



## **LAB REPORT 03**

**SUBMITTED TO :** MR. MUBASHIR IQBAL

**SUBMITTED BY:** AHMED SALEEM RANA

**REG ID:** 24-CYS-023

**SECTION :** B

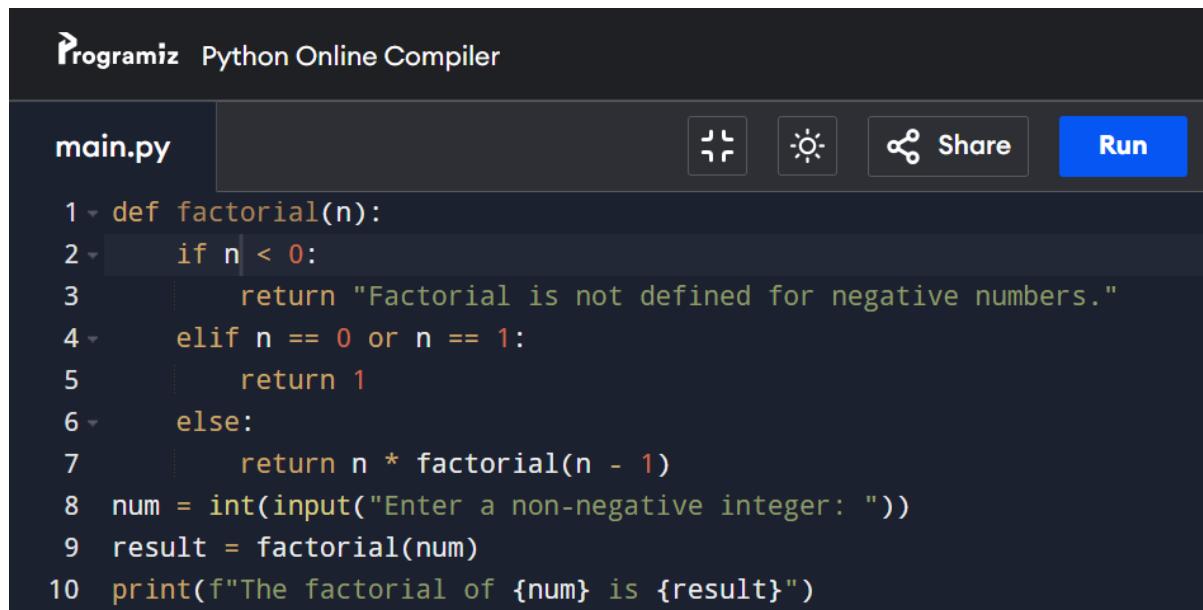
**COURSE TITLE:** ARTIFACIAL INTEELLIGENCE LAB

**HITEC UNIVERSITY, TAXILA CANTT**

## TASK 01

Write a Python function to calculate the factorial of a number (a non-negative integer). The function accepts the number as an argument.

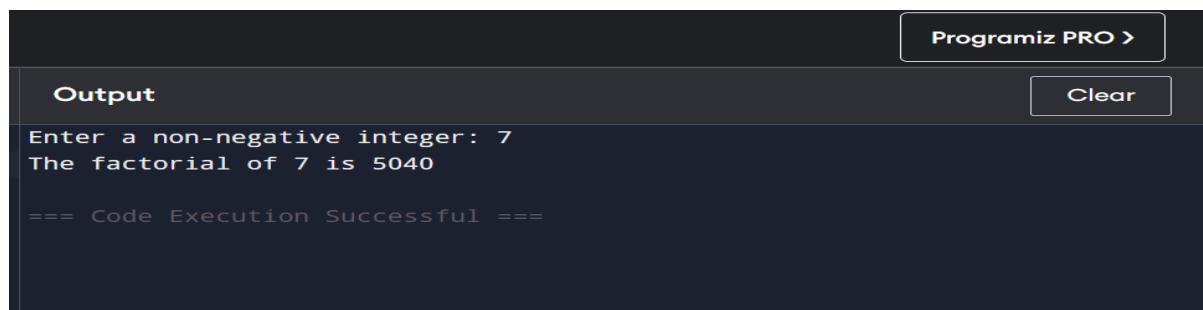
### INPUT:



The screenshot shows the Programiz Python Online Compiler interface. The code file is named 'main.py'. The code defines a factorial function that handles negative numbers, returns 1 for 0 or 1, and uses recursion for other values. It then prompts the user for a non-negative integer, calculates the factorial, and prints the result. The interface includes a 'Run' button and several status indicators.

```
1 def factorial(n):
2     if n < 0:
3         return "Factorial is not defined for negative numbers."
4     elif n == 0 or n == 1:
5         return 1
6     else:
7         return n * factorial(n - 1)
8 num = int(input("Enter a non-negative integer: "))
9 result = factorial(num)
10 print(f"The factorial of {num} is {result}")
```

### OUTPUT:



The screenshot shows the output window of the compiler. It displays the user input '7', the program's response 'The factorial of 7 is 5040', and a success message '==== Code Execution Successful ===='. There are buttons for 'Programiz PRO >' and 'Clear' at the top right.

Output

```
Enter a non-negative integer: 7
The factorial of 7 is 5040

==== Code Execution Successful ====
```

### EXPLANATION:

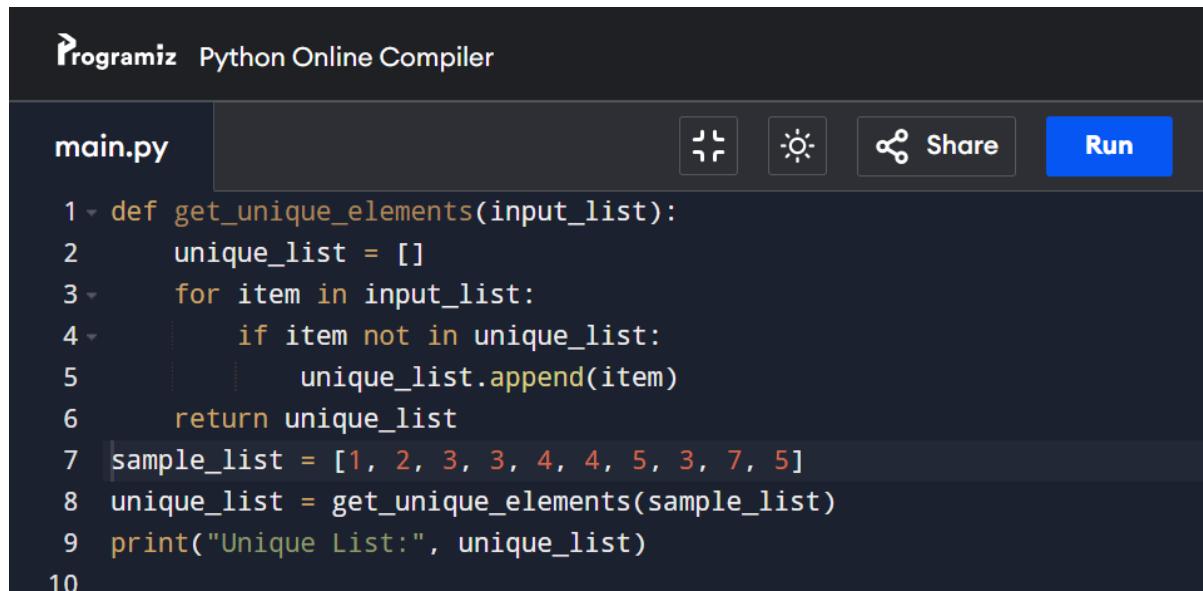
This program defines a function `factorial(n)` that calculates the factorial of a number using a loop. It multiplies all numbers from 1 to n and stores the result

in fact. The user enters a number, and the program prints its factorial using the function.

## TASK 02

Write a Python function that takes a list and returns a new list with unique elements of the first list. a) Sample List : [1,2,3,3,4,4,5,3,7,5] b) Unique List: [1, 2, 3, 4, 5, 7]

### INPUT:

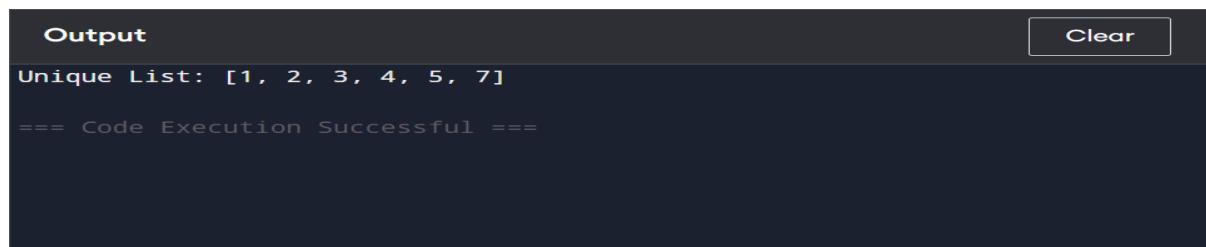


The screenshot shows the Programiz Python Online Compiler interface. The code editor contains the following Python script named main.py:

```
1 def get_unique_elements(input_list):
2     unique_list = []
3     for item in input_list:
4         if item not in unique_list:
5             unique_list.append(item)
6     return unique_list
7 sample_list = [1, 2, 3, 3, 4, 4, 5, 3, 7, 5]
8 unique_list = get_unique_elements(sample_list)
9 print("Unique List:", unique_list)
10
```

The interface includes a 'Run' button and other standard compiler controls.

### OUTPUT:



The screenshot shows the compiler's output window. It displays the printed output from the script:

```
Output
Clear
Unique List: [1, 2, 3, 4, 5, 7]
== Code Execution Successful ==
```

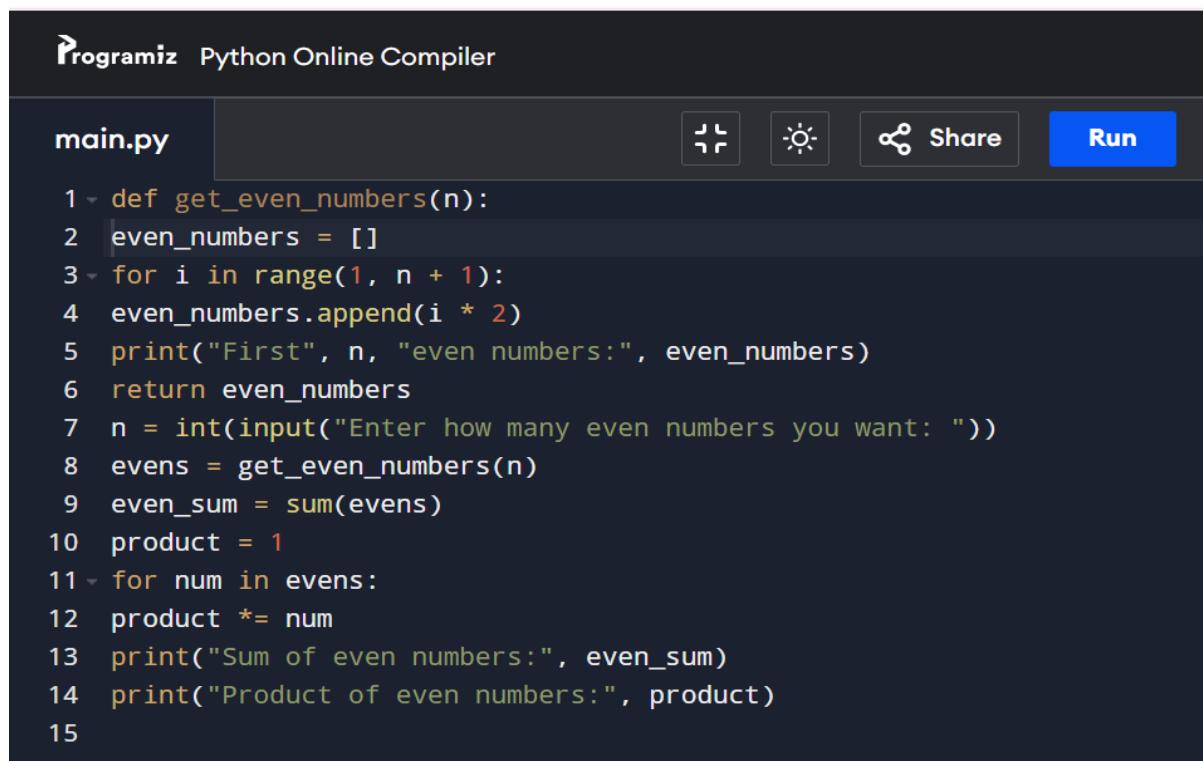
### EXPLANATION:

This program defines a function unique\_list(lst) that removes duplicate elements from a list. It converts the list into a set (which automatically removes duplicates) and then back into a list. The program then prints the unique elements from sample list.

### **TASK 03**

Write a Python function to show the first ‘n’ even numbers and then return the sum and product of that even numbers outside the function.

#### INPUT:



The screenshot shows the Programiz Python Online Compiler interface. The code editor contains the following Python script named main.py:

```
main.py
1 def get_even_numbers(n):
2     even_numbers = []
3     for i in range(1, n + 1):
4         even_numbers.append(i * 2)
5     print("First", n, "even numbers:", even_numbers)
6     return even_numbers
7
8 n = int(input("Enter how many even numbers you want: "))
9 evens = get_even_numbers(n)
10 even_sum = sum(evens)
11 product = 1
12 for num in evens:
13     product *= num
14 print("Sum of even numbers:", even_sum)
15 print("Product of even numbers:", product)
```

The interface includes a file tab for 'main.py', several configuration icons (full screen, brightness, share), and a blue 'Run' button.

## OUTPUT:

The screenshot shows a dark-themed terminal window titled "Output". At the top right are two buttons: "Programiz PRO >" and "Clear". The terminal displays the following text:  
Enter how many even numbers you want: 6  
First 6 even numbers: [2, 4, 6, 8, 10, 12]  
Sum of even numbers: 42  
Product of even numbers: 46080  
==== Code Execution Successful ====  
The text is white on a black background.

## EXPLANATION:

This program defines a function even\_numbers(n) that prints and returns the first n even numbers. It creates even numbers by multiplying each number from 1 to n by 2. The main program takes a number from the user, calls the function, and stores the even numbers in nums. Then it calculates and prints the sum of all even numbers using sum(nums) and the product using a loop

## **TASK 04**

Write a python program to match two string using functions.

## INPUT:

The screenshot shows the Programiz Python Online Compiler interface. The title bar says "Programiz Python Online Compiler". Below the title bar, there's a toolbar with icons for file operations, a "Share" button, and a "Run" button. The main area is a code editor with a dark theme. The code in the editor is as follows:

```
main.py
1 def match_strings(str1, str2):
2     if str1 == str2:
3         return "Strings match!"
4     else:
5         return "Strings do not match."
6
7 # Example usage
8 string1 = input("Enter the first string: ")
9 string2 = input("Enter the second string: ")
10
11 result = match_strings(string1, string2)
12 print(result)
```

## OUTPUT:

The screenshot shows a dark-themed online compiler interface. At the top right is a button labeled "Programiz PRO >". Below it is a "Output" section with a "Clear" button. The output text is:  
Enter the first string: 3,4,5,6  
Enter the second string: 7,8,9  
Strings do not match.  
==== Code Execution Successful ===

## EXPLANATION:

This program defines a function `match_strings(str1, str2)` that compares two strings. If both strings are the same, it prints "Strings match!", otherwise it prints "Strings do not match." The user is asked to enter two strings, and the function checks whether they are equal or not.

## TASK 05

Write a python program to name user login system using function.

### INPUT:

The screenshot shows a dark-themed online compiler interface. At the top left is the "Programiz Python Online Compiler" logo. The code editor window contains a file named "main.py" with the following content:

```
1 def login_system(stored_username):  
2     username = input("Enter your username: ")  
3     if username == stored_username:  
4         return "Login successful!"  
5     else:  
6         return "Login failed. Username not found."  
7     correct_username = "ahmed"  
8     result = login_system(correct_username)  
9     print(result)  
10    |
```

Below the code editor are several buttons: a copy icon, a clipboard icon, a share icon labeled "Share", and a blue "Run" button.

## OUTPUT:

The screenshot shows a dark-themed interface for a code editor or interpreter. At the top right is a button labeled "Programiz PRO >". Below it is a "Output" section with a "Clear" button. The output text area contains the following content:

```
Enter your username: ali
Login failed. Username not found.

==== Code Execution Successful ===
```

## EXPLANATION:

This program defines a function `login(username, password)` that checks if the entered username and password are correct. If the username is "admin" and the password is "1234", it prints "Login Successful!" Otherwise, it prints "Invalid Username or Password!" The user enters their login details, and the function verifies them.

## CONCLUSION:

These Python programs show the basic use of functions, loops, and conditions. The factorial program finds the product of numbers from 1 to n. The unique list program removes duplicate values from a list. The even numbers program displays the first n even numbers and finds their sum and product. The string matching program checks if two strings are the same, and the login program verifies user name and password.

## GITHUB REPOSITORY LINK: