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

# 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.

# Define a function, function1

# that takes a number as an argument and returns True or False

def function1(num):

s1 = str(num)

s2 = s1[::-1]

rev\_num = int(s2)

if num == rev\_num:

return True

else:

return False

# Call the function, function1 with given inputs and check the results

print(function1(7227)) # It should return True

print(function1(12567)) # It should return False

print(function1(44444444)) # It should return True

print(function1(9939)) # It should return False

print(function1(1112111)) # It should return 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

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

# Declare a function, function1 with takes string as input

# and return the product of numbers

def function1(s1):

l1 = s1.split(", ")

product = 1

for i in l1:

product = product \* int(i)

return product

# Check the function, function1 with different inputs given and check the result

# ("2, 3") ➞ 6

print(function1("2, 3")) # The output should be 6

# ("1, 2, 3, 4") ➞ 24

print(function1("1, 2, 3, 4")) # The output should be 24

# ("54, 75, 453, 0") ➞ 0

print(function1("54, 75, 453, 0")) # The output should be 0

# ("10, -2") ➞ -20

print(function1("10, -2")) # The output should be -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.

# Create a function that squares every digit of a number.

# Define a function, function1 which takes a number as argument

# that squares every digit of a number

def function1(num):

s1 = str(num)

s2 = ""

for i in range(0,len(s1)):

j = int(s1[i])\*\*2

s2 = s2+str(j)

return int(s2)

# Check function, function1 with multiple inputs given

# and check the result

print(function1(9119)) # The output should be 811181

print(function1(2483)) # The output should be 416649

print(function1(3212)) # The output should be 9414

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]

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

# Define a function, function1 with takes a list as input

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

def function1(l1):

# covert list into set to sort and remove duplicates.

# convert set back to list

return list(set(l1))

# Check with multiple inputs and verify the result

l2 = [1, 3, 3, 5, 5]

print(function1(l2)) # it should give output [1, 3, 5]

l2 = [4, 4, 4, 4]

print(function1(l2)) # it should give output [4]

l2 = [5, 7, 8, 9, 10, 15]

print(function1(l2)) # it should give output [5, 7, 8, 9, 10, 15]

l2 = [3, 3, 3, 2, 1]

print(function1(l2)) # it should give output [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.

# Create a function that returns the mean of all digits.

# Define a function, function1 that returns the mean of all digits.

def function1(num):

s1 = str(num)

sum = 0

for i in range(len(s1)):

sum = sum + int(s1[i])

return sum / len(s1) # return the mean of all digits

# Check the function, function1 with different inputs and check the results

print(function1(42)) # The output should be 3

print(function1(12345)) # The output should be 3

print(function1(666)) # The output should be 6