users succeeded: 152
Users attempted (Overall): 28302
Users succeeded (Overall): 25381

You ClassRoom Demo is Ready
With the new tool CheckIO ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map, and so on.



23 5 11

py.checkio.org/en/mission/correct-sentence/

Finish update :



For the input of your function, you will be given one sentence. You have to return a corrected version, that starts with a capital letter and ends with a period (dot).

Pay attention to the fact that not all of the fixes are necessary. If a sentence already ends with a period (dot), then adding another one will be a mistake.

greetings, friends

Greetings, friends.

Input: A string (str).

Output: A string (str).

Examples:

```
1 assert correct_sentence("greetings, friends") == "Greetings, friends."
2 assert correct_sentence("Greetings, friends") == "Greetings, friends."
3 assert correct_sentence("Greetings, friends.") == "Greetings, friends."
4 assert correct_sentence("greetings, friends.") == "Greetings, friends."
```

Solve

Become Awesome = No Ads + No Limits + More Content;

Best Solutions
Solution Search
Rand. Solution
Next mission

Share: [f](#) [t](#)
<https://py.checkio.org/m>

Player of the Month
17 sandro
[Current LeaderBoard](#)

You
ClassRoom
Demo is
Ready

With the new tool CheckIO
ClassRoom, you will be able
to analyze your students'
progress, have your own
leaderboard, change
missions order on the map,
and so on.

py.checkio.org/en/mission/correct-sentence/

• No Ads
• No Limits
• More Content

How to improve this mission? <https://github.com/CheckIO-Missions/checkio-mission-correct-sentence>

by:  40 oduvan [Follow] 157

Some hints are available. Click the button below

Where should I start solving this mission?

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

```
1 def correct_sentence(text: str) -> str:
2     # Capitalize the first letter and ensure the sentence ends with a period
3     return text[0].upper() + text[1:] + ("." if text.endswith(".") else ".")
4
5 print("Example:")
6 print(correct_sentence("greetings, friends"))
7
8 # These "asserts" are used for self-checking
9 assert correct_sentence("greetings, friends") == "Greetings, friends."
10 assert correct_sentence("Greetings, friends") == "Greetings, friends."
11 assert correct_sentence("Greetings, friends.") == "Greetings, friends."
12 assert correct_sentence("greetings, friends.") == "Greetings, friends."
13
14 print("The mission is done! Click 'Check Solution' to earn rewards!")
```

>>> exec show

19 days ago hide
Users attempted: 196
Users succeeded: 180
Users attempted (Overall): 57965
Users succeeded (Overall): 49311

You ClassRoom Demo is Ready
With the new tool CheckIO ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map, and so on.



Nearest Value

Elementary+

English

Find the nearest value to the given one.

You are given a set of integers and a value for which you need to find the nearest one.

For example, we have the following sequence of numbers: 4, 7, 10, 11, 12, 17, and we need to find the nearest value to the number 9. If we sort this sequence in the ascending order, then to the left of number 9 will be number 7 and to the right - will be number 10. But 10 is closer than 7, which means that the correct answer is 10.

A few clarifications:

- If 2 numbers are at the same distance, you need to choose the smallest one;
- The sequence of numbers is always non-empty;
- The given value can be in this sequence, which means that it's the answer;
- The sequence may contain both positive and negative numbers, but they are always integers;
- The sequence isn't sorted and consists only unique numbers.

Input: Two arguments. A set of integers. The sought value as an integer.

Output: An integer.

Examples:

```
1 assert nearest_value({17, 4, 7, 10, 11, 12}, 9) == 10
2 assert nearest_value({17, 4, 7, 10, 11, 12}, 8) == 7
3 assert nearest_value({17, 4, 8, 10, 11, 12}, 9) == 8
4 assert nearest_value({17, 4, 9, 10, 11, 12}, 9) == 9
```

INITIATION

← Prev Next →

124 100 %

Story Discuss Best Solutions Solution Search Rand. Solution Next mission

Share: [f](#) [t](#)

[https://py.checkio.org/m](#)

Fresh Awesome

	2	MR_ki
	17	hao.daniel
	6	medea447

[Become Awesome](#)

19 days ago [hide](#)

The screenshot shows a web browser window with a mission page from py.checkio.org/en/mission/nearest-value/. The page has a purple header with the title 'Para qué valor()' and a sub-header '(23) Nearest Value'. The main content area features a profile for 'oduvan' (40 points) and a message: 'Some hints are available. Click the button below'. Below this is a code editor with Python code for finding the nearest value in a set. The sidebar on the left shows navigation icons and a sidebar on the right displays mission statistics and a 'ClassRoom' demo message.

How to improve this mission? <https://github.com/CheckiO-Missions/checkio-mission-nearest-value>

by:  40 [oduvan](#) [\[Follow\]](#) 68

Some hints are available. Click the button below

I have no idea how to start solving this mission!

Run Code (ctrl + /) Check Solution (ctrl + ,)

Become Awesome = No ADs + No Limits + More Content;

```
1 def nearest_value(values: set[int], one: int) -> int:
2     # Sort the values and find the nearest one based on absolute difference
3     return min(values, key=lambda x: (abs(x - one), x))
4
5 print("Example:")
6 print(nearest_value({4, 7, 10, 11, 12, 17}, 9))
7
8 # These "asserts" are used for self-checking
9 assert nearest_value({17, 4, 7, 10, 11, 12}, 9) == 10
10 assert nearest_value({17, 4, 7, 10, 11, 12}, 8) == 7
11 assert nearest_value({17, 4, 8, 10, 11, 12}, 9) == 8
12 assert nearest_value({17, 4, 9, 10, 11, 12}, 9) == 9
13 assert nearest_value({17, 4, 7, 10, 11, 12}, 0) == 4
14 assert nearest_value({17, 4, 7, 10, 11, 12}, 100) == 17
15 assert nearest_value({100, 5, 8, 89, 10, 12}, 7) == 8
16 assert nearest_value({2, 3, -1}, 0) == -1
17 assert nearest_value({5}, 5) == 5
18 assert nearest_value({5}, 7) == 5
19
20 print("The mission is done! Click 'Check Solution' to earn rewards!")
21
```

>>> exec show

19 days ago [hide](#)
Users attempted: 103
Users succeeded: 83
Users attempted (Overall): 26116
Users succeeded (Overall): 22315

You ClassRoom Demo is Ready
With the new tool CheckiO ClassRoom, you will be able to analyze your students' progress, have your own leaderboard, change missions order on the map, and so on.