

Python Introduction Lab

Assignment 1



Submitted by:

HARSHPREET SINGH

04013202717

CSE-1

Q1 Create a program that asks the user to enter their name and their age. Print out a message addressed to them that tells them the year that they will turn 100 years old.

Extras:

- Add on to the previous program by asking the user for another number and printing out that many copies of the previous message.
- Print out that many copies of the previous message on separate lines. (Hint: the string "\n" is the same as pressing the ENTER button)

Pseudocode

Code for printing the age of user after 100 years

Import datetime

Function definition of printname():

 name= Input for name

 age= Input for age

 number = input for number of times message is to be displayed

 age_100= age after 100 years

 print the age after 100 years

printing multiple copies

 if i less than number

 print the age after 100 years.

Call function printname()

Program Screenshot

```
In [1]: import datetime

def printName():
    name = input("Enter Your Name: \t")
    age = int(input("Enter You Age: \t"))
    number = int(input("Enter the Number of times you want to print Message: \t"))
    age_100 = (datetime.datetime.now().year) + (100-age)
    print("Hello! "+name+", and you will be of 100 years in "+ str(age_100)+"\n")

    print("Printing Multiple copies \n")
    for i in range(number):
        print("Hello! "+name+", and you will be of 100 years in "+ str(age_100)+"\n")

printName()
```

Output

```
Enter Your Name:      Harshpreet Singh
Enter You Age:  22
Enter the Number of times you want to print Message:  7
Hello! Harshpreet Singh, and you will be of 100 years in 2099

Printing Multiple copies

Hello! Harshpreet Singh, and you will be of 100 years in 2099
Hello! Harshpreet Singh, and you will be of 100 years in 2099
Hello! Harshpreet Singh, and you will be of 100 years in 2099
Hello! Harshpreet Singh, and you will be of 100 years in 2099
Hello! Harshpreet Singh, and you will be of 100 years in 2099
Hello! Harshpreet Singh, and you will be of 100 years in 2099
Hello! Harshpreet Singh, and you will be of 100 years in 2099
```

Github Link of Jupyter Notebook

[https://github.com/Harshpreets10/Machine-Learning-Lab/blob/main/Python%20Introduction%20Lab%20Assginement%201%20\(HARSHPREET%20SINGH%2C%2004013202717\).ipynb](https://github.com/Harshpreets10/Machine-Learning-Lab/blob/main/Python%20Introduction%20Lab%20Assginement%201%20(HARSHPREET%20SINGH%2C%2004013202717).ipynb)

Q2. Take a list, say for example this one:

```
a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
```

And write a program that prints out all the elements of the list that are less than 5.

Extras:

- Instead of printing the elements one by one, make a new list that has all the elements less than 5 from this list in it and print out this new list.
- Write this in one line of Python.
- Ask the user for a number and return a list that contains only elements from the original list that are smaller than that number given by the user.

Pseudocode

Code for printing out all the elements of the list which are less than 5

lst is the given list

Definition of function printelements(lst):

```
new_lst= empty list
for i in lst
    if i less than 5
        append element i to new_lst
print new_lst
```

Appending using one line

```
new_lst2= empty list  
enter elements in new_lst2 using list comprehension  
print new_lst2
```

Asking user for a number and returning a list that contains elements from the original list that are smaller than the number given by the user

number= input for the number given by user

new_lst3 = empty list

append elements which are less than number to new_lst3 using list comprehension

print elements of new_lst3

Program Screenshot

```
In [4]: def printelements(lst):  
    # Printing Elements Less than 5  
    new_lst = []  
    for i in lst:  
        if i<5:  
            print(i)  
            ##1. Appending in List  
            new_lst.append(i)  
    print("New List is {}".format(new_lst))  
  
    ##2. Appending in one line  
    new_lst2 = [x for x in lst if x<5]  
    print("New List in one line is{}".format(new_lst2))  
  
    ##3. Asking user and responding accordingly  
    number = int(input("Enter the number to find element smaller than the number: "))  
    new_lst3 = [x for x in lst if x<number]  
    print("Elements smaller than {} are {}".format(number, new_lst3))  
  
a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]  
printelements(a)
```

Output

```
1
1
2
3
New List is [1, 1, 2, 3]
New List in one line is[1, 1, 2, 3]
Enter the number to find element smaller than the number: 2
Elements smaller than 2 are [1, 1]
```

Github Link of Jupyter Notebook

[https://github.com/Harshpreets10/Machine-Learning-Lab/blob/main/Python%20Introduction%20Lab%20Assginment%201%20\(HARSHPREET%20SINGH%2C%2004013202717\).ipynb](https://github.com/Harshpreets10/Machine-Learning-Lab/blob/main/Python%20Introduction%20Lab%20Assginment%201%20(HARSHPREET%20SINGH%2C%2004013202717).ipynb)

Q3. Write a program that asks the user how many Fibonacci numbers to generate and then generates them. Take this opportunity to think about how you can use functions. Make sure to ask the user to enter the number of numbers in the sequence to generate.

(Hint: The Fibonacci sequence is a sequence of numbers where the next number in the sequence is the sum of the previous two numbers in the sequence. The sequence looks like this: 1, 1, 2, 3, 5, 8, 13 ...)

Pseudocode

Code for printing out a Fibonacci series till the number of terms entered by the user

num= number of terms to be displayed

Definition of function fibonacci(num)

If num less than equal to 1

 Return num

Else

 Return (Fibonacci(num-1) + (fibonacci(num-2)))

nterms= input for number of terms to be displayed

for i in range of nterms

 call fibonacci(i)

Program Screenshot

```
In [17]: def fibonacci(num):
    if num <= 1:
        return num
    else:
        return(fibonacci(num-1) + fibonacci(num-2))

nterms = int(input("Enter the number of numbers for fibonacci sequence: "))
print("Fibonacci sequence:")
for i in range(nterms):
    print(fibonacci(i))
```

Output

```
Enter the number of numbers for fibonacci sequence: 10
Fibonacci sequence:
0
1
1
2
3
5
8
13
21
34
```

GitHub Link of Jupyter Notebook

[https://github.com/Harshpreets10/Machine-Learning-Lab/blob/main/Python%20Introduction%20Lab%20Assginment%201%20\(HARSHPREET%20SINGH%20C%2004013202717\).ipynb](https://github.com/Harshpreets10/Machine-Learning-Lab/blob/main/Python%20Introduction%20Lab%20Assginment%201%20(HARSHPREET%20SINGH%20C%2004013202717).ipynb)

Q4. Write a program (function!) that takes a list and returns a new list that contains all the elements of the first list minus all the duplicates.

Extras:

- Write two different functions to do this - one using a loop and constructing a list, and another using set.

Pseudocode

Code for printing a list without any duplicated terms

lst= list given as input to the function

function to implement code using loop

Definition of function lst_unique(lst)

```
unique= empty list
for l in lst
    if l not in unique
        append element l to unique
    for x in unique
        print x
```

function to implement code using sets

Definition of function using_set(lst)

```
list_set= set made from list lst
unique_list = list made from set list_set
for x in unique_list
    print x
```

a = input list

```
call function lst_unique(a)  
call function using_set(a)
```

Program Screenshot

```
In [19]: def lst_unique(lst):  
    unique = []  
    for l in lst:  
        if l not in unique:  
            unique.append(l)  
    for x in unique:  
        print(x)  
  
def using_set(lst):  
    list_set = set(lst)  
    unique_list = list(list_set)  
    for x in unique_list:  
        print(x)  
  
a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 21, 89]  
print("Using Loop and Constructing a list")  
lst_unique(a)  
  
print("\nUsing Sets")  
using_set(a)
```

Output

Using Loop and Constructing a list

```
1  
2  
3  
5  
8  
13  
21  
34  
55  
89
```

Using Sets

```
1  
2  
3  
34  
5  
8  
13  
21  
55  
89
```

Github Link of Jupyter Notebook

[https://github.com/Harshpreets10/Machine-Learning-Lab/blob/main/Python%20Introduction%20Lab%20Assginment%201%20\(HARSHPREET%20SINGH%2C%2004013202717\).ipynb](https://github.com/Harshpreets10/Machine-Learning-Lab/blob/main/Python%20Introduction%20Lab%20Assginment%201%20(HARSHPREET%20SINGH%2C%2004013202717).ipynb)

Q5. Ask the user for a number and determine whether the number is prime or not. (For those who have forgotten, a prime number is a number that has no divisors.). Use functions

Pseudocode

Code for checking if given number is a prime number or not

Definition of function prime_check()

```
num= input for number to be checked
set flag =1
for i in range of 2 to (num divided by 2)
    if num%2 == 0
        set flag= 0
        break
    if flag == 0
        print number is not prime
    else
        print number is prime
```

Program Screenshot

```
In [27]: def prime_check():
    num = int(input("Enter Number you want to check \t"))
    flag = 1
    for i in range(2, int(num/2)):
        if(num%i == 0):
            flag = 0
            break
    if(flag==0):
        print("{} is not a Prime Number".format(num))
    else:
        print("{} is a Prime Number".format(num))
```

Output

```
In [28]: prime_check()  
Enter Number you want to check 12  
12 is not a Prime Number
```

```
In [29]: prime_check()  
Enter Number you want to check 67  
67 is a Prime Number
```

Github Link of Jupyter Notebook

[https://github.com/Harshpreets10/Machine-Learning-Lab/blob/main/Python%20Introduction%20Lab%20Assginment%201%20\(HARSHPREET%20SINGH%2C%2004013202717\).ipynb](https://github.com/Harshpreets10/Machine-Learning-Lab/blob/main/Python%20Introduction%20Lab%20Assginment%201%20(HARSHPREET%20SINGH%2C%2004013202717).ipynb)