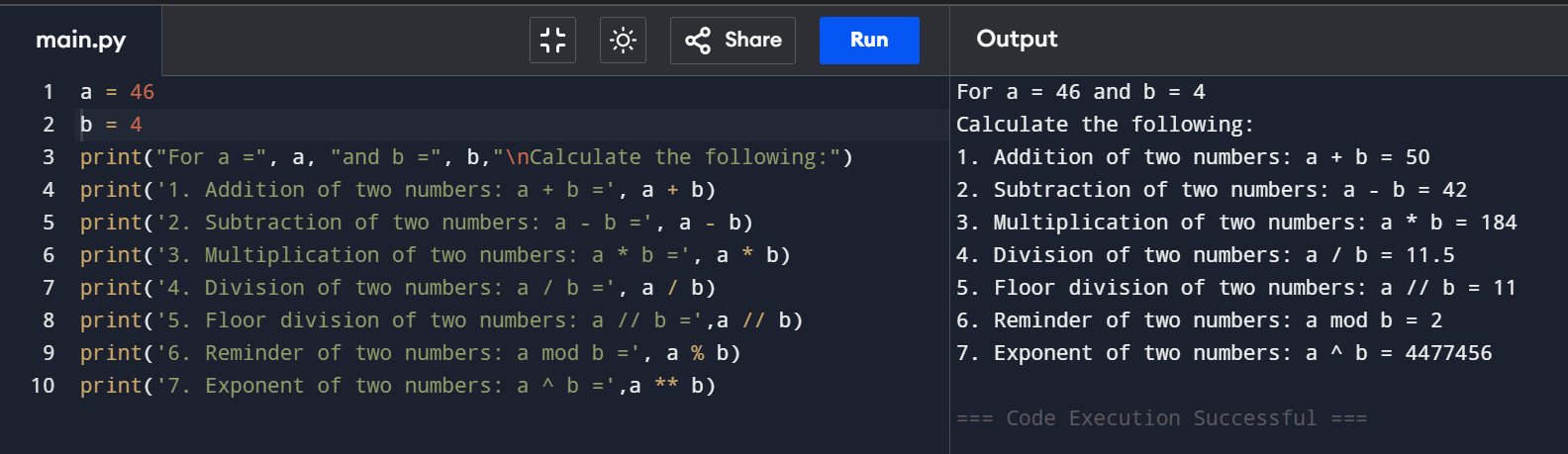
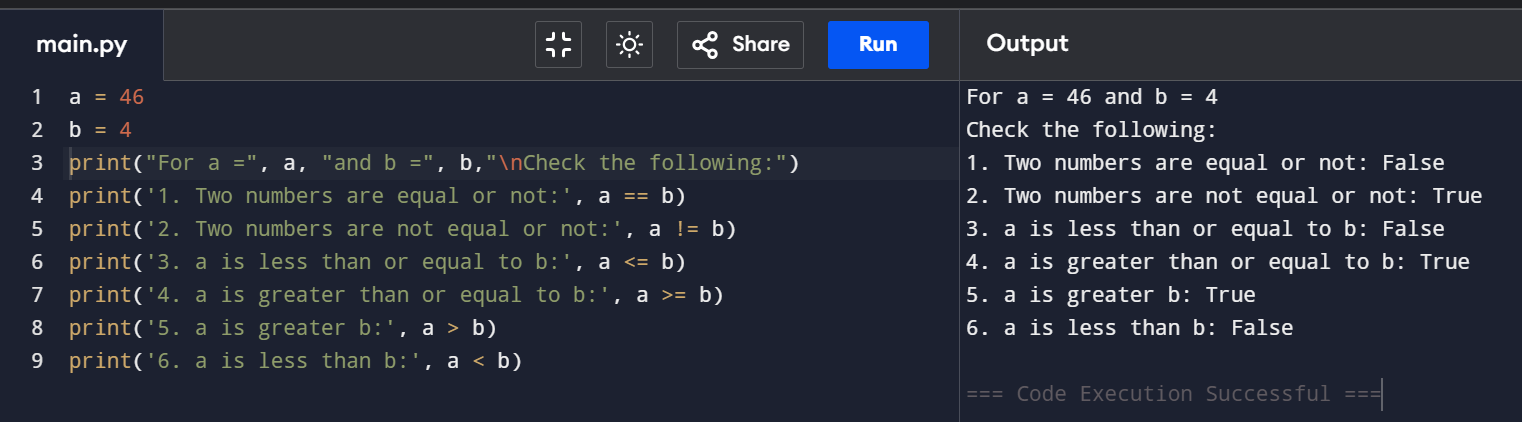
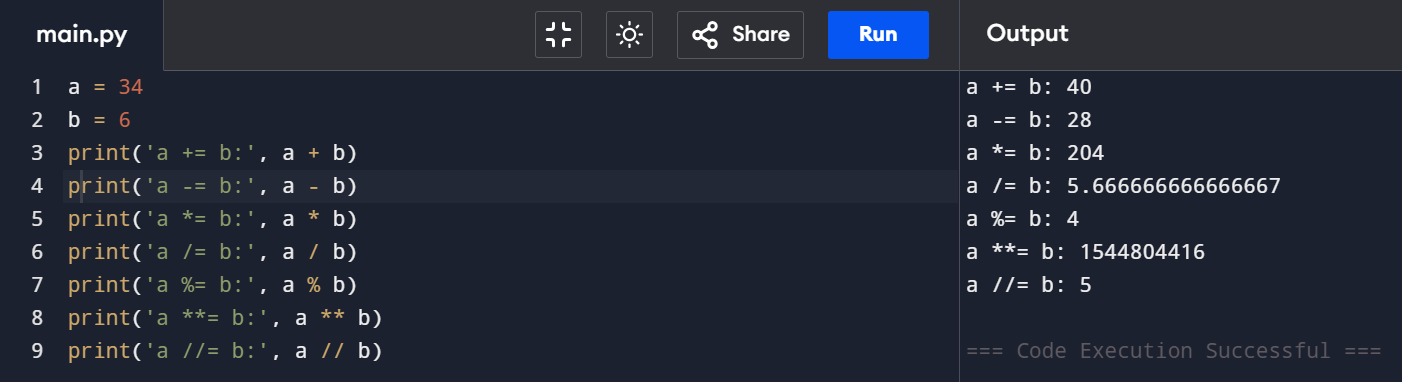
Arithmetic operations between 2 numbers



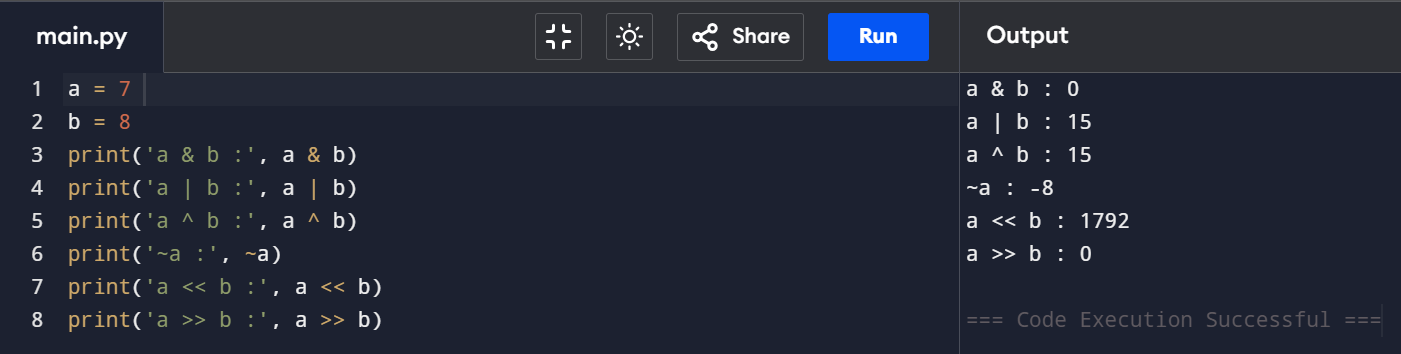
Comparison operations between 2 numbers



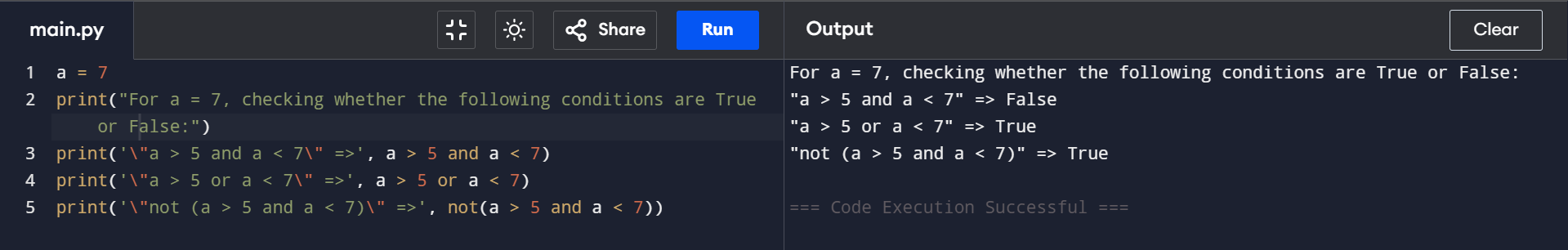
Assignment operators



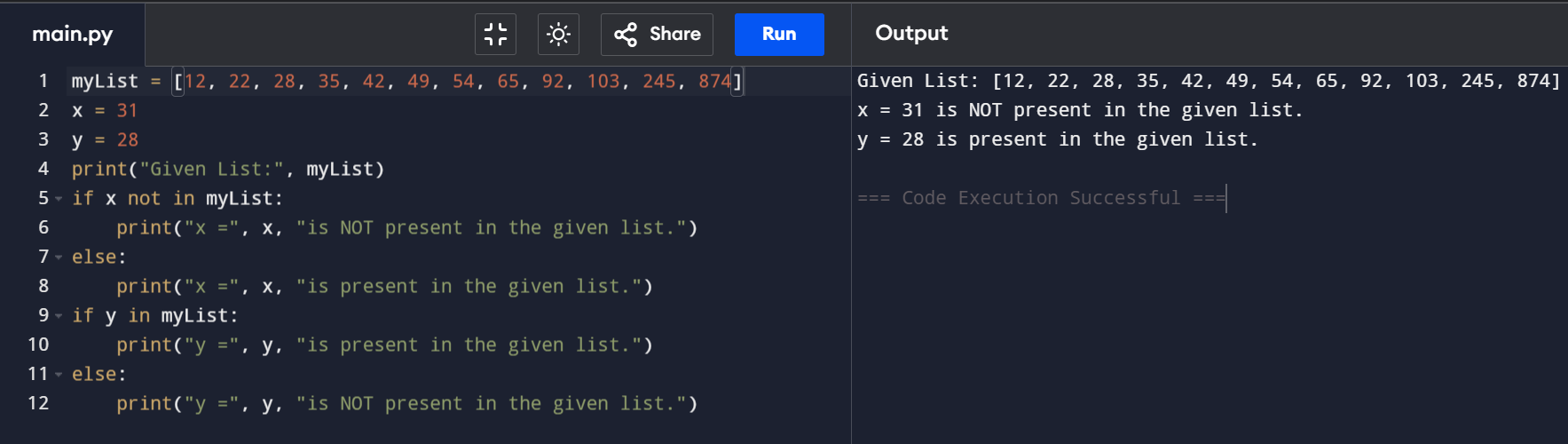
Bitwise operation between 2 numbers



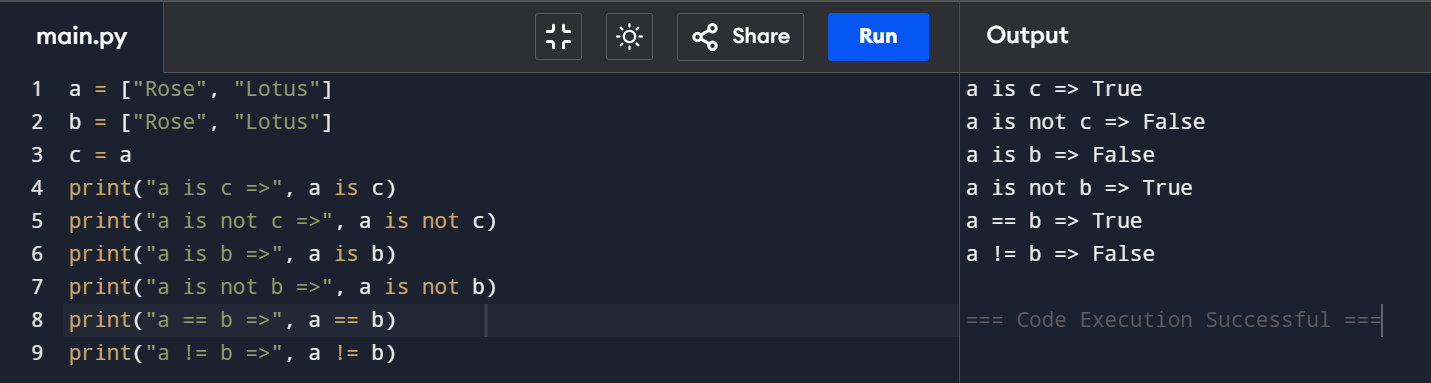
Logical operators



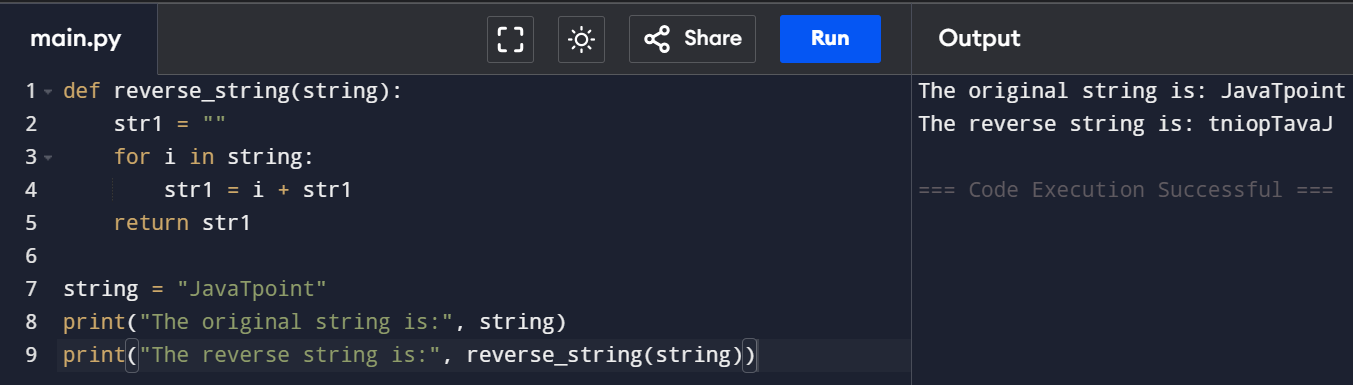
Membership operators



Identity operators



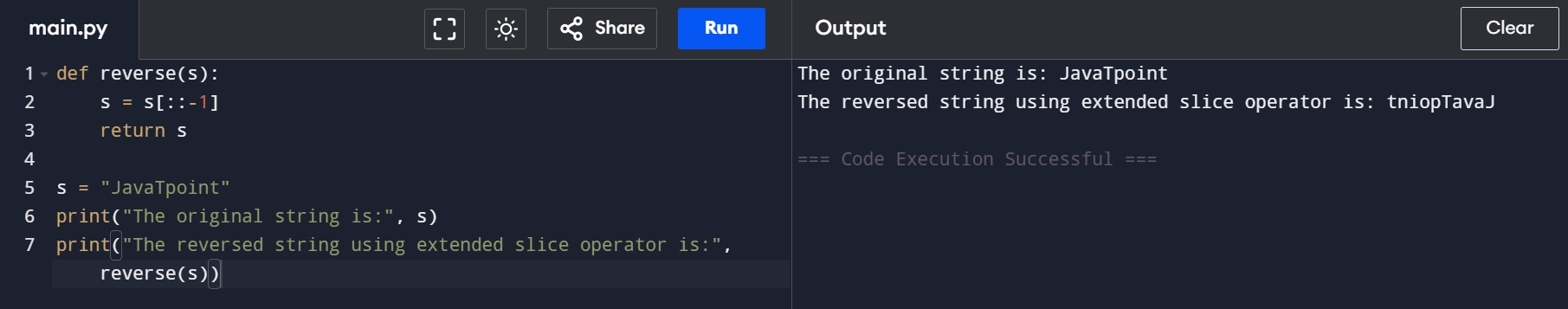
Reverse a string using for loop



Reverse a string using while loop



Reverse a string using slice operator



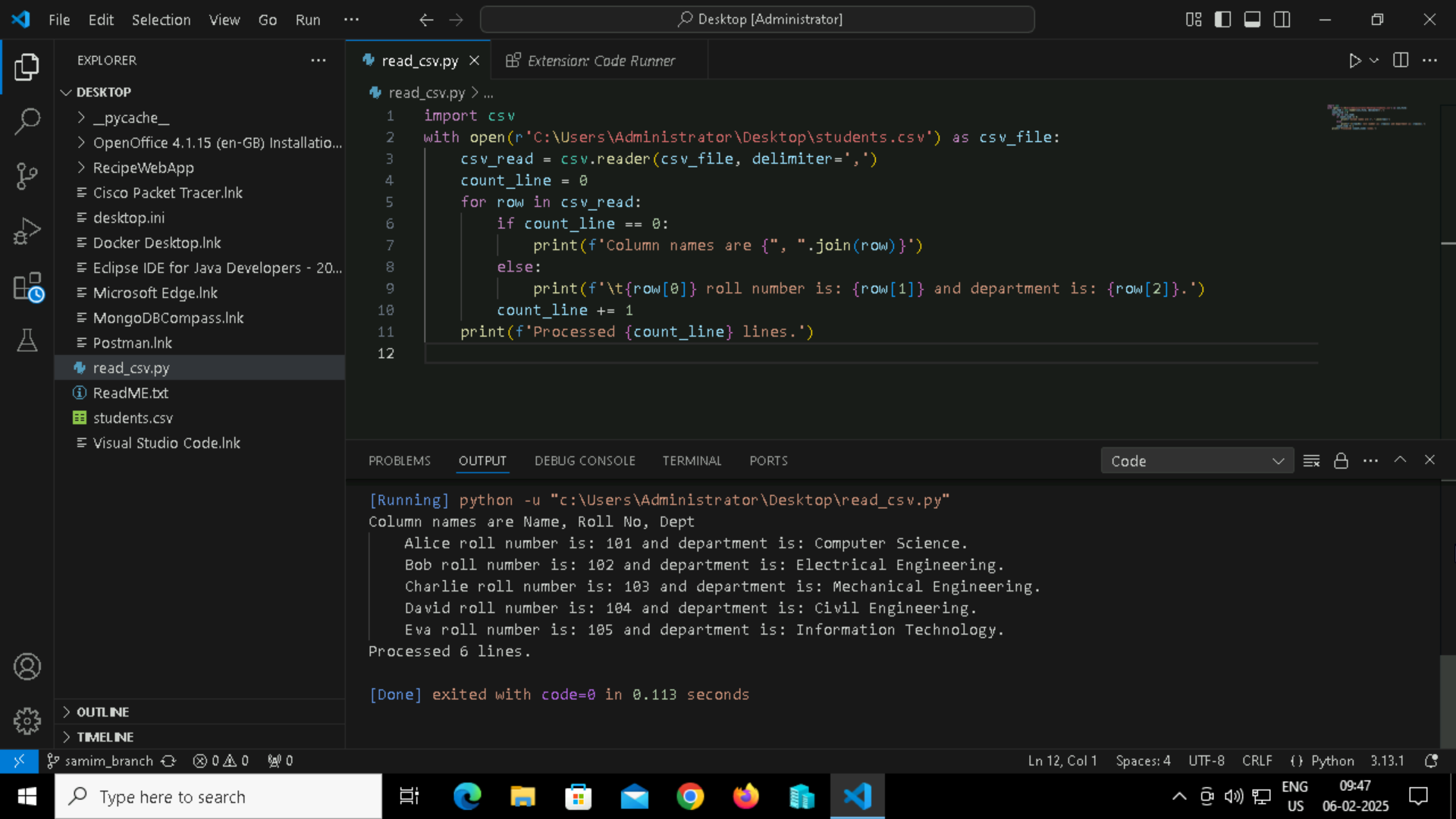
Reverse function using join



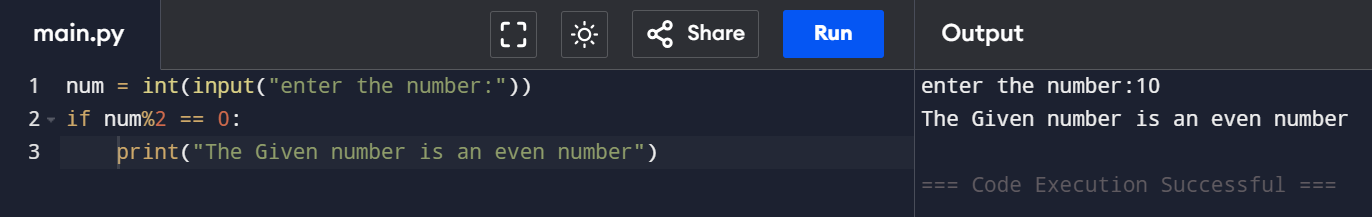
Reverse using recursion



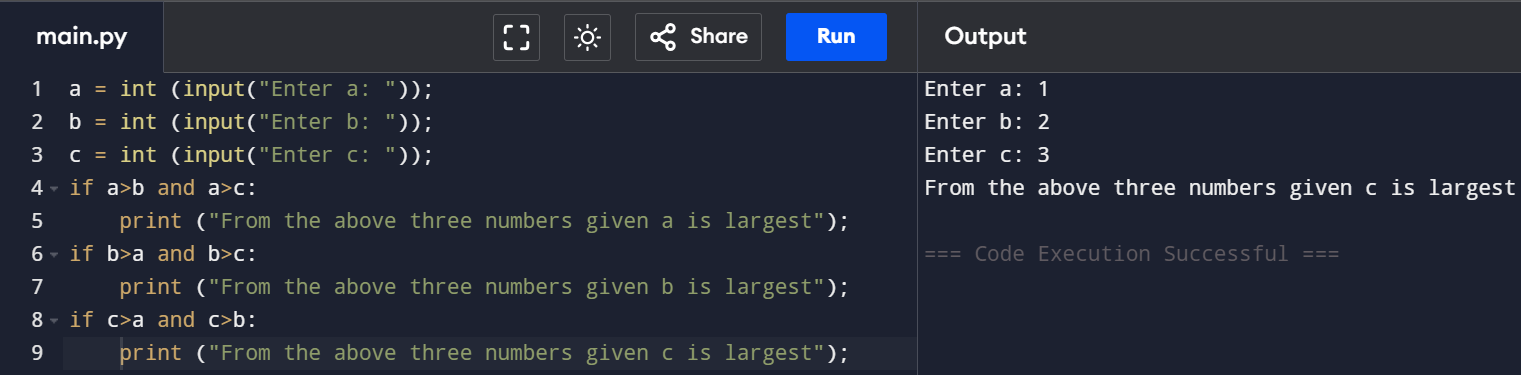
Read a csv file in python



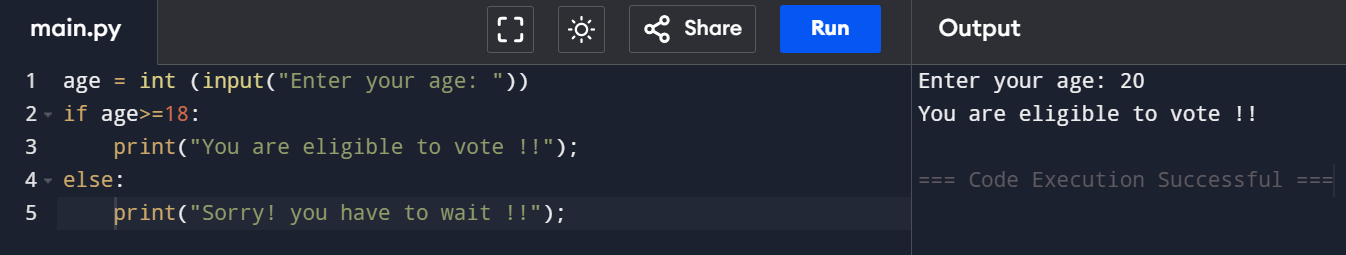
Simple python program to understand if statement



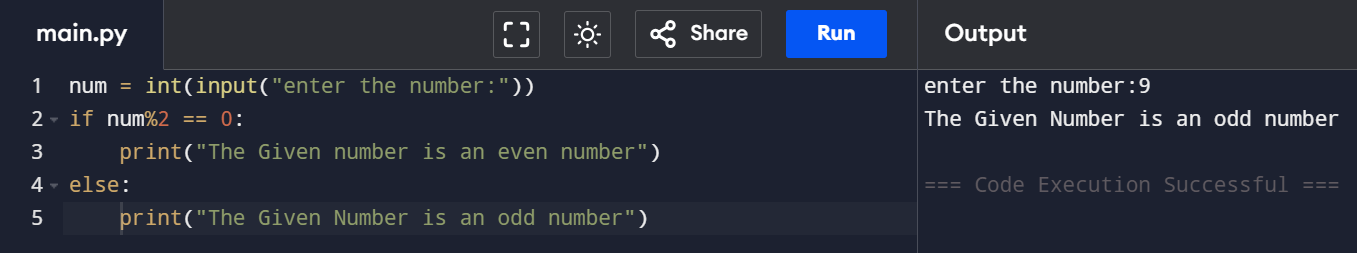
Program to print largest of 3 numbers



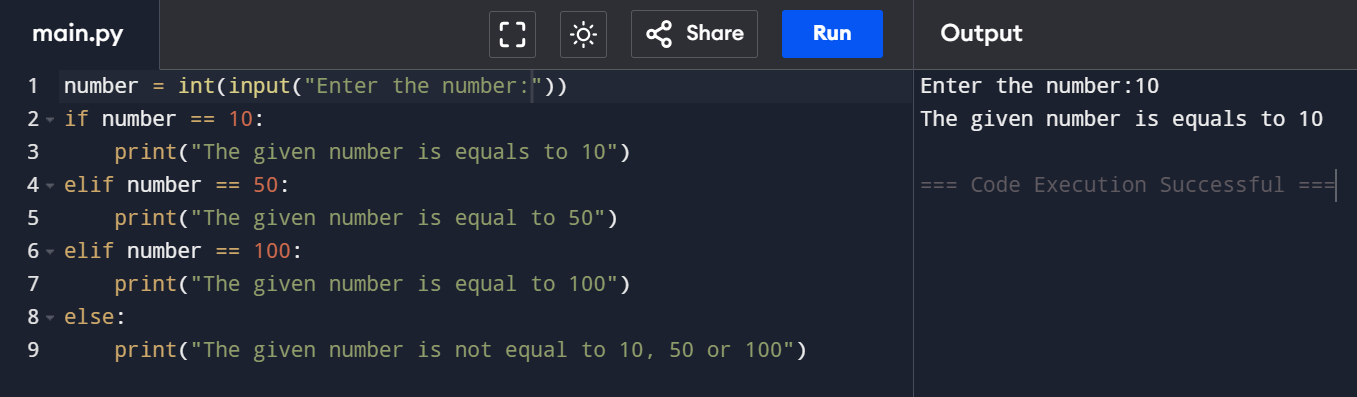
Program to check whether a person is eligible to vote or not



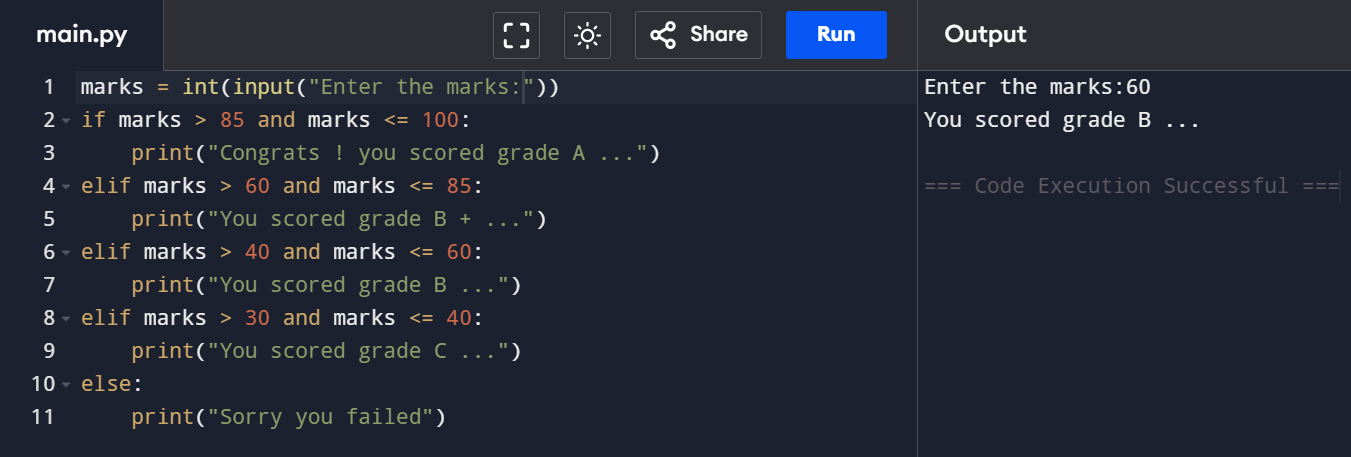
Program to check if the given number is even or odd



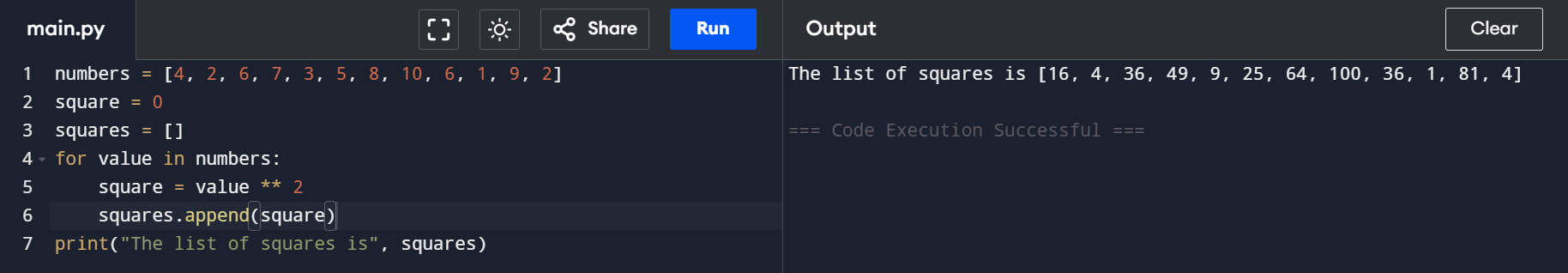
Python program 1 elif statement



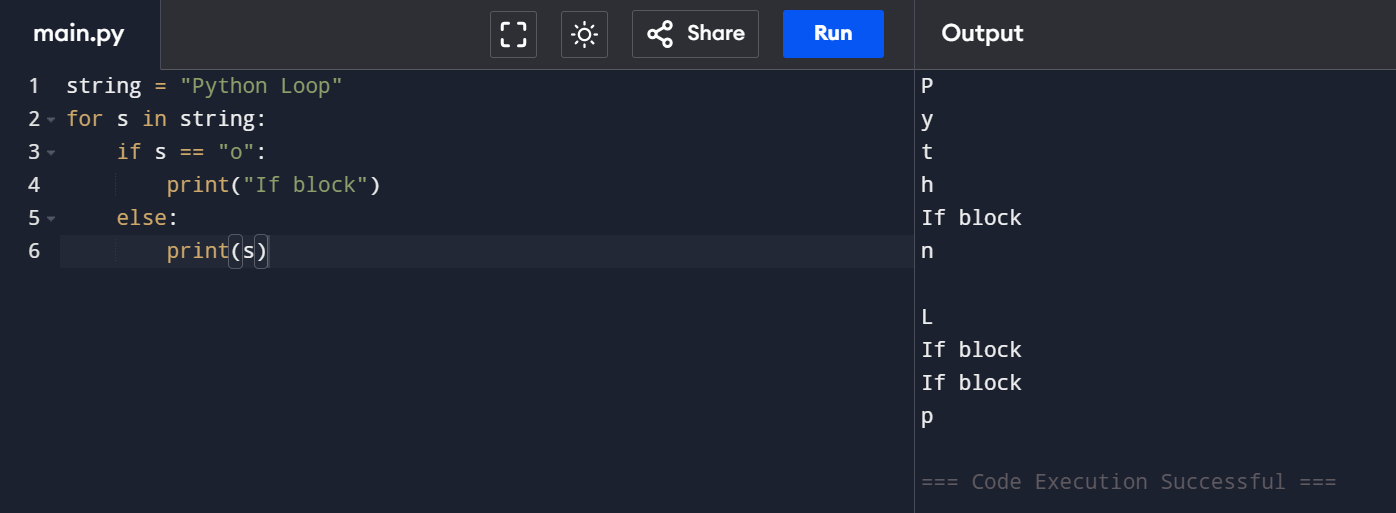
Python program 2 elif statement



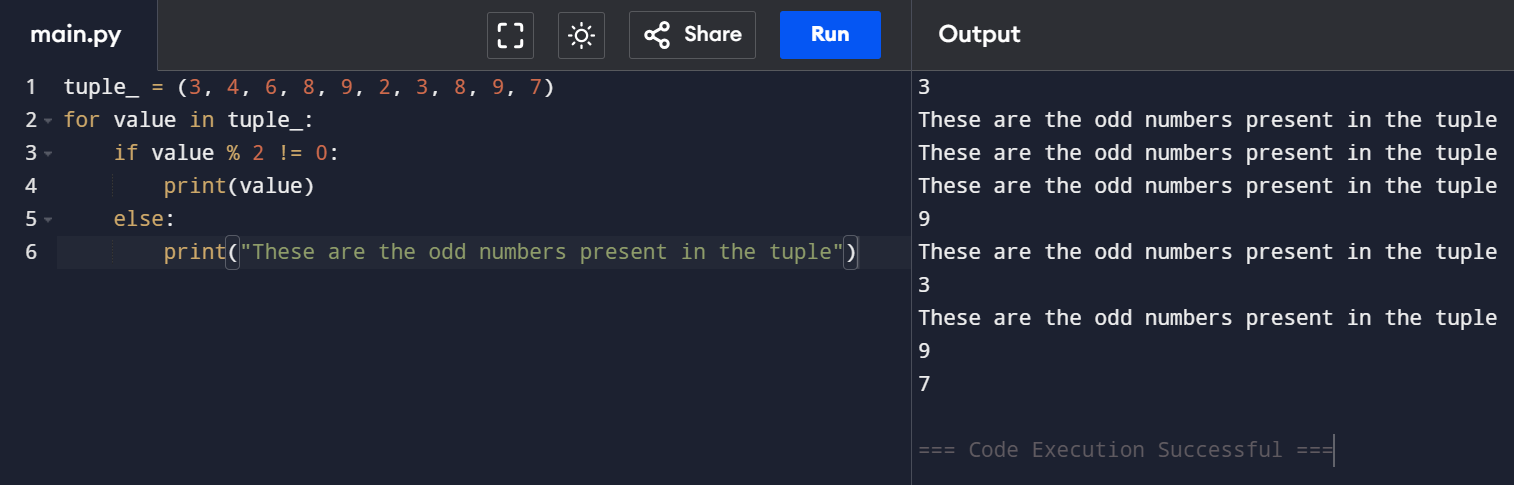
List of squares using for loop



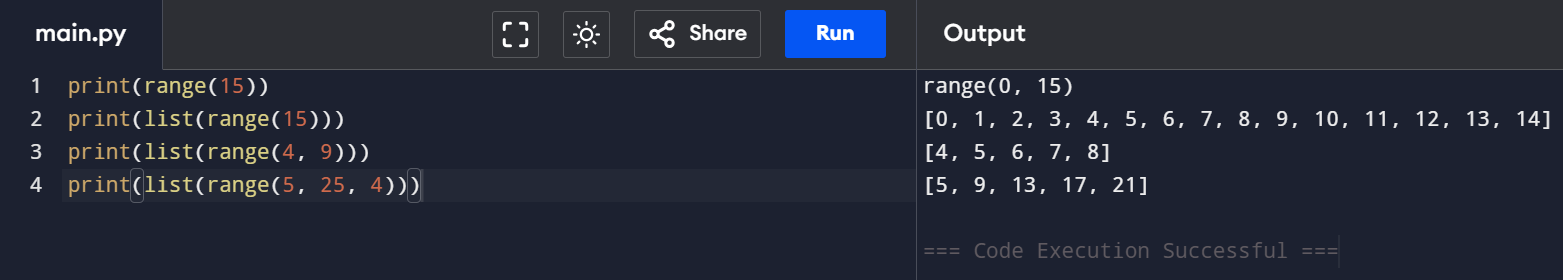
Using loop in string manipulation



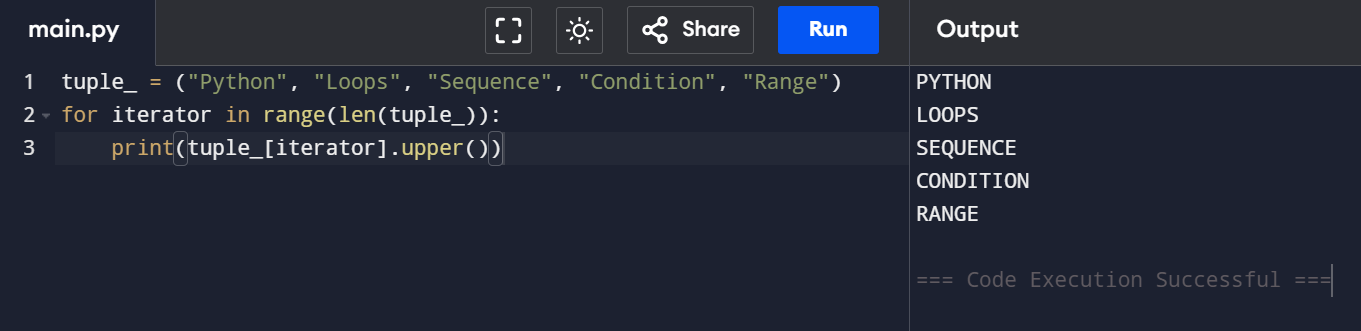
Using else statement with for loop



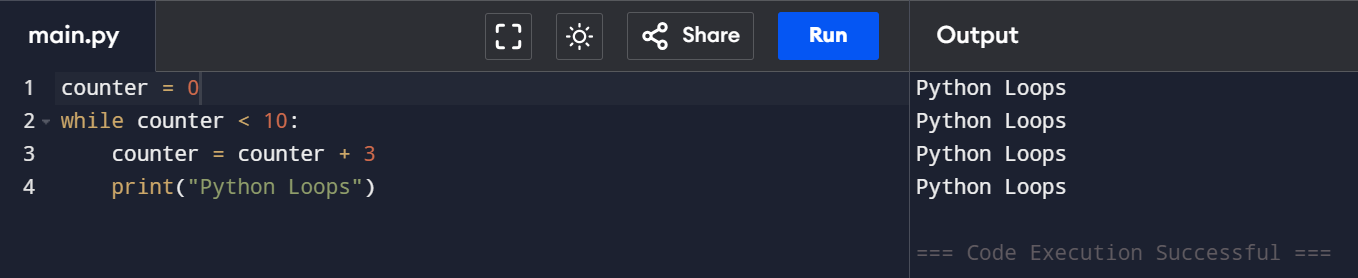
Range function



Python program to iterate over a sequence using with the help of indexing



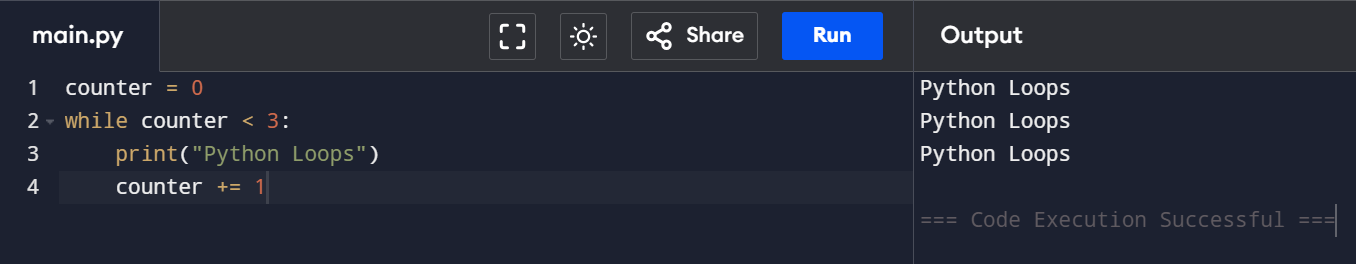
While loop



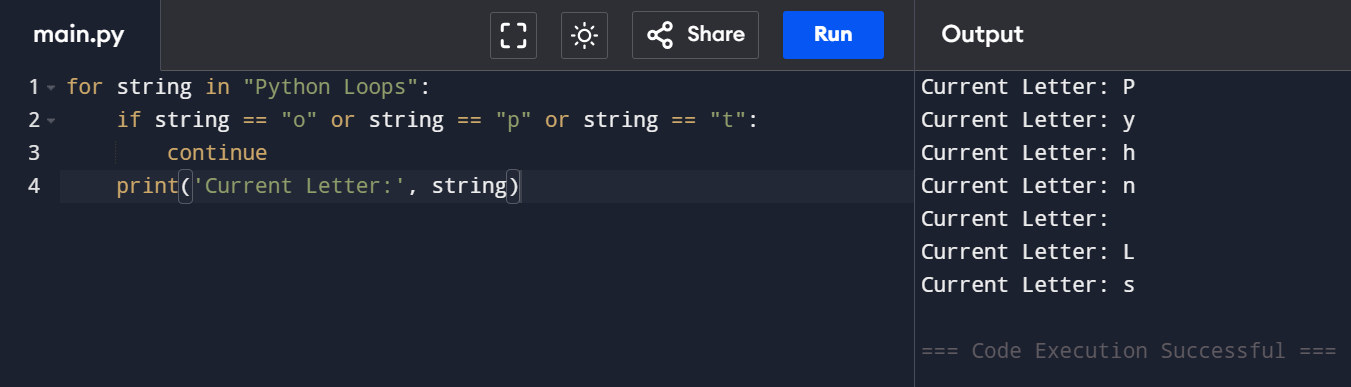
Else statement inside while loop



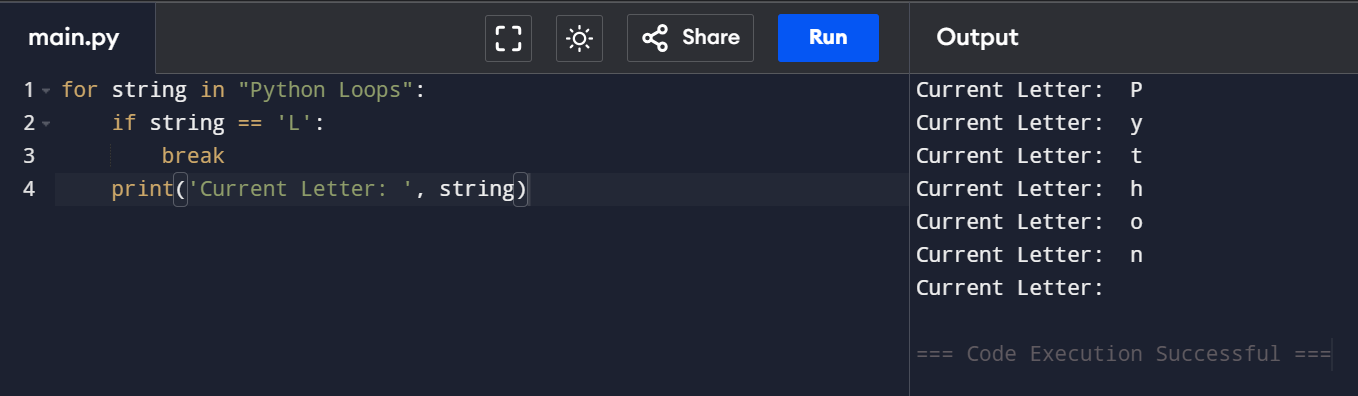
Single statement while block



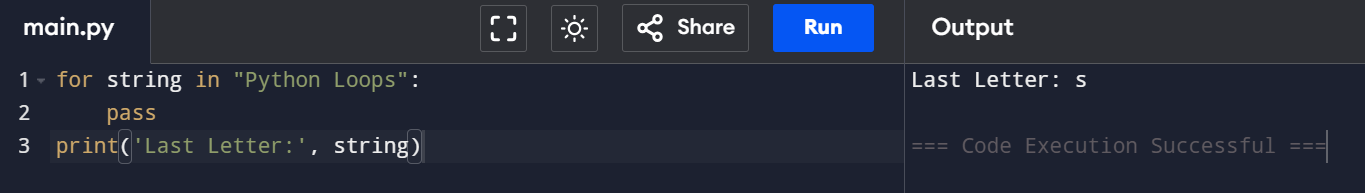
Continue statement



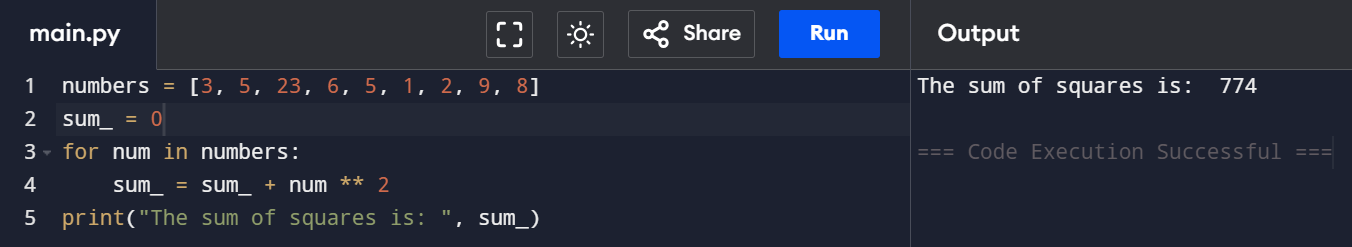
Break statement



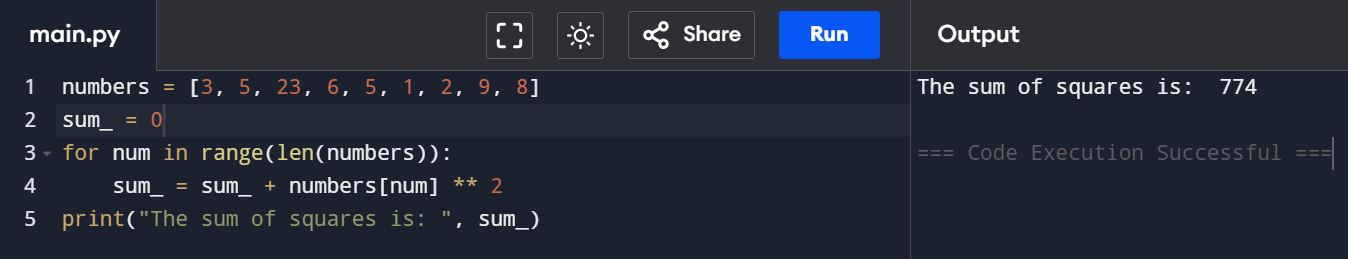
Pass statement



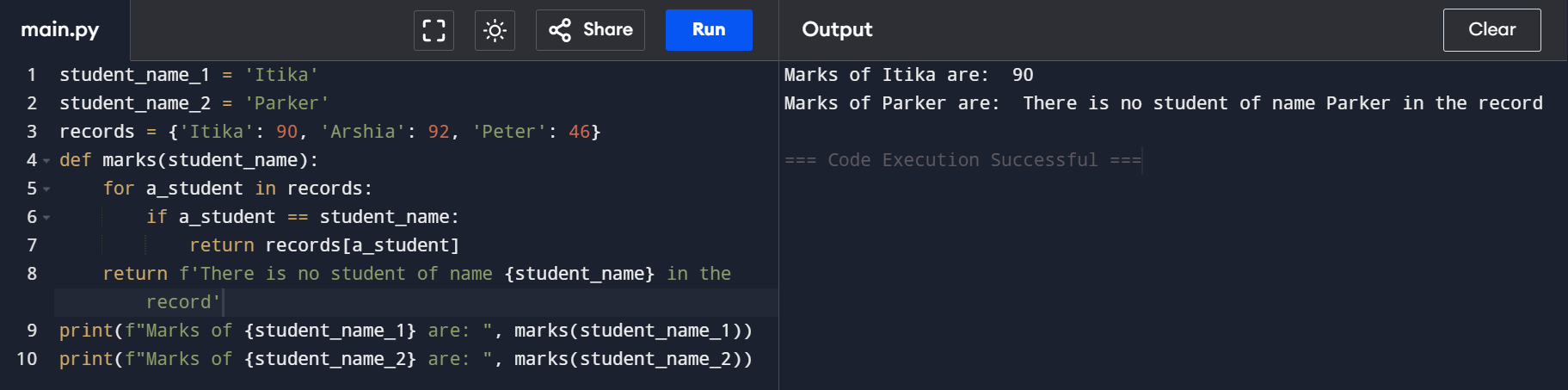
Code to find the sum of squares of each element of the list using for loop



Code to find the sum of squares of each element of the list using for loop



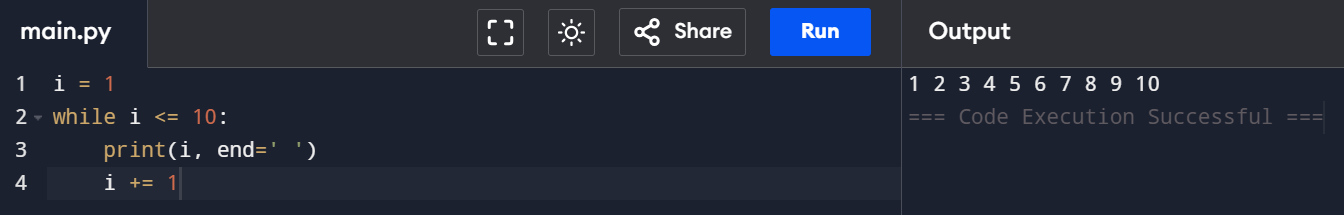
code to print marks of a student from the record



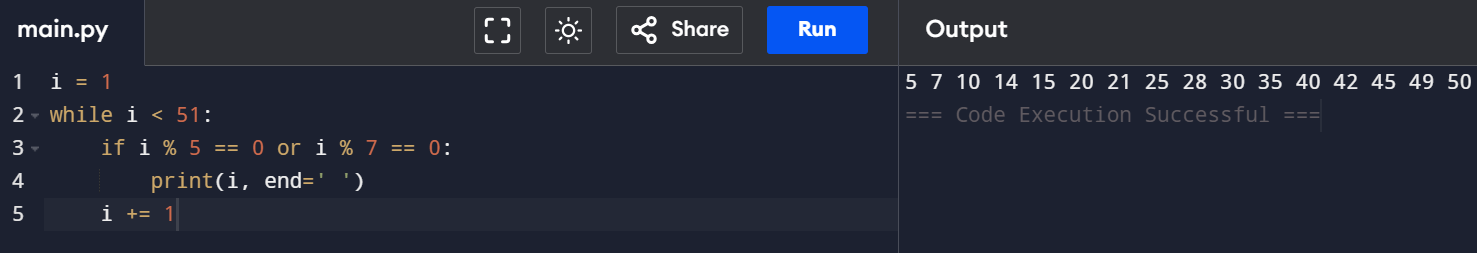
Nested loops



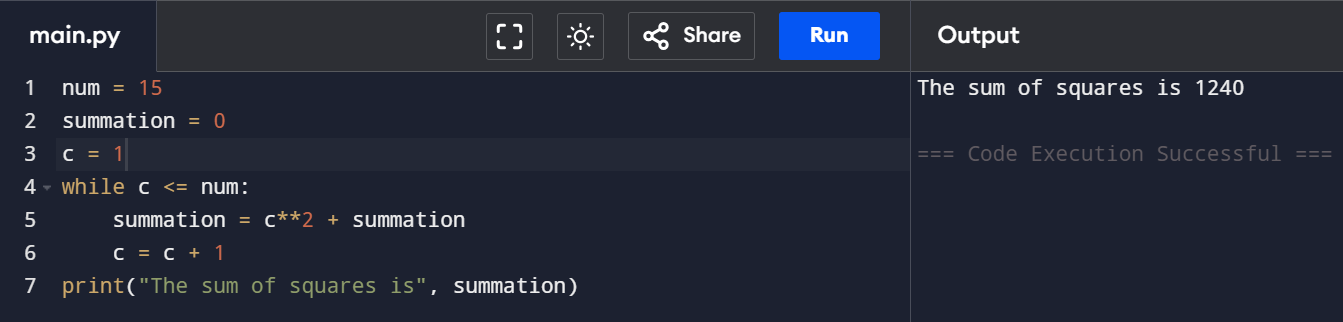
while loops in Python for printing numbers from 1 to 10



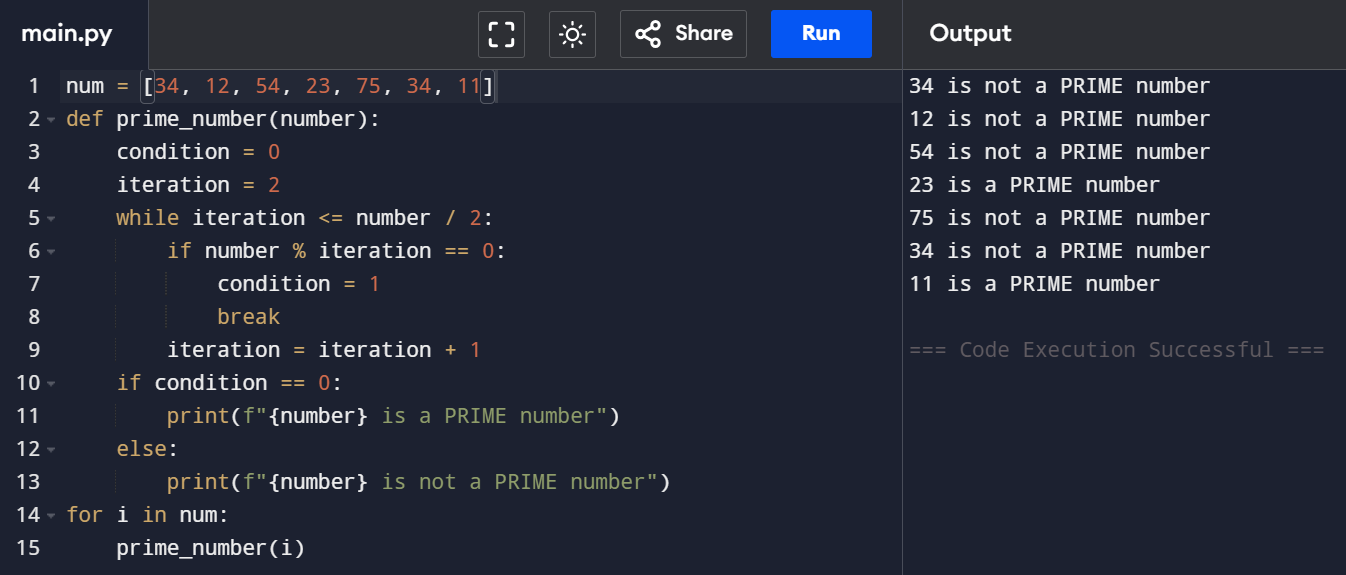
while loops in Python for Printing those numbers divisible by either 5 or 7 within 1 to 50



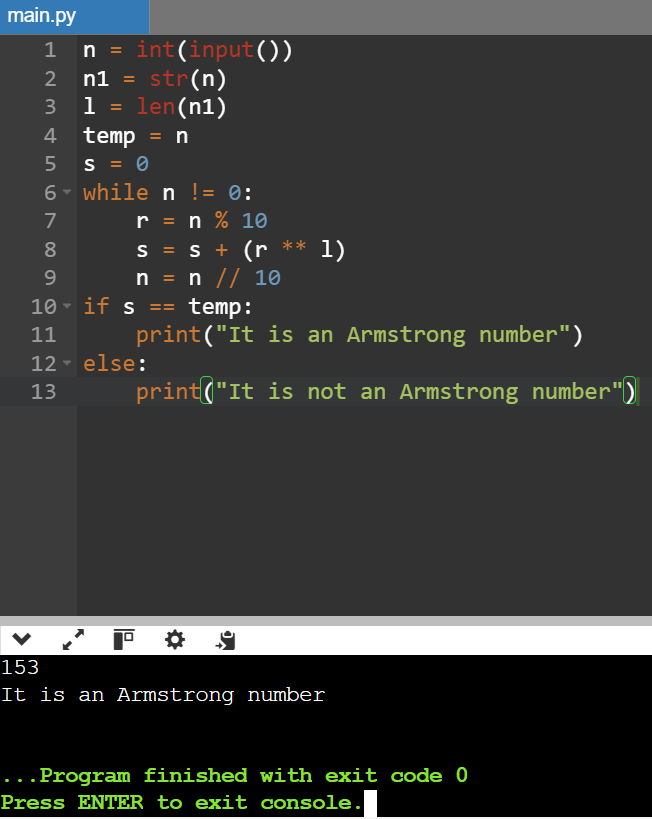
the sum of squares of the first 15 natural numbers using a while loop.



while loops in Python for a number is Prime number or not.



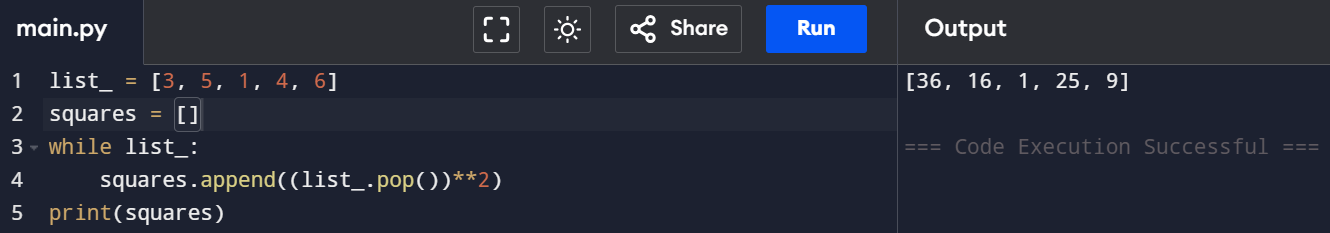
while loops in Python for a number is Armstrong number or not.



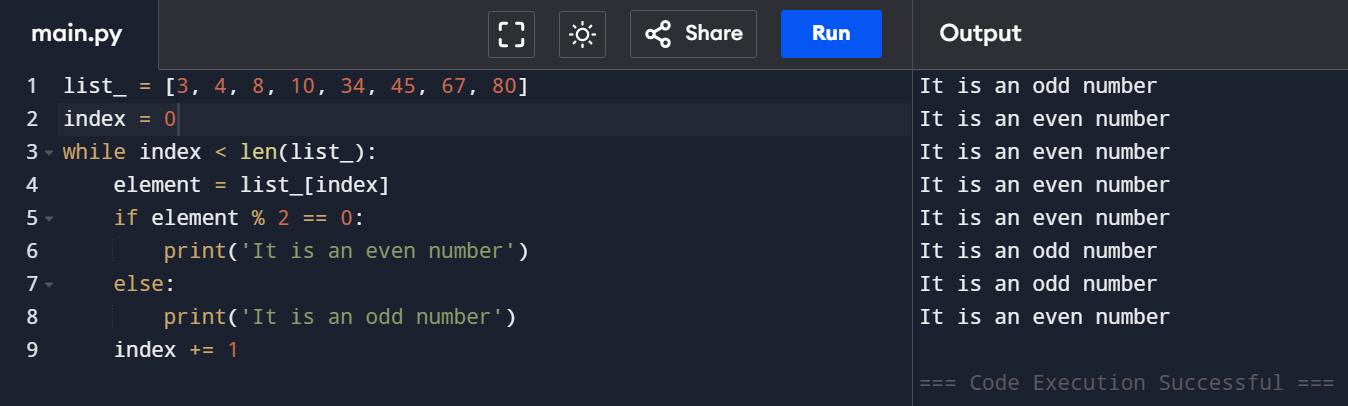
while loop for printing the multiplication table of a given number.



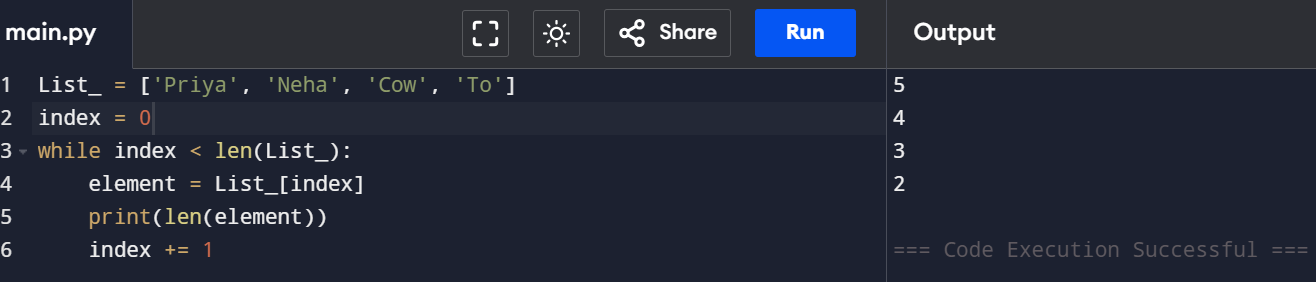
while loops in Python for square every number of a list



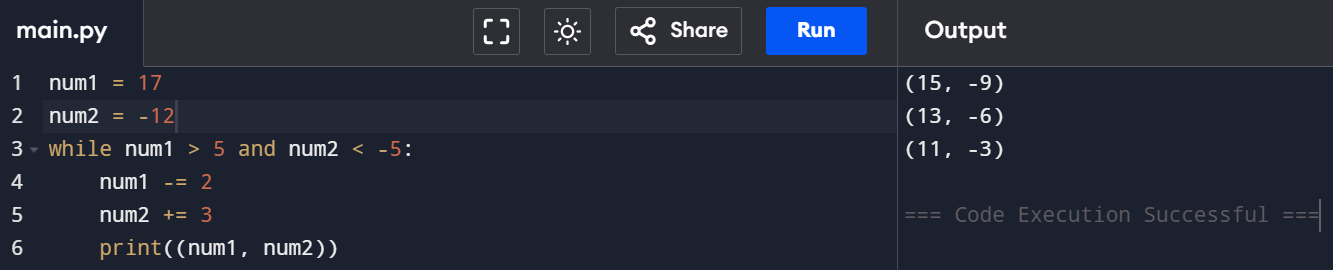
while loops in Python for determine odd and even number from every number of a list



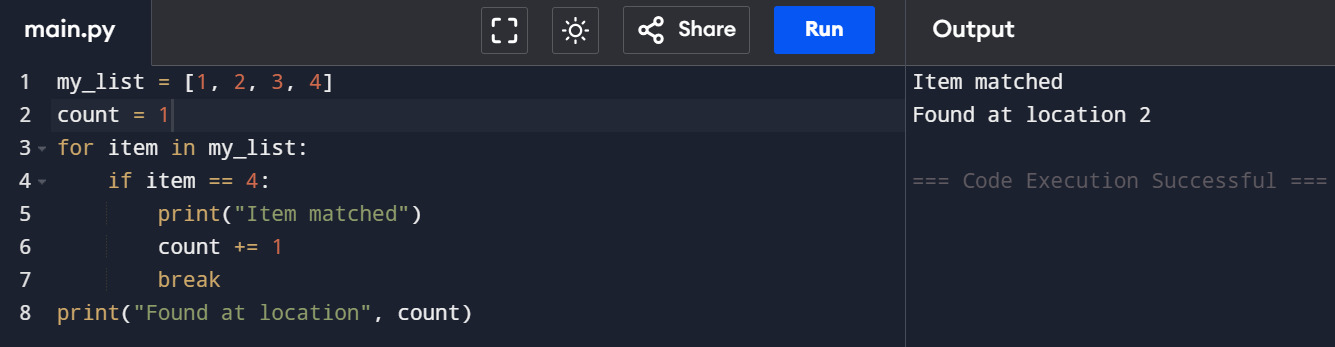
while loops in Python for determine the number letters of every word from the given list.



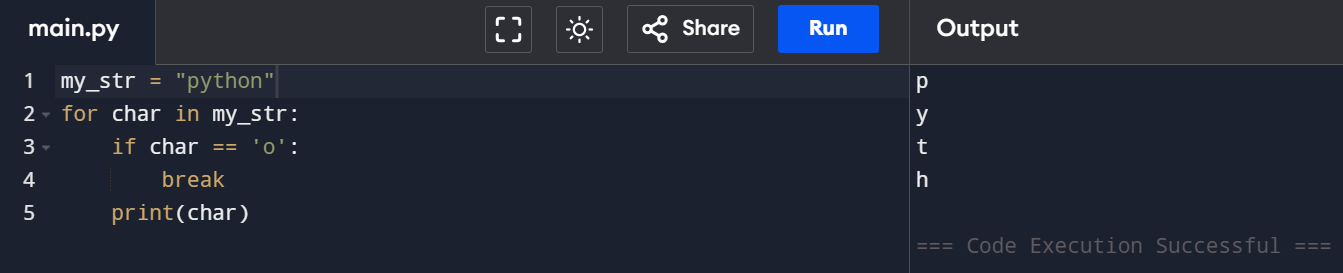
while loops in Python for multiple condition.



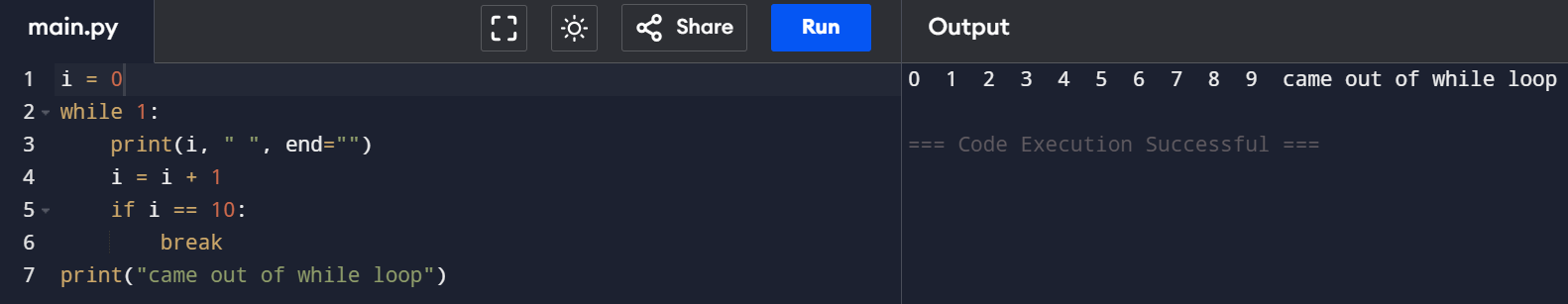
break statement with for loop



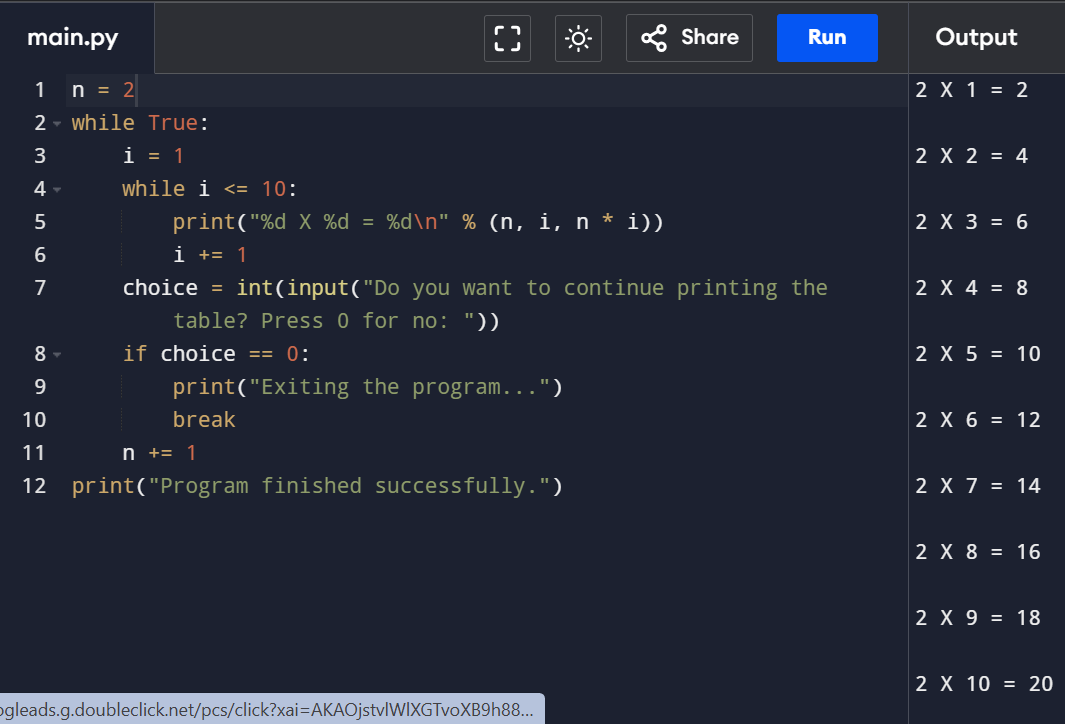
Breaking out of a loop early



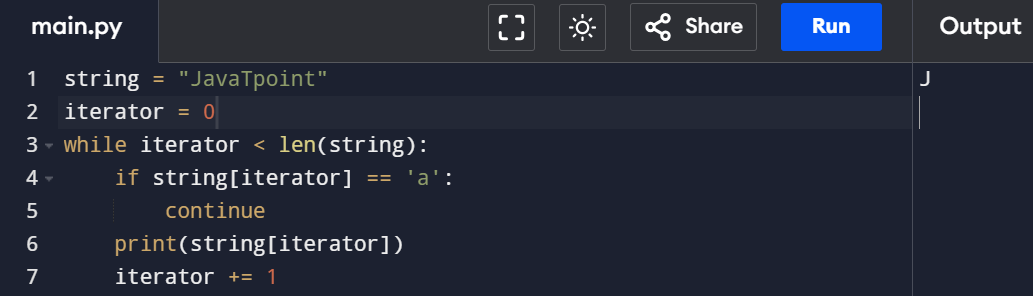
break statement with while loop



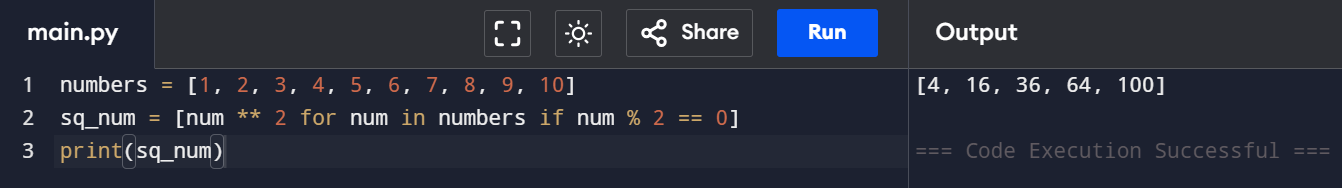
break statement with nested loops



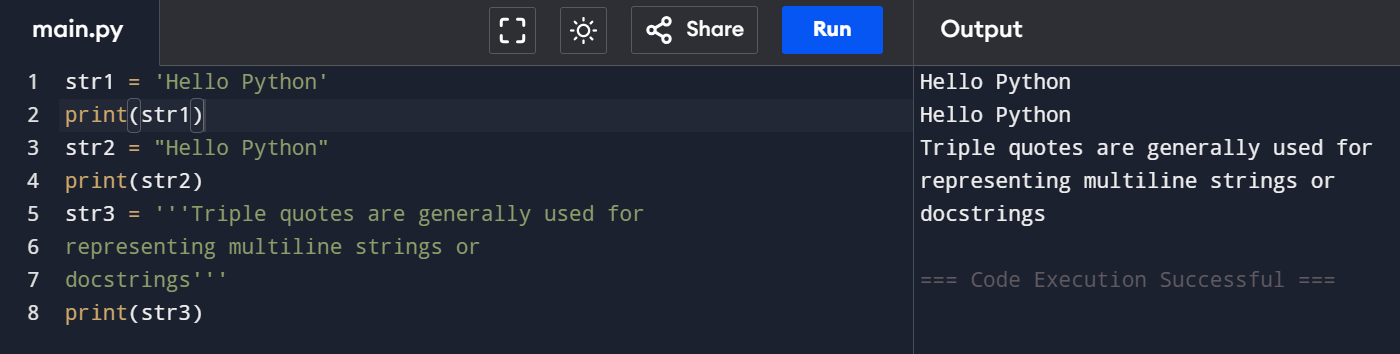
Python Continue Statements in while Loop



Python Continue statement in list comprehension

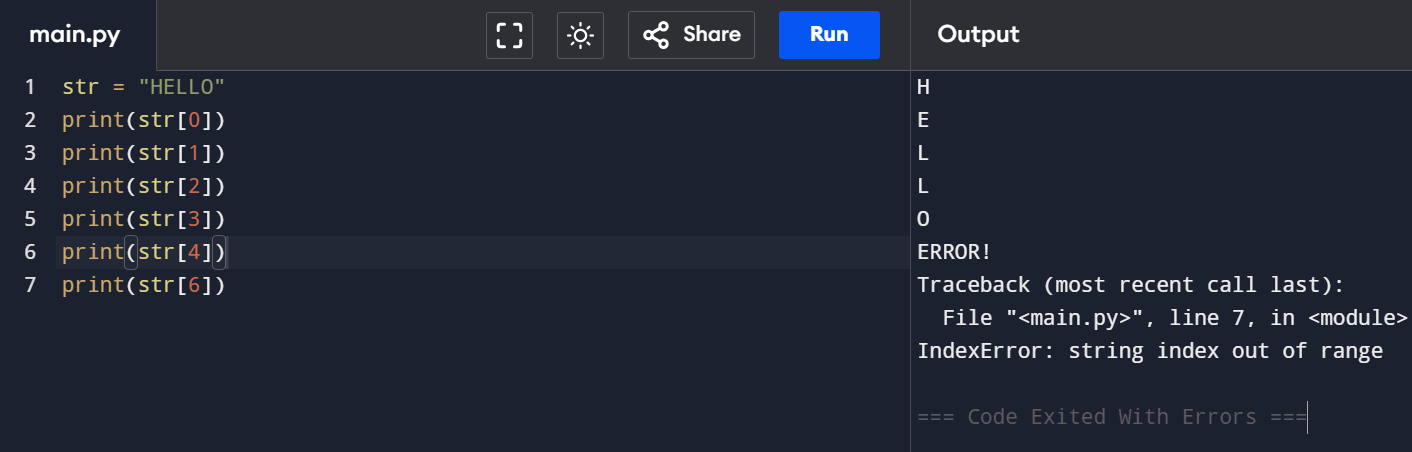


Creating String in Python

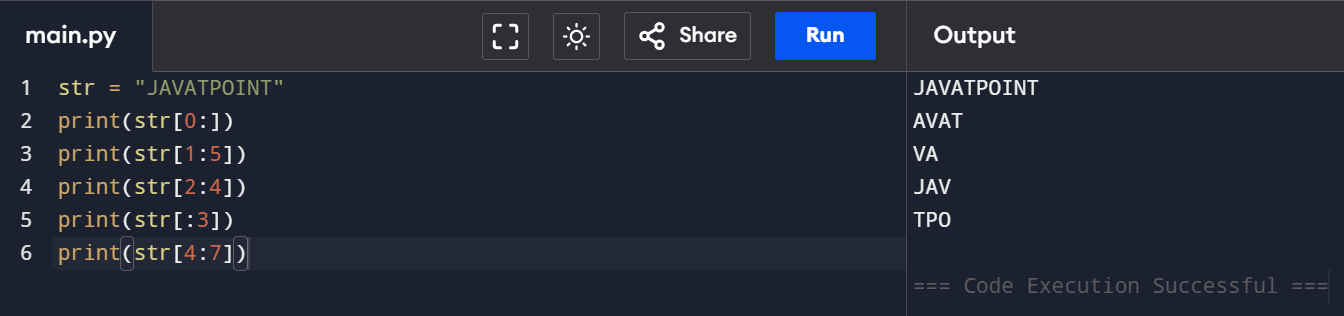


Strings indexing and splitting

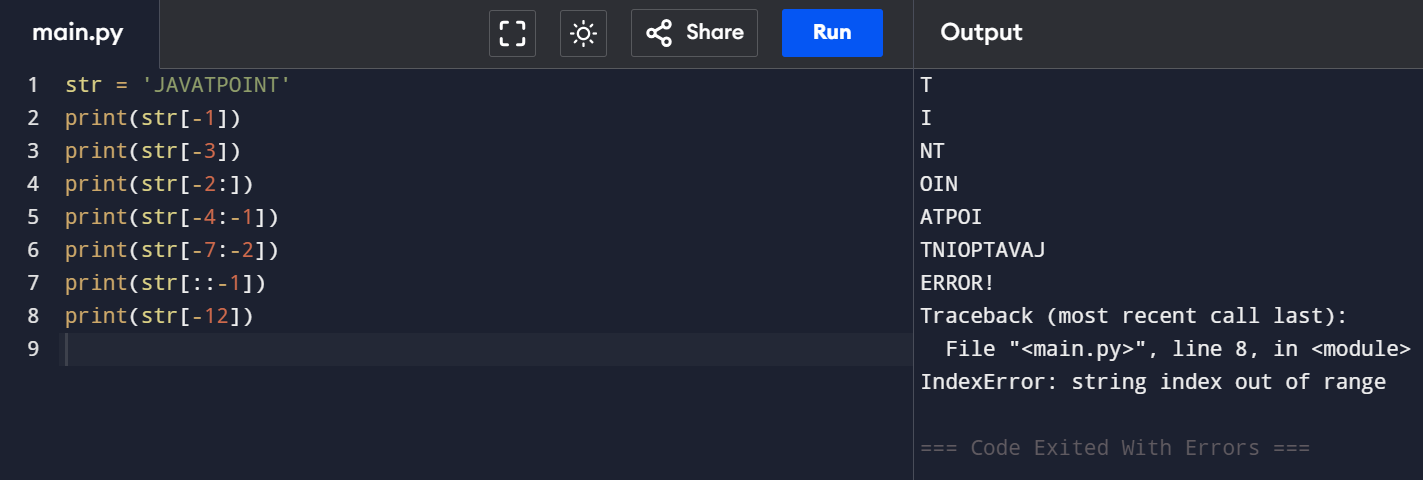
Example 1



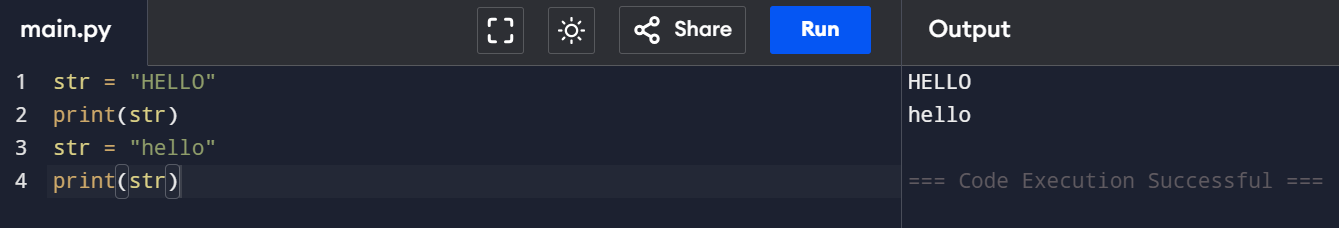
Example 2



Reverse

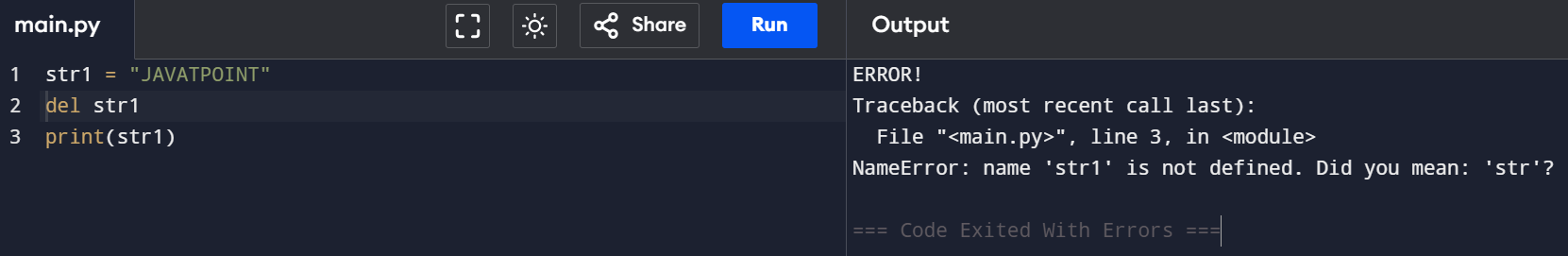


Reassigning Strings

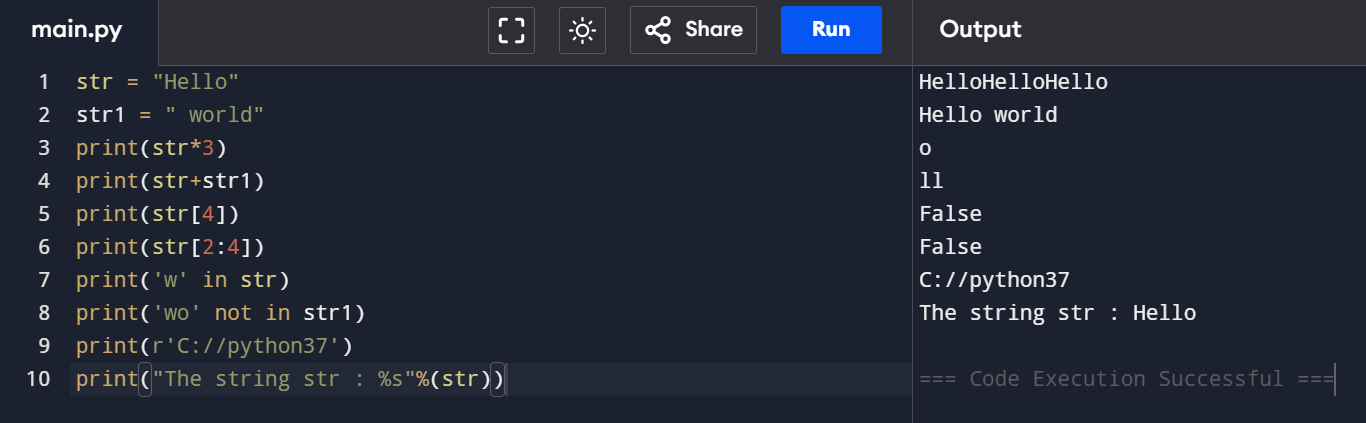


Deleting the String

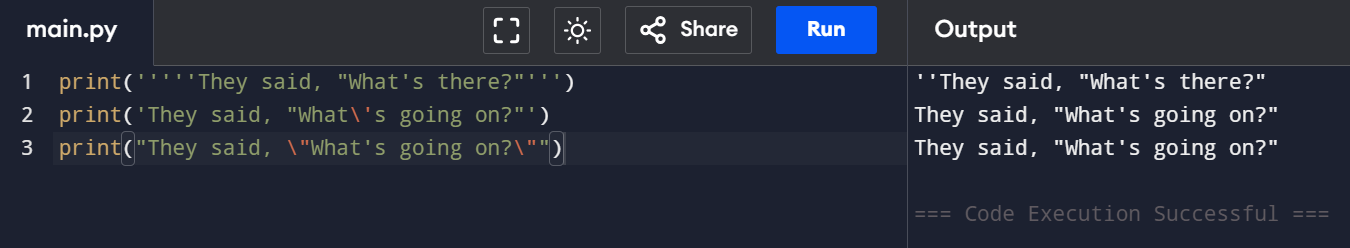
Deleting a string



use of Python operators



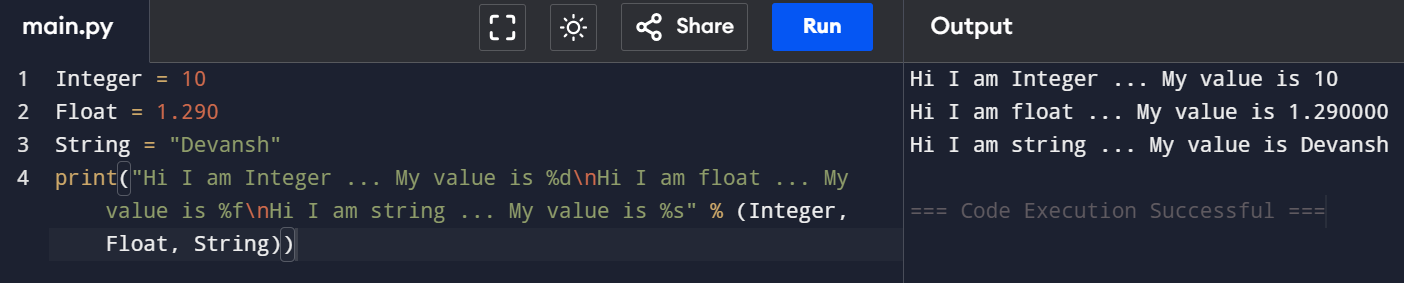
Escape Sequence



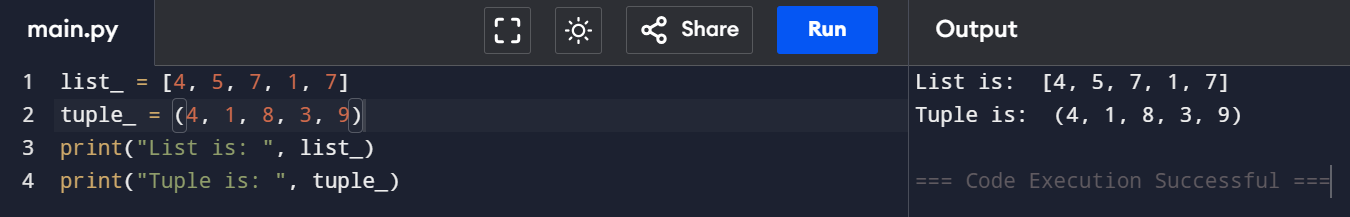
The format() method



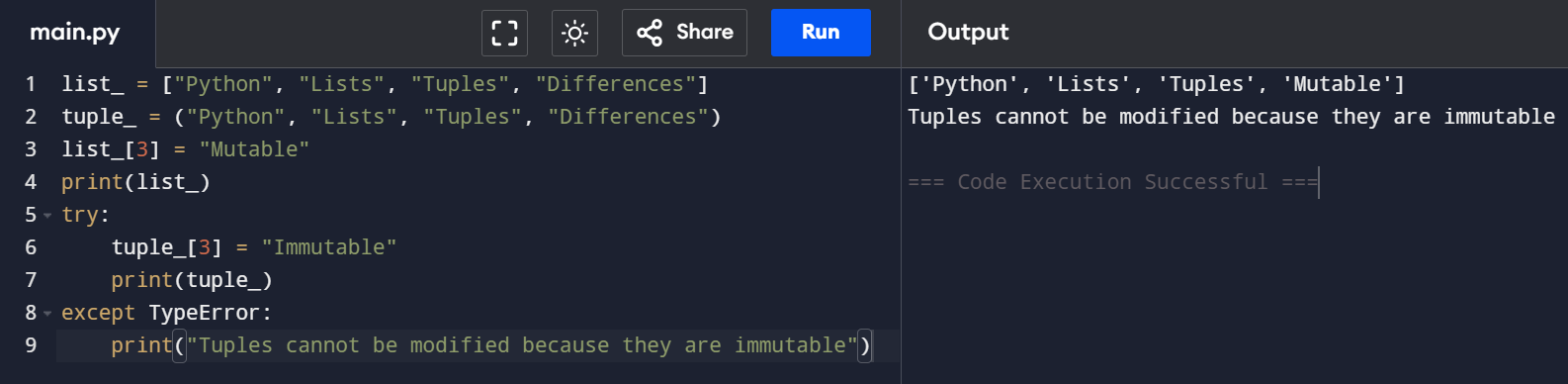
Python String Formatting Using % Operator



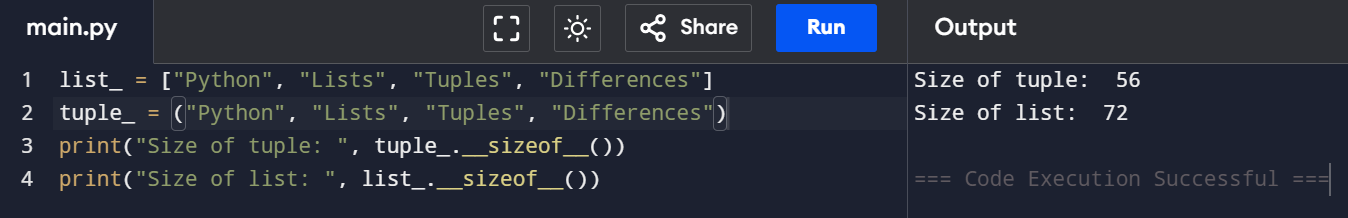
Python code to show the difference between creating a list and a tuple



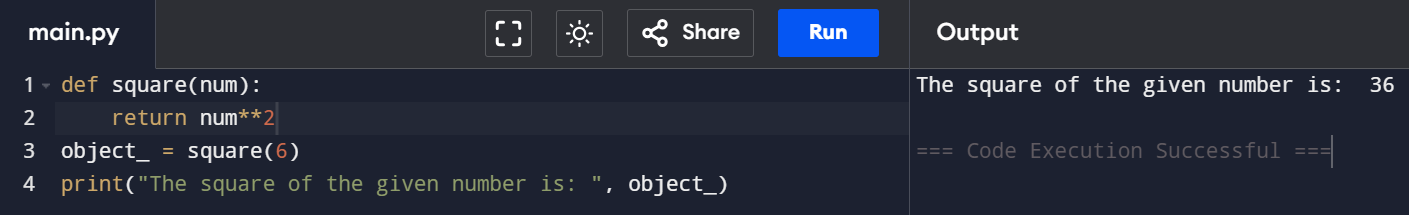
Updating the element of list and tuple at a particular index



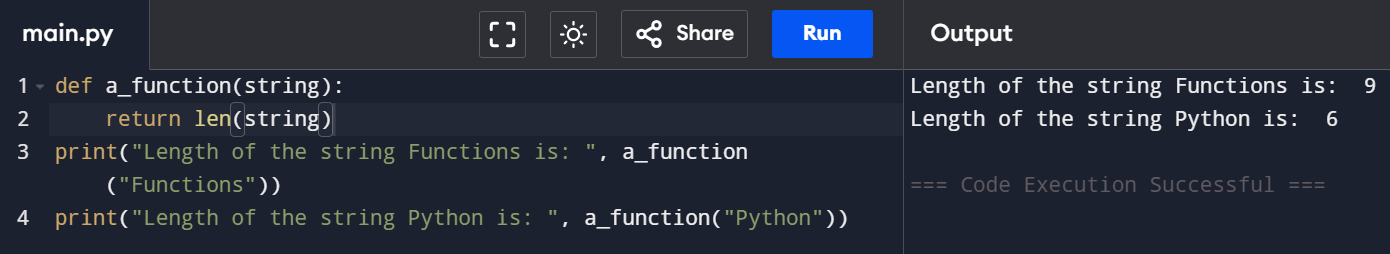
Code to show the difference in the size of a list and a tuple



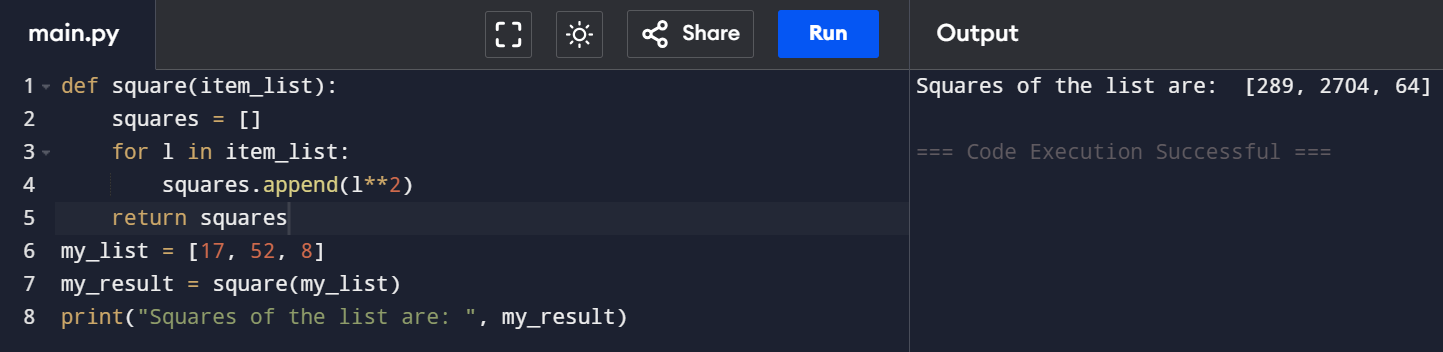
Python Code for User-Defined function



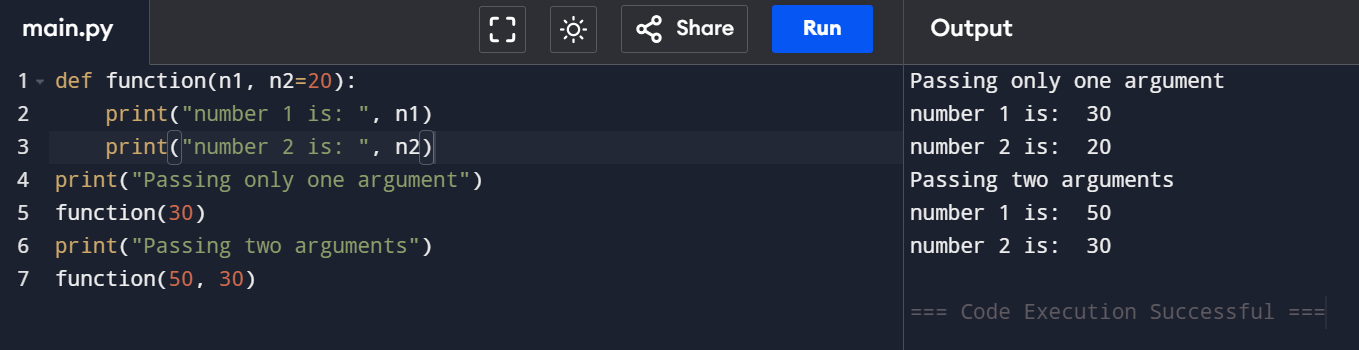
Python Code for calling a function



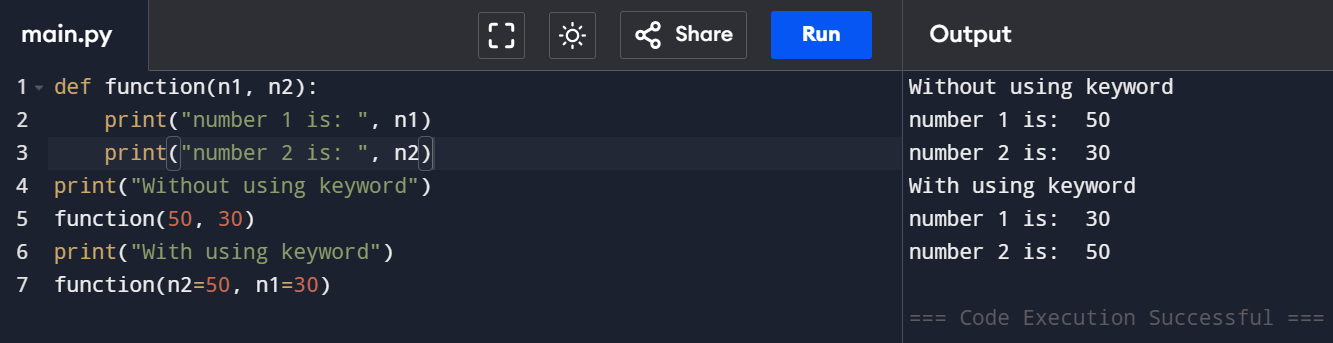
Python Code for Pass by Reference vs. Value



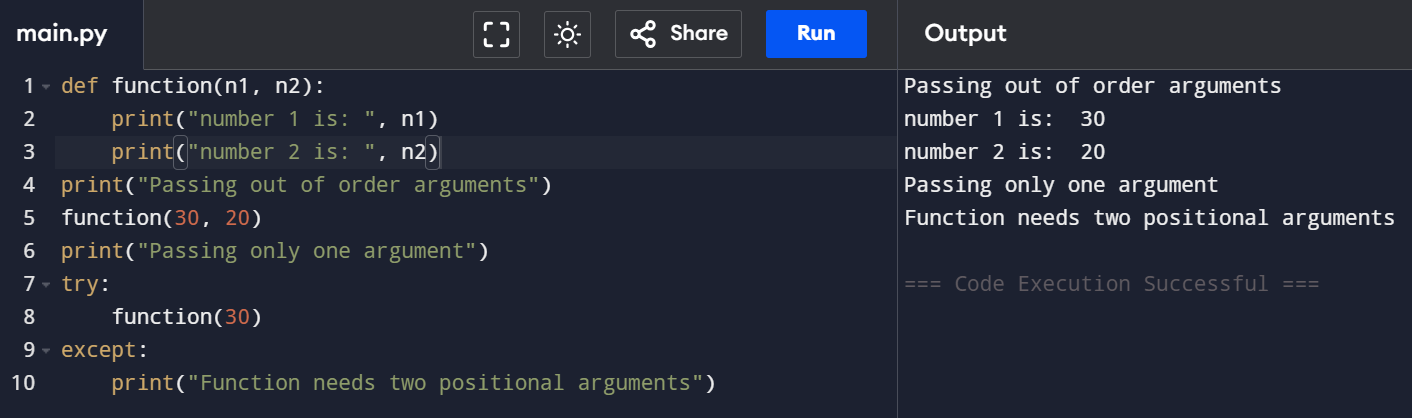
Python code to demonstrate the use of default arguments



Python code to demonstrate the use of keyword arguments



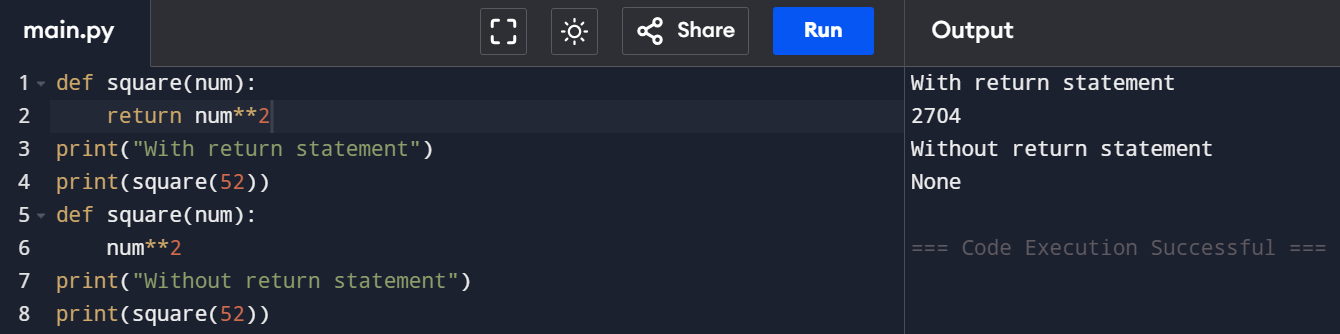
Python code to demonstrate the use of default arguments



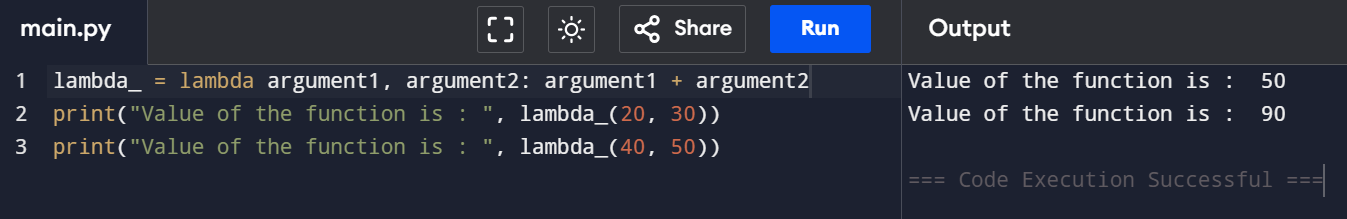
Python code to demonstrate the use of variable-length arguments



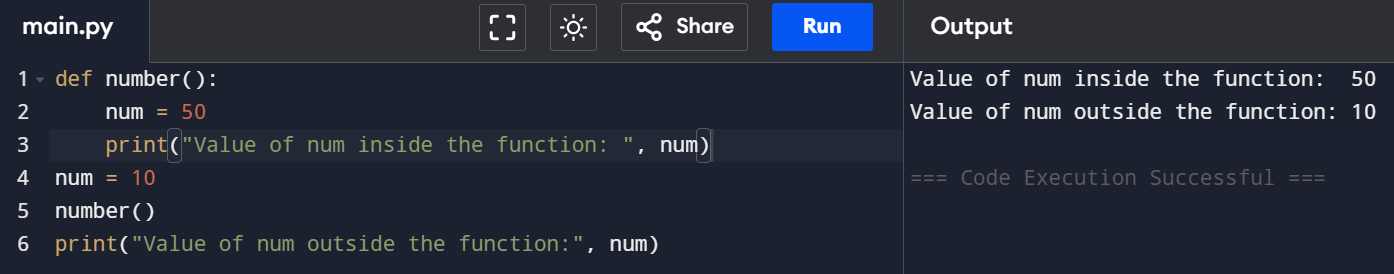
Python code to demonstrate the use of return statements



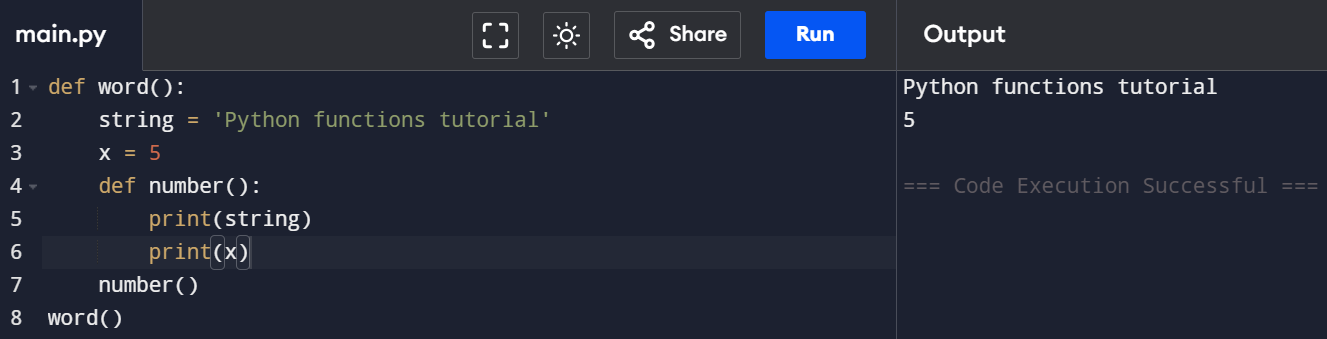
Python code to demonstrate ananymous functions



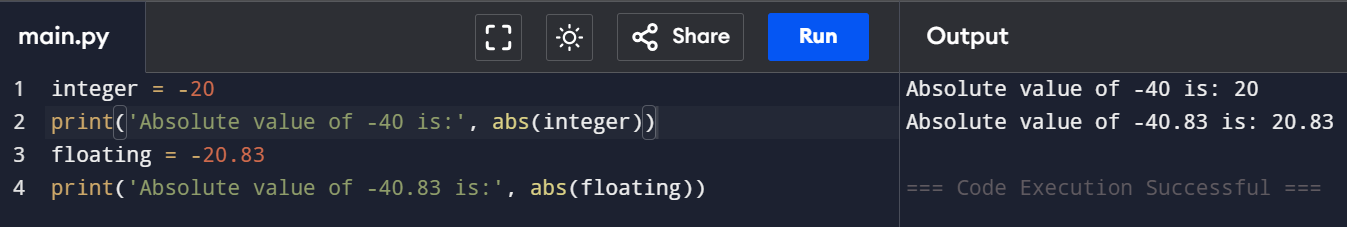
Python code to demonstrate scope and lifetime of variables



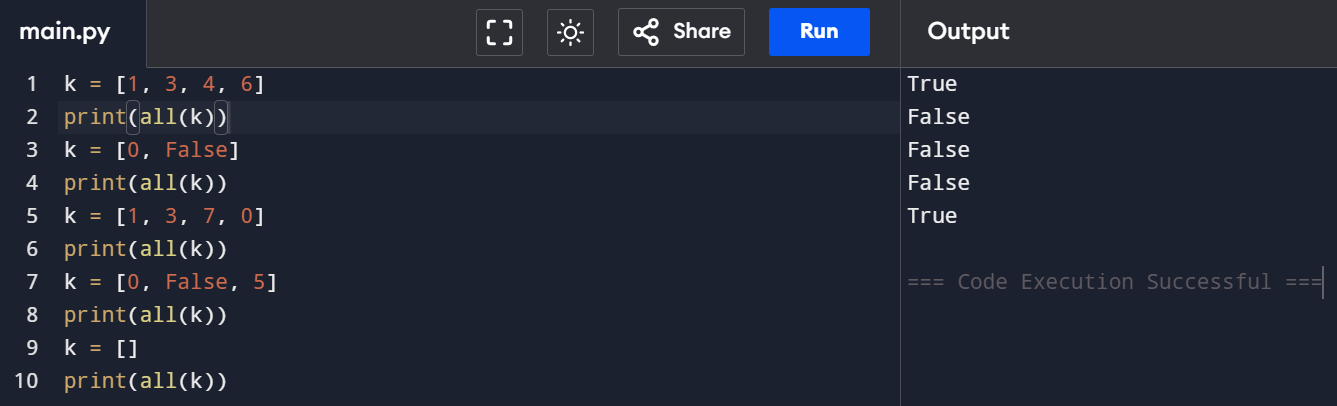
Python code to show how to access variables of a nested functions



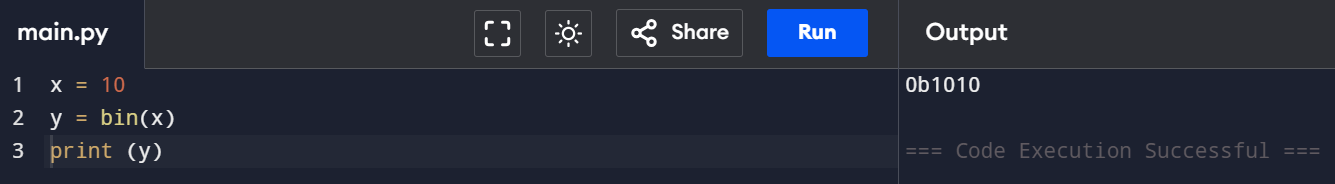
Python abs() Function



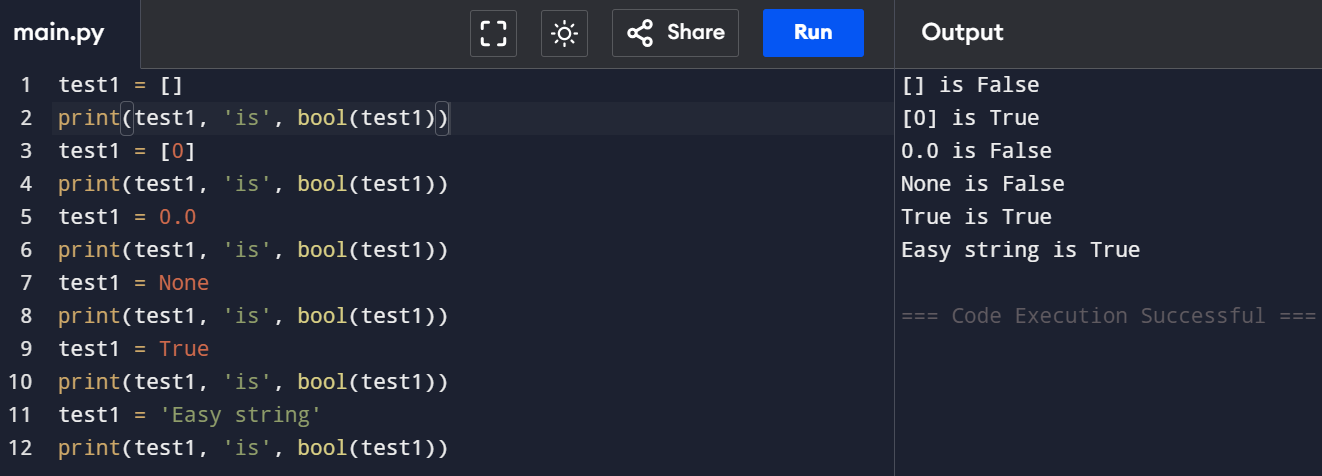
Python all() Function



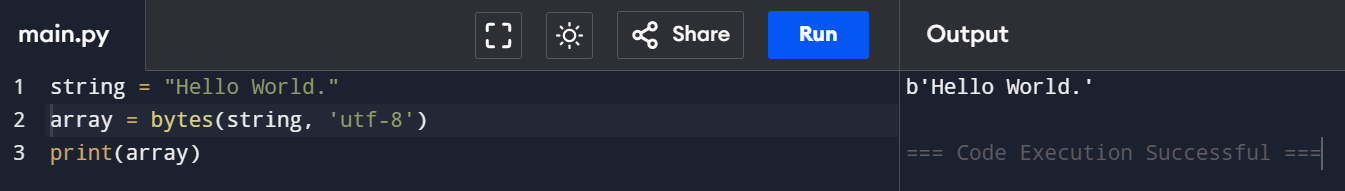
Python bin() Function



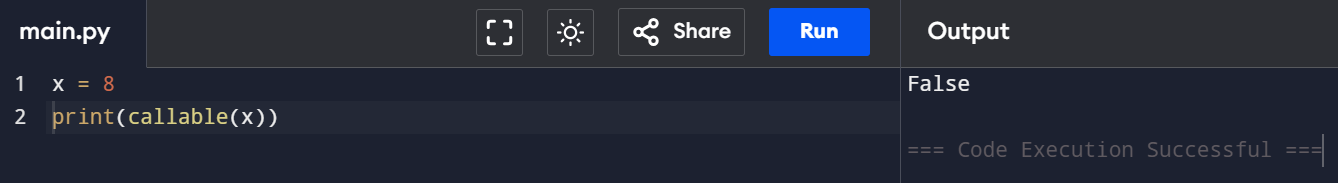
Python bool()



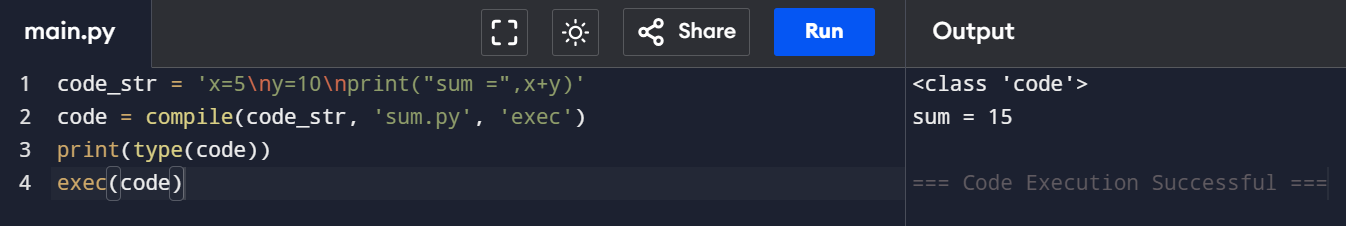
Python bytes()



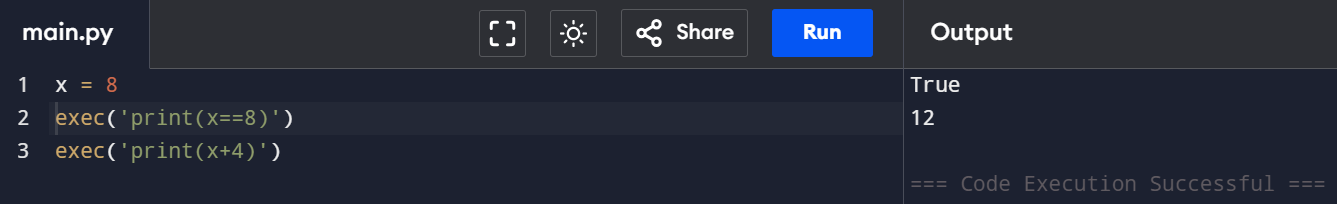
Python callable()



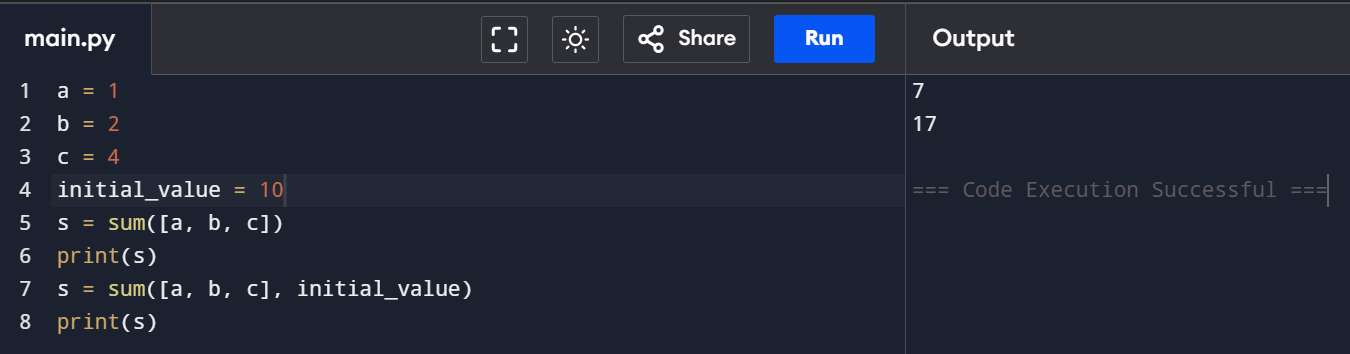
Python compile()



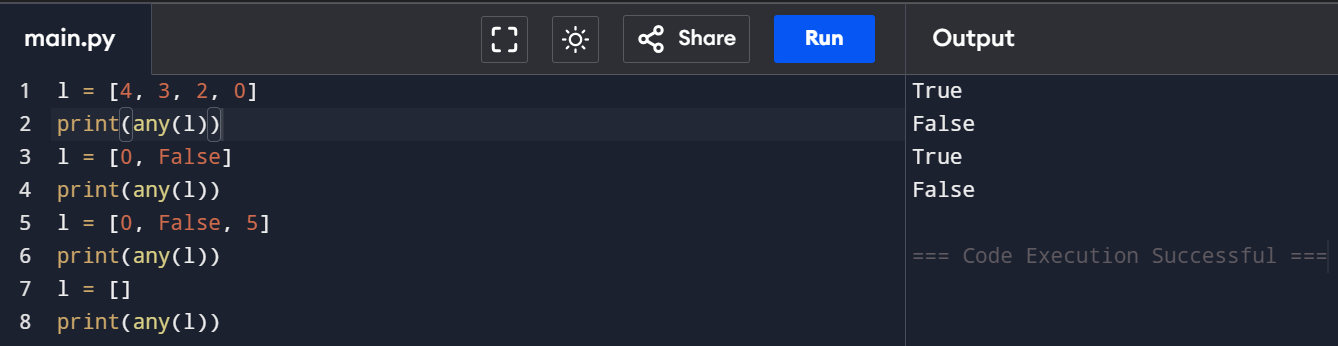
Python exec()



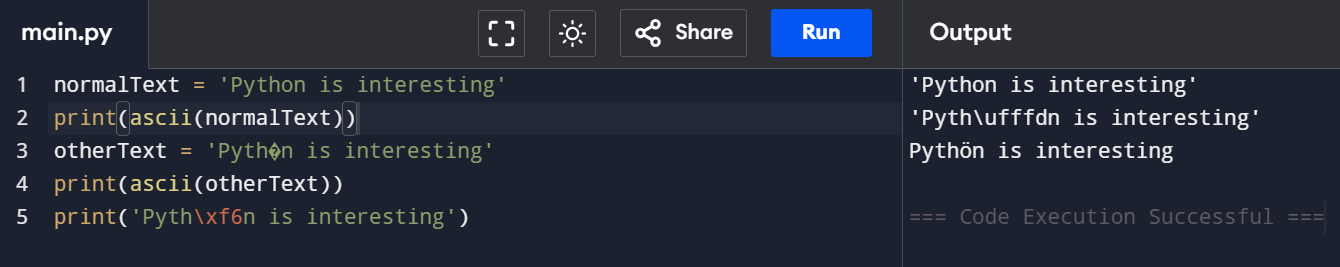
Python sum()



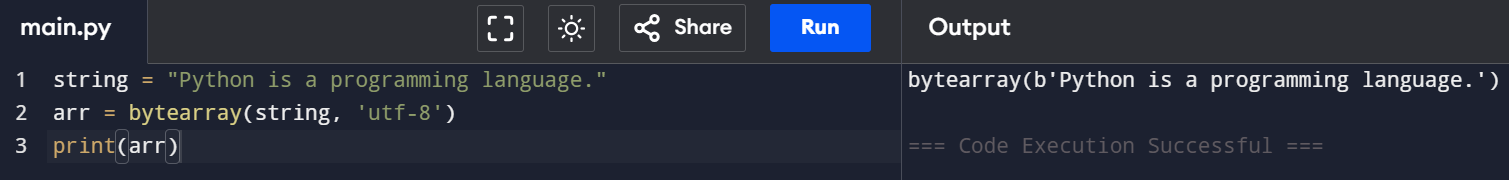
Python any() Function



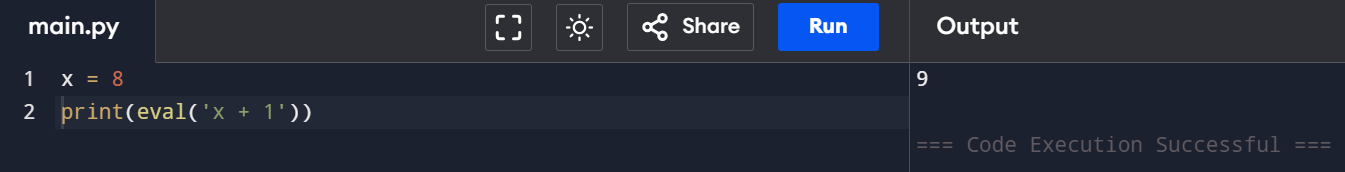
Python ascii() Function



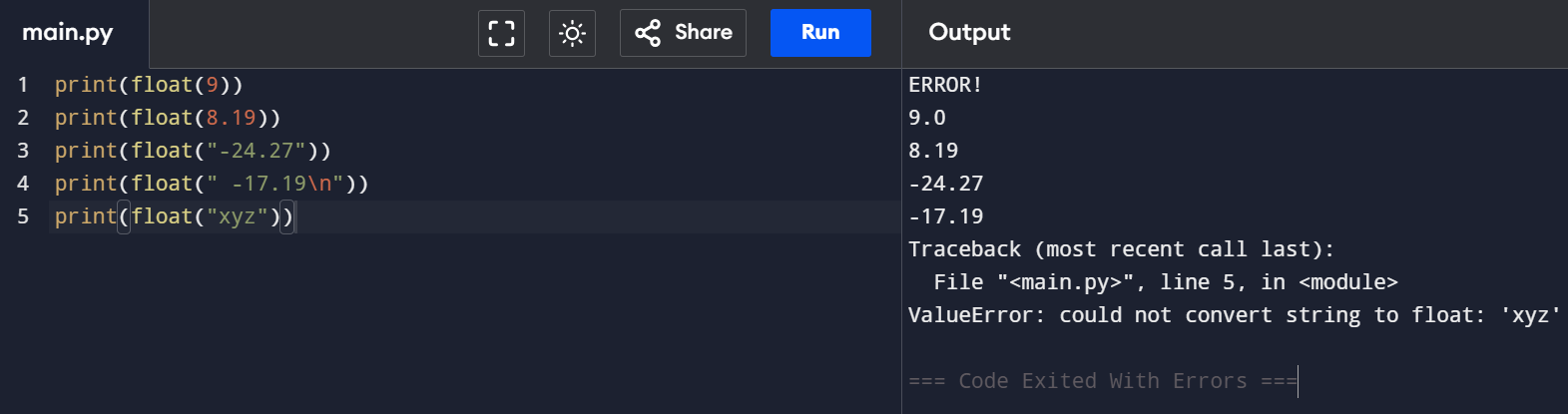
Python bytearray()



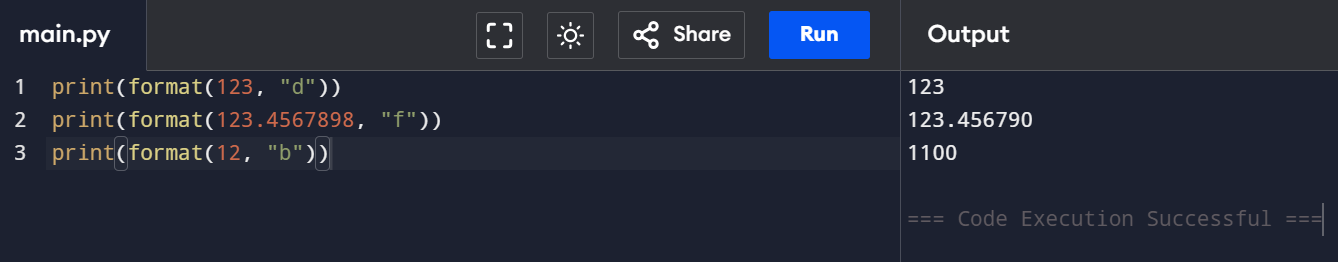
Python eval()



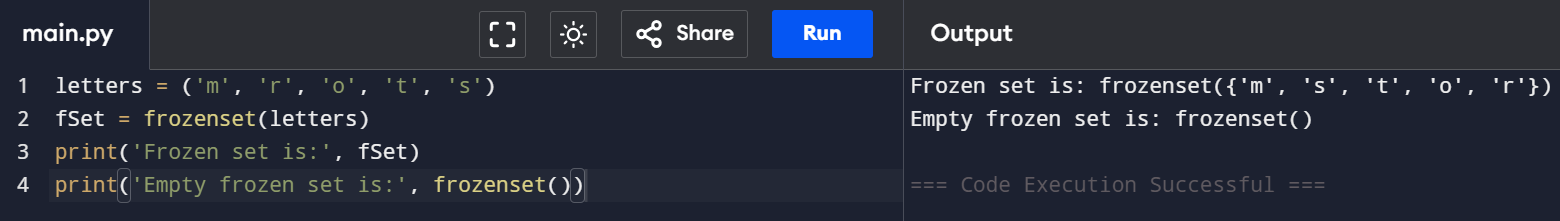
Python float()



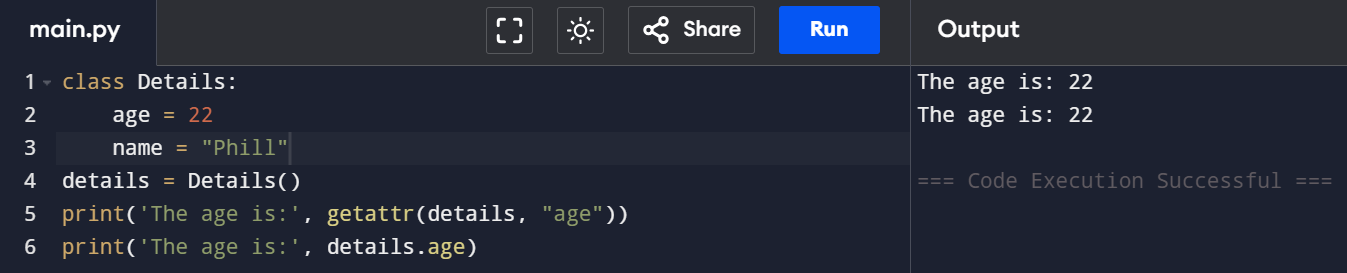
Python format() Function



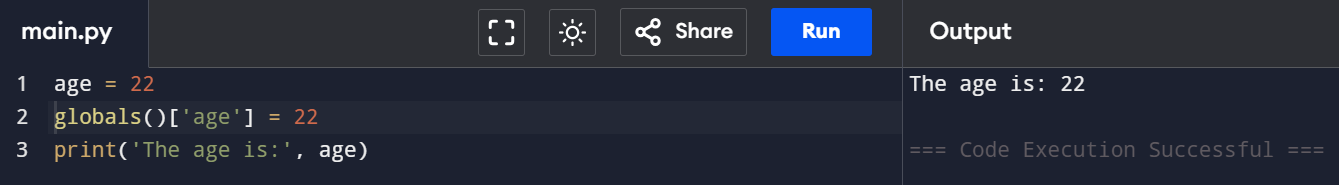
Python frozenset()



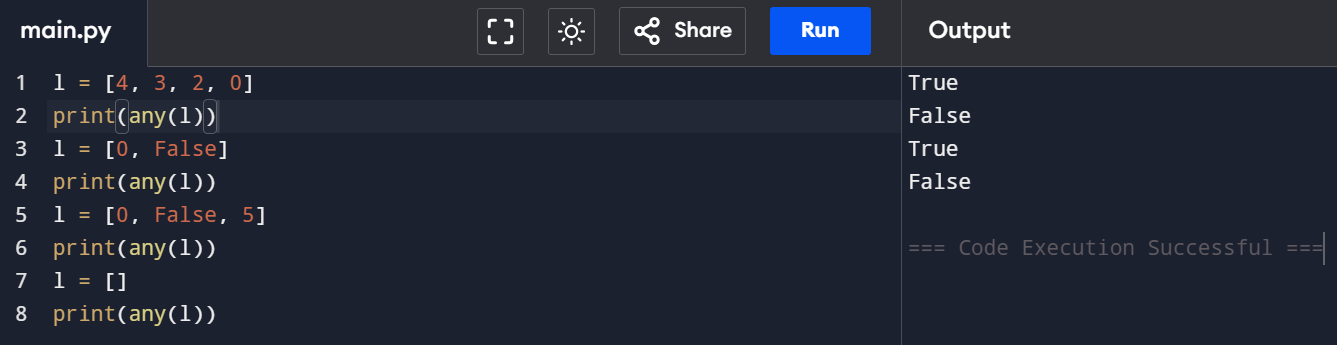
Python getattr() Function



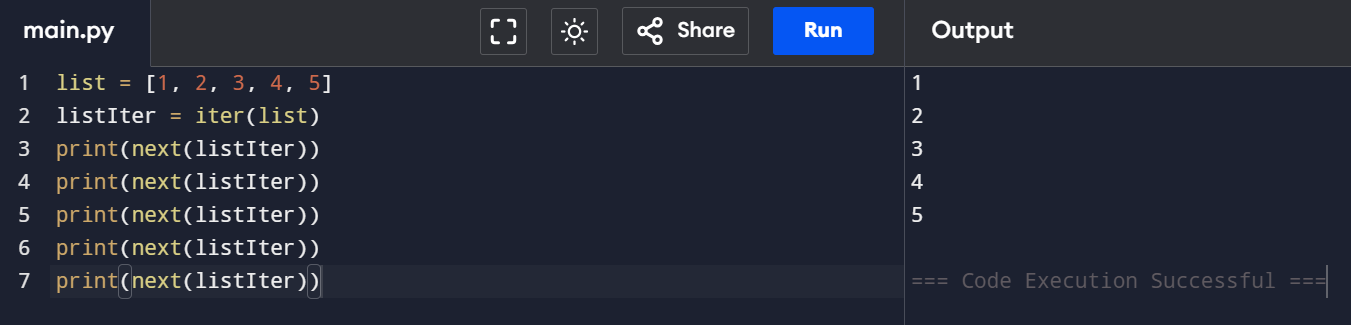
Python globals() Function



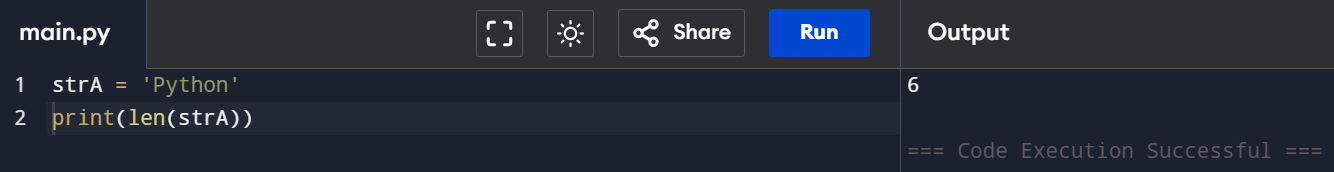
Python hasattr() Function



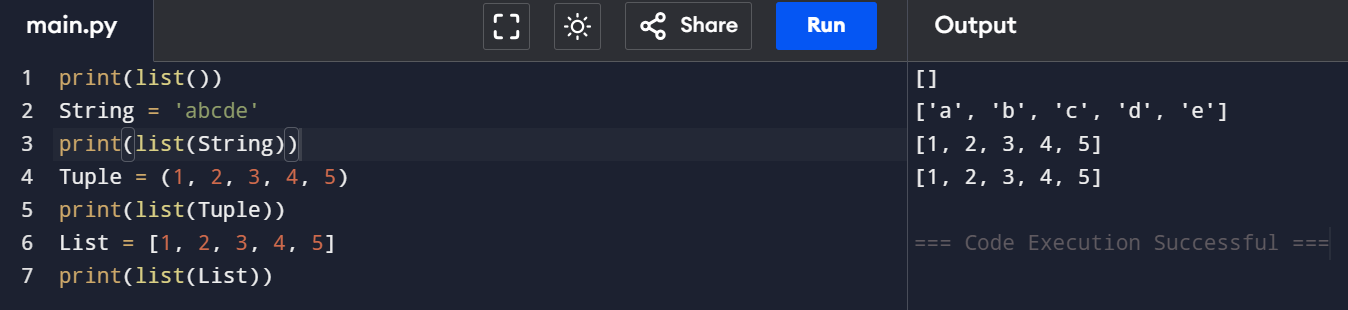
Python iter() Function



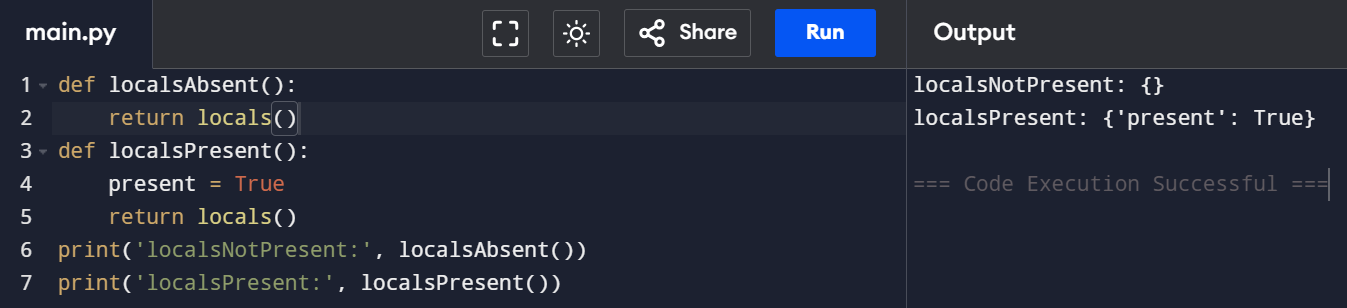
Python len() Function



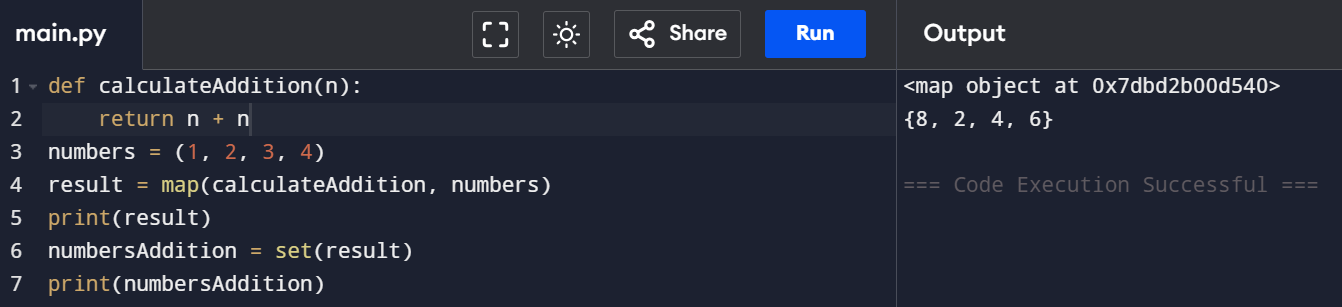
Python list()



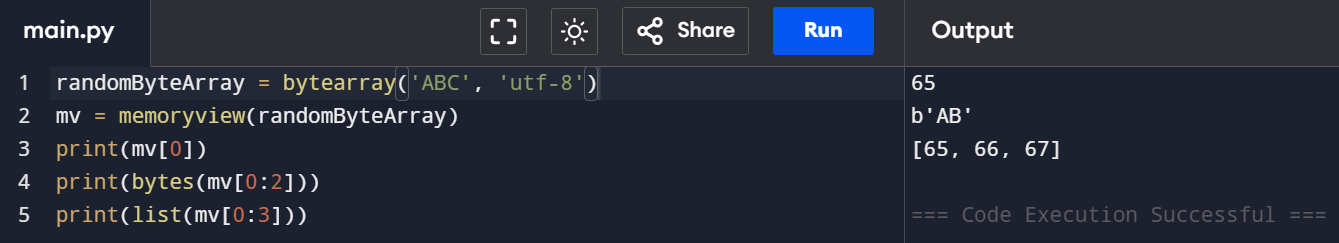
Python locals() Function



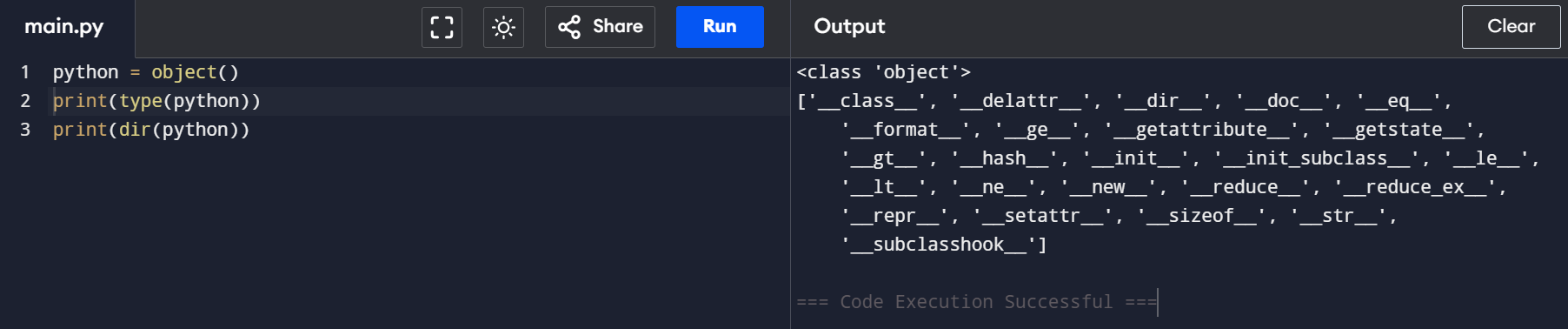
Python map() Function



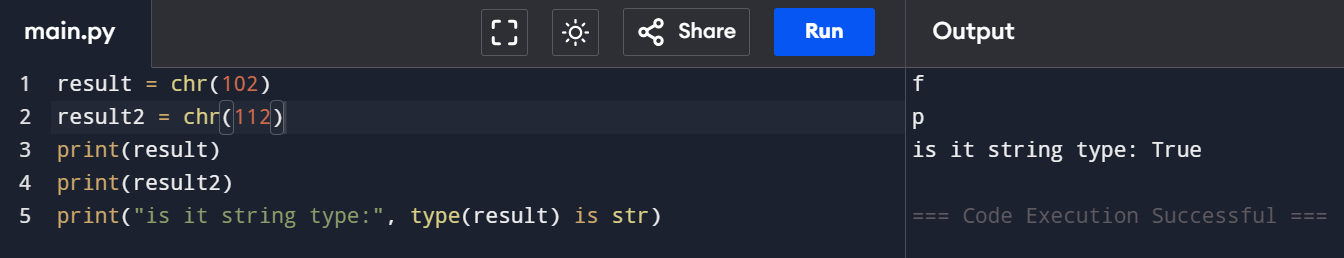
Python memoryview () Function



Python object()



Python chr() Function



Python complex() function

