Question 1

Create a function that takes a number as an argument and returns True or False depending on whether the number is symmetrical or not. A number is symmetrical when it is the same as its reverse.

**Examples**

is\_symmetrical(7227) ➞ True

is\_symmetrical(12567) ➞ False

is\_symmetrical(44444444) ➞ True

is\_symmetrical(9939) ➞ False

is\_symmetrical(1112111) ➞ True

Question 2

Given a string of numbers separated by a comma and space, return the product of the numbers.

### Examples

multiply\_nums("2, 3") ➞ 6

multiply\_nums("1, 2, 3, 4") ➞ 24

multiply\_nums("54, 75, 453, 0") ➞ 0

multiply\_nums("10, -2") ➞ -20

Question 3

Create a function that squares every digit of a number.

### Examples

square\_digits(9119) ➞ 811181

square\_digits(2483) ➞ 416649

square\_digits(3212) ➞ 9414

### Notes

The function receives an integer and must return an integer.

Question 4

Create a function that sorts a list and removes all duplicate items from it.

### Examples

setify([1, 3, 3, 5, 5]) ➞ [1, 3, 5]

setify([4, 4, 4, 4]) ➞ [4]

setify([5, 7, 8, 9, 10, 15]) ➞ [5, 7, 8, 9, 10, 15]

setify([3, 3, 3, 2, 1]) ➞ [1, 2, 3]

Question 5

Create a function that returns the mean of all digits.

### Examples

mean(42) ➞ 3

mean(12345) ➞ 3

mean(666) ➞ 6

### Notes

* The mean of all digits is the sum of digits / how many digits there are (e.g. mean of digits in 512 is (5+1+2)/3(number of digits) = 8/3=2).
* The mean will always be an integer.

Solutions:

Solution 1:

def is\_symmetrical(num):

return str(num) == str(num)[::-1]

print(is\_symmetrical(7227)) # True

print(is\_symmetrical(12567)) # False

print(is\_symmetrical(44444444)) # True

print(is\_symmetrical(9939)) # False

print(is\_symmetrical(1112111)) # True

Solution 2:

def multiply\_nums(nums):

num\_list = [int(x) for x in nums.split(", ")]

result = 1

for num in num\_list:

result \*= num

return result

print(multiply\_nums("2, 3")) # 6

print(multiply\_nums("1, 2, 3, 4")) # 24

print(multiply\_nums("54, 75, 453, 0")) # 0

print(multiply\_nums("10, -2")) # -20

Solution 3:

def square\_digits(num):

return int(''.join(str(int(digit)\*\*2) for digit in str(num)))

print(square\_digits(9119)) # 811181

print(square\_digits(2483)) # 416649

print(square\_digits(3212)) # 9414

Solution 4:

def setify(lst):

return list(set(lst))

print(setify([1, 3, 3, 5, 5])) # [1, 3, 5]

print(setify([4, 4, 4, 4])) # [4]

print(setify([5, 7, 8, 9, 10, 15])) # [5, 7, 8, 9, 10, 15]

print(setify([3, 3, 3, 2, 1])) # [1, 2, 3]

Solution 5:

def mean(num):

digits = [int(digit) for digit in str(num)]

return sum(digits) // len(digits)

print(mean(42)) # 3

print(mean(12345)) # 3

print(mean(666)) # 6