1. Program to remove duplicates numbers entirely from the sorted array

Program :

def remove\_duplicates(arr):

count\_dict = {}

for num in arr:

if num in count\_dict:

count\_dict[num] += 1

else:

count\_dict[num] = 1

result = [num for num in arr if count\_dict[num] == 1

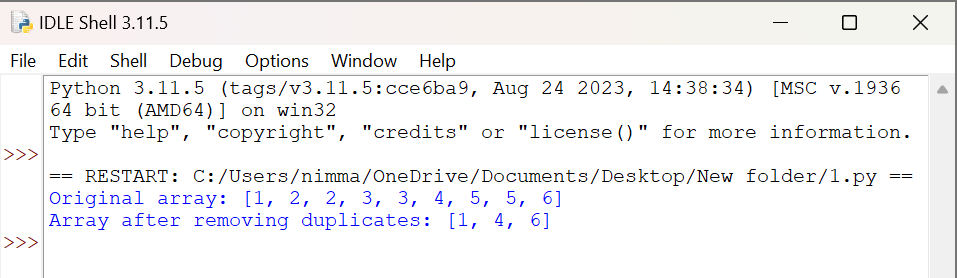
return result

sorted\_array = [1, 2, 2, 3, 3, 4, 5, 5, 6]

print("Original array:", sorted\_array)

print("Array after removing duplicates:", remove\_duplicates(sorted\_array))

Output :



1. Python program to create a list of all numbers in a range which are perfect squares and the sum of the digits of the number is less than 10

Program:

import math

def sum\_of\_digits(n):

return sum(int(digit) for digit in str(n))

def perfect\_squares\_with\_digit\_sum\_less\_than\_10(start, end):

start\_root = math.ceil(math.sqrt(start))

end\_root = math.floor(math.sqrt(end))

perfect\_squares = [

i \* i for i in range(start\_root, end\_root + 1)

if sum\_of\_digits(i \* i) < 10

]

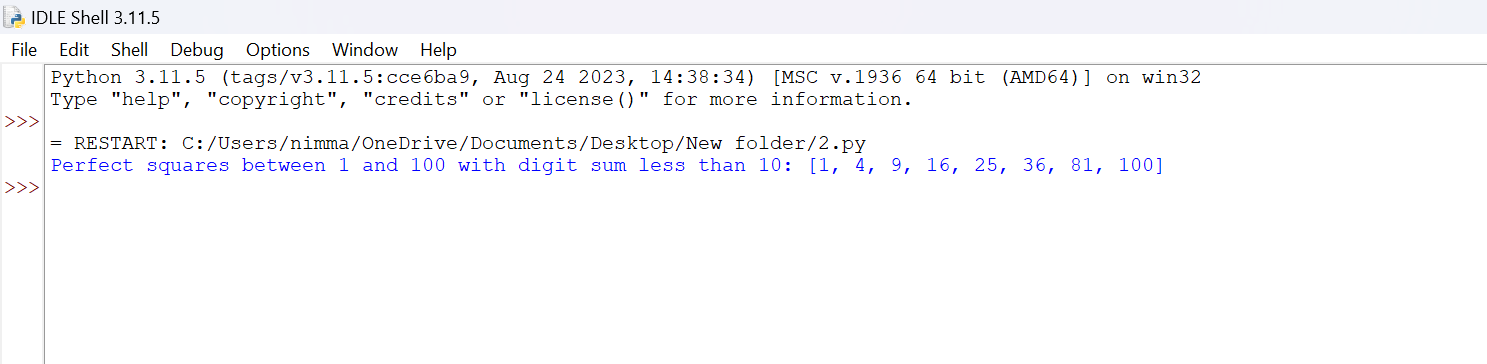
return perfect\_squares

start\_range = 1

end\_range = 100

print(f"Perfect squares between {start\_range} and {end\_range} with digit sum less than 10: {perfect\_squares\_with\_digit\_sum\_less\_than\_10(start\_range, end\_range)}")

Output:



1. Write a program to find the numbers of composite numbers in an array of elements

Program:

def is\_composite(n):

if n < 2:

return False

if n == 2 or n == 3:

return False

if n % 2 == 0 or n % 3 == 0:

return True

i = 5

while i \* i <= n:

if n % i == 0 or n % (i + 2) == 0:

return True

i += 6

return False

def count\_composite\_numbers(arr):

return sum(1 for num in arr if is\_composite(num))

array = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]

print(f"Number of composite numbers in the array: {count\_composite\_numbers(array)}")

Output:

