Program - The current population of a town is 10000. The population of the town is increasing at the rate of 10% per year. You have to write a program to find out the population at the end of each of the last 10 years.

In [ ]:

```
# Code here
curr_pop = 10000

for i in range(10,0,-1):
  print(i,curr_pop)
  curr_pop = curr_pop/1.1
```

```
10 10000
9 9090.90909090909
8 8264.462809917353
7 7513.148009015775
6 6830.134553650703
5 6209.213230591548
4 5644.739300537771
3 5131.5811823070635
2 4665.07380209733
1 4240.976183724845
```

In [7]:

```
current_population = 10000
growth_rate = 10

for i in range(10,0,-1):
    current_population = current_population - ((10/100)*current_population)
    print(i, "years:", current_population)
```

```
10 years: 9000.0
9 years: 8100.0
8 years: 7290.0
7 years: 6561.0
6 years: 5904.9
5 years: 5314.41
4 years: 4782.969
3 years: 4304.6721
2 years: 3874.20489
1 years: 3486.784401
```

Sequence sum

1/1! + 2/2! + 3/3! + ...

In [9]:

```
#lets import math too for factorial, it will be easy
import math

num=int(input("Enter number, upto where you want to add the sequence: "))
sequencesum=0

for i in range(1,num+1):
    term=(i/math.factorial(i))
    sequencesum=sequencesum+term

print(sequencesum)
```

```
Enter number, upto where you want to add the sequence: 6
2.7166666666666663
```

Nested Loops

In [ ]:

```
# this is normal loop because, without ending inside loop, outside loop will not start & until completion of in
#with outer loop 1111, 2222 will not start.

for i in range(1,5):
  for j in range(1,5):
    print(i,j)
```

```
1 1
1 2
1 3
1 4
2 1
2 2
2 3
2 4
3 1
3 2
3 3
3 4
4 1
4 2
4 3
4 4
```

Pattern 1

*

**

***

In [10]:

```
# code here

#first loop provide row number
for i in range(1,11):
    for j in range(1,i+1):
        print("*", end=" ")
    print()
```

```
*
* *
* * *
* * * *
* * * * *
* * * * * *
* * * * * * *
* * * * * * * *
* * * * * * * * *
* * * * * * * * * *
```

In [11]:

```
# if we do not put last print, what will happen?
#first loop provide row number
for i in range(1,11):
    for j in range(1,i+1):
        print("*", end=" ")
```

```
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Loop Control Statement

- Break
- Continue
- Pass

In [ ]:

```
for i in range(1,10):
  if i == 5:
    break
  print(i)
```

```
1
2
3
4
```

In [ ]:

```
lower = int(input('enter lower range'))
upper = int(input('enter upper range'))

for i in range(lower,upper+1):
  for j in range(2,i):
    if i%j == 0:
      break
  else:
    print(i)
```

```
enter lower range10
enter upper range100
11
13
17
19
23
29
31
37
41
43
47
53
59
61
67
71
73
79
83
89
97
```

In [ ]:

```
# Continue
for i in range(1,10):
  if i == 5:
    continue
  print(i)
```

```
1
2
3
4
6
7
8
9
```

In [ ]:

```
for i in range(1,10):
  pass
```

Strings are sequence of Characters

In Python specifically, strings are a sequence of Unicode Characters

- Creating Strings
- Accessing Strings
- Adding Chars to Strings
- Editing Strings
- Deleting Strings
- Operations on Strings
- String Functions

Creating Stings

In [ ]:

```
s = 'hello'
s = "hello"
# multiline strings
s = '''hello'''
s = """hello"""
s = str('hello')
print(s)
```

```
hello
```

In [ ]:

```
"it's raining outside"
```

Out[ ]:

```
"it's raining outside"
```

Accessing Substrings from a String

In [ ]:

```
# Positive Indexing
s = 'hello world'
print(s[41])
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-61-633ba99ed6e5> in <module>
      1 # Positive Indexing
      2 s = 'hello world'
----> 3 print(s[41])

IndexError: string index out of range
```

In [ ]:
```
# Negative Indexing
s = 'hello world'
print(s[-3])
```

```
r
```

In [ ]:
```
# Slicing
s = 'hello world'
print(s[6:0:-2])
```

```
wol
```

In [ ]:
```
print(s[::-1])
```

```
dlrow olleh
```

In [ ]:
```
s = 'hello world'
print(s[-1:-6:-1])
```

```
dlrow
```

Editing and Deleting in Strings

In [ ]:
```
s = 'hello world'
s[0] = 'H'

# Python strings are immutable
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-80-0c8a824e3b73> in <module>
      1 s = 'hello world'
----> 2 s[0] = 'H'

TypeError: 'str' object does not support item assignment
```

In [ ]:
```
s = 'hello world'
del s
print(s)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-81-9ae37fbf1c6c> in <module>
      1 s = 'hello world'
      2 del s
----> 3 print(s)

NameError: name 's' is not defined
```

In [ ]:
```
s = 'hello world'
del s[-1:-5:2]
print(s)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-82-d0d823eafb6b> in <module>
      1 s = 'hello world'
----> 2 del s[-1:-5:2]
      3 print(s)

TypeError: 'str' object does not support item deletion
```

Operations on Strings

- Arithmetic Operations
- Relational Operations
- Logical Operations
- Loops on Strings
- Membership Operations

In [ ]:
```
print('delhi' + ' ' + 'mumbai')
```

```
delhi mumbai
```

In [ ]:
```
print('delhi'*5)
```

```
delhidelhidelhidelhidelhi
```

```
In [ ]: print("*"*50)
```

```
**************************************************
```

```
In [ ]: 'delhi' != 'delhi'

Out[ ]: False
```

```
In [ ]: 'mumbai' > 'pune'
        # lexiographically

Out[ ]: False
```

```
In [ ]: 'Pune' > 'pune'

Out[ ]: False
```

```
In [ ]: 'hello' and 'world'

Out[ ]: 'world'
```

```
In [ ]: 'hello' or 'world'

Out[ ]: 'hello'
```

```
In [ ]: '' and 'world'

Out[ ]: ''
```

```
In [ ]: '' or 'world'

Out[ ]: 'world'
```

```
In [ ]: 'hello' or 'world'

Out[ ]: 'hello'
```

```
In [ ]: 'hello' and 'world'

Out[ ]: 'world'
```

```
In [ ]: not 'hello'

Out[ ]: False
```

```
In [ ]: for i in 'hello':
          print(i)
```

```
h
e
l
l
o
```

```
In [ ]: for i in 'delhi':
          print('pune')
```

```
pune
pune
pune
pune
pune
```

```
In [ ]: 'D' in 'delhi'

Out[ ]: False
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

Common Functions

- len
- max
- min

- sorted

```
In [ ]: len('hello world')

Out[ ]: 11
```

```
In [ ]: max('hello world')

Out[ ]: 'w'
```

```
In [ ]: min('hello world')

Out[ ]: ' '
```

```
In [ ]: sorted('hello world',reverse=True)

Out[ ]: ['w', 'r', 'o', 'o', 'l', 'l', 'l', 'h', 'e', 'd', ' ']
```

```
In [ ]:
```

Capitalize/Title/Upper/Lower/Swapcase

```
In [ ]: s = 'hello world'
        print(s.capitalize())
        print(s)

        Hello world
        hello world
```

```
In [ ]: s.title()

Out[ ]: 'Hello World'
```

```
In [ ]: s.upper()

Out[ ]: 'HELLO WORLD'
```

```
In [ ]: 'Hello Wolrd'.lower()

Out[ ]: 'hello wolrd'
```

```
In [ ]: 'HeLlO WorLD'.swapcase()

Out[ ]: 'hElLo wORld'
```

Count/Find/Index

```
In [ ]: 'my name is nitish'.count('i')

Out[ ]: 3
```

```
In [ ]: 'my name is nitish'.find('x')

Out[ ]: -1
```

```
In [ ]: 'my name is nitish'.index('x')

---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-121-12e2ad5b75e9> in <module>
----> 1 'my name is nitish'.index('x')

ValueError: substring not found
```

```
In [ ]:
```

endswith/startswith

```
In [ ]: 'my name is nitish'.endswith('sho')

Out[ ]: False
```

```
In [ ]: 'my name is nitish'.startswith('1my')

Out[ ]: False
```

format

```
In [ ]: name = 'nitish'
        gender = 'male'

        'Hi my name is {1} and I am a {0}'.format(gender,name)

Out[ ]: 'Hi my name is nitish and I am a male'
```

isalnum/ isalpha/ isdigit/ isidentifier

```
In [ ]: 'nitish1234%'.isalnum()

Out[ ]: False
```

```
In [ ]: 'nitish'.isalpha()

Out[ ]: True
```

```
In [ ]: '123abc'.isdigit()

Out[ ]: False
```

```
In [ ]: 'first-name'.isidentifier()

Out[ ]: False
```

```
In [ ]:
```

Split/Join

```
In [ ]: 'hi my name is nitish'.split()

Out[ ]: ['hi', 'my', 'name', 'is', 'nitish']
```

```
In [ ]: " ".join(['hi', 'my', 'name', 'is', 'nitish'])

Out[ ]: 'hi my name is nitish'
```

Replace

```
In [ ]: 'hi my name is nitish'.replace('nitisrgewrhgh','campusx')

Out[ ]: 'hi my name is nitish'
```

Strip

```
In [ ]: 'nitish                           '.strip()

Out[ ]: 'nitish'
```

Example Programs

```
In [ ]: # Find the length of a given string without using the len() function

        s = input('enter the string')

        counter = 0

        for i in s:
          counter += 1

        print('length of string is',counter)

        enter the stringnitish
        length of string is 6
```

```
In [ ]: # Extract username from a given email.
        # Eg if the email is nitish24singh@gmail.com
        # then the username should be nitish24singh

        s = input('enter the email')

        pos = s.index('@')
```

```
print(s[0:pos])
```

```
enter the emailsupport@campusx.in
support
```

In [ ]:

```
# Count the frequency of a particular character in a provided string.
# Eg 'hello how are you' is the string, the frequency of h in this string is 2.

s = input('enter the email')
term = input('what would like to search for')

counter = 0
for i in s:
  if i == term:
    counter += 1

print('frequency',counter)
```

```
enter the emailhi how are you
what would like to search foro
frequency 2
```

In [ ]:

```
# Write a program which can remove a particular character from a string.
s = input('enter the string')
term = input('what would like to remove')

result = ''

for i in s:
  if i != term:
    result = result + i

print(result)
```

```
enter the stringnitish
what would like to removei
ntsh
```

In [ ]:

```
# Write a program that can check whether a given string is palindrome or not.
# abba
# malayalam

s = input('enter the string')
flag = True
for i in range(0,len(s)//2):
  if s[i] != s[len(s) - i -1]:
    flag = False
    print('Not a Palindrome')
    break

if flag:
  print('Palindrome')
```

```
enter the stringpython
Not a Palindrome
```

In [ ]:

```
# Write a program to count the number of words in a string without split()

s = input('enter the string')
L = []
temp = ''
for i in s:

  if i != ' ':
    temp = temp + i
  else:
    L.append(temp)
    temp = ''

L.append(temp)
print(L)
```

```
enter the stringhi how are you
['hi', 'how', 'are', 'you']
```

In [ ]:

```
# Write a python program to convert a string to title case without using the title()
s = input('enter the string')

L = []
for i in s.split():
  L.append(i[0].upper() + i[1:].lower())

print(" ".join(L))
```

```
enter the stringhi my namE iS NitiSh
Hi My Name Is Nitish
```

In [ ]:

```
# Write a program that can convert an integer to string.

number = int(input('enter the number'))
```

```
digits = '0123456789'
result = ''
while number != 0:
  result = digits[number % 10] + result
  number = number//10

print(result)
print(type(result))
```

```
enter the number345
345
<class 'str'>
```

In [ ]:

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js