```
Today Agenda
-----
- Iterations
- Strings,Methods
- List , Methods
- Tuple , Method
```

Iterations

- For Loop
- While

In [2]:

```
# print the Required table
num = int(input("Enter the Required table number"))
for i in range(1,11):
    print(num,"*",i,"=",num*i)
```

```
Enter the Required table number125
125 * 1 = 125
125 * 2 = 250
125 * 3 = 375
125 * 4 = 500
125 * 5 = 625
125 * 6 = 750
125 * 7 = 875
125 * 8 = 1000
125 * 9 = 1125
125 * 10 = 1250
```

In [3]:

```
# while
# reverse the number
Number = int(input("Please Enter any Number: "))
Reverse = 0   125 12
while(Number > 0):
    Reminder = Number %10.
    Reverse = (Reverse *10) + Reminder.
    Number = Number //10.
print("\n Reverse of entered number is = %d" %Reverse)
```

```
  File "<ipython-input-3-485c986b1a21>", line 4
    Reverse = 0   125 12
                  ^
SyntaxError: invalid syntax
```

In [4]:

```
While num>0:
    Res=num%10
    d = d*10+Res
    num=num//10
Print(Res)
```

```
  File "<ipython-input-4-a0b39cb555bb>", line 1
    While num>0:
          ^
SyntaxError: invalid syntax
```

In [5]:

```
N=0
Rem=n%10
N=(n*10)+rem
Num=num//10
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-5-cfa1dec32653> in <module>
      1 N=0
----> 2 Rem=n%10
      3 N=(n*10)+rem
      4 Num=num//10

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

In [7]:

```
n=125
print(n[::-1])
```

```
521
```

String

- Group of Characters
- String is immutable
- String assign in single or double or thrible quotations

In [8]:

```
s = 'String'
s1 = "String"
s2 = "'String'"
```

In [9]:

```
1 s1
```

Out[9]:

```
'String'
```

In [10]:

```
1 s2
```

Out[10]:

```
"'String'"
```

In [11]:

```
1 s
```

Out[11]:

```
'String'
```

In [12]:

```
1 s[0]
```

Out[12]:

```
'S'
```

In [13]:

```
1 s[0]='s'
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-13-a0518bb1a172> in <module>
----> 1 s[0]='s'

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

In [14]:

```
1 del s[1]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-14-062bd26529e9> in <module>
----> 1 del s[1]

TypeError: 'str' object doesn't support item deletion
```

In [15]:

```
str1 = "Ranganayakulu"
str1[0]
```

Out[15]:

```
'R'
```

In [16]:

```
str1[0:5]
```

Out[16]:

```
'Ranga'
```

In [17]:

```
str1[::2]
```

Out[17]:

```
'Rnaaauu'
```

In [18]:

```
str1[::-1]
```

Out[18]:

```
'ulukayanagnaR'
```

```
cap
toupper
count
find
split
tolower
lower
lstrip
rstrip
title
startswith
endswith
len()
join
strip
rfind
index
isalnum()
maketrans()
zfill()
isnumeric()
swapcase()
```

In [19]:

```
wor = "DataScience"
wor.capitalize()
```

Out[19]:

```
'Datascience'
```

In [20]:

```
"datascience".capitalize()
```

Out[20]:

```
'Datascience'
```

In [21]:

```
# it converts upper letters to lower letters
wor.casefold()
```

Out[21]:

```
'datascience'
```

In [24]:

```
wor.center(25)
```

Out[24]:

```
'       DataScience       '
```

In [25]:

```
wor.count('a')
```

Out[25]:

```
2
```

In [27]:

```
type(wor.encode())
```

Out[27]:

```
bytes
```

In [28]:

```
wor.endswith('a')
```

Out[28]:

```
False
```

In [29]:

```
1 wor.expandtabs()
```

Out[29]:

```
'DataScience'
```

In [30]:

```
1 p = "ranga\tsiva\t123"
2 p.expandtabs(1)
```

Out[30]:

```
'ranga siva 123'
```

In [31]:

```
1 p.expandtabs(5)
```

Out[31]:

```
'ranga     siva 123'
```

In [32]:

```
1 p.expandtabs()
```

Out[32]:

```
'ranga   siva    123'
```

In [33]:

```
1 wor.find('2')
```

Out[33]:

```
-1
```

In [34]:

```
1 wor
```

Out[34]:

```
'DataScience'
```

In [35]:

```
1 wor.find('D')
```

Out[35]:

```
0
```

In [36]:

```
# format
k = "Hii this is {name}, i am teaching {course}".format(name="Ranga",course="datascien
print(k)
```

```
Hii this is Ranga, i am teaching datascience
```

In [39]:

```
print("hii,my name is{}".format("harshini"))
```

```
hii,my name isharshini
```

In [40]:

```
str1 = "AndhraLoyola"
str1.index('d')
```

Out[40]:

```
2
```

In [41]:

```
str1.isalnum()
```

Out[41]:

```
True
```

In [42]:

```
str2="apssdc123"
str2.isalnum()
```

Out[42]:

```
True
```

In [43]:

```
str3 = "ap@sdc"
str3.isalnum()
```

Out[43]:

```
False
```

In [44]:

```
str1.isalpha()
```

Out[44]:

```
True
```

In [45]:

```
1 str2.isalpha()
```

Out[45]:

```
False
```

In [46]:

```
1 str2
```

Out[46]:

```
'apssdc123'
```

In [47]:

```
1 str1.isascii()
```

Out[47]:

```
True
```

In [48]:

```
1 str1.isidentifier()
```

Out[48]:

```
True
```

In [49]:

```
1 str1.islower()
```

Out[49]:

```
False
```

In [50]:

```
1 str1
```

Out[50]:

```
'AndhraLoyola'
```

In [51]:

```
1 str1.isupper()
```

Out[51]:

```
False
```

In [52]:

```
1 str1
```

Out[52]:

'AndhraLoyola'

In [54]:

```
1 p = "ANDHRA"
2 p.isupper()
```

Out[54]:

True

In [55]:

```
1 str1.isspace()
```

Out[55]:

False

In [56]:

```
1 d = " "
2 d.isspace()
```

Out[56]:

True

In [57]:

```
1 ["alekhya"," ","harsini","hemanth","madevi"," ","Anil"]
```

Out[57]:

['alekhya', ' ', 'harsini', 'hemanth', 'madevi', ' ', 'Anil']

In [58]:

```
1 d.istitle()
```

Out[58]:

False

In [59]:

```
1 p = "tile"
2 p.istitle()
```

Out[59]:

False

In [60]:

```
p = "Tile"
p.istitle()
```

Out[60]:

```
True
```

In [61]:

```
p = "TAIL"
p.istitle()
```

Out[61]:

```
False
```

In [64]:

```
str1.ljust(25)
```

Out[64]:

```
'AndhraLoyola               '
```

In [66]:

```
str1.partition('a')
```

Out[66]:

```
('Andhr', 'a', 'Loyola')
```

In [67]:

```
str1.rpartition('a')
```

Out[67]:

```
('AndhraLoyol', 'a', '')
```

In [68]:

```
str1.swapcase()
```

Out[68]:

```
'aNDHRAlOYOLA'
```

In [69]:

```
# Lower --> upper
# upper --> lower
str1.split('a')
```

Out[69]:

```
['Andhr', 'Loyol', '']
```

In [70]:

```
y = "       space           "
y.strip()
```

Out[70]:

```
'space'
```

In [71]:

```
y.rstrip()
```

Out[71]:

```
'       space'
```

In [72]:

```
y.lstrip()
```

Out[72]:

```
'space           '
```

List

- Mutable
- ordered Collection of elements
- Iterable
- Collection of similar types of data
- Different types of data
- indexing is possible
- Collection of elements,mutable,duplicate characters and elements,[]

List

- It is declare with []
- Seperates the values using ,(comma)

In [73]:

```
Stds_names= ['Lakshmi','Jagruthi','Sowjanya','Hemanth','Anil','Alekhya']
```

In [76]:

```
Stds_names[0]
```

Out[76]:

```
'Lakshmi'
```

In [77]:

```
1 Stds_names[0:3]
```

Out[77]:

```
['Lakshmi', 'Jagruthi', 'Sowjanya']
```

insert()
remove()
extend()
append()
reverse()
len()
sort()
clear()
copy()
index()
pop()
count()
sum(),max(),min()

length(),min(),max(),sort(),pop(),copy(),remove(),append(),extend(),insert(),count()sum(),index(),clear()

In [79]:

```
1 # insert
2 stds = ['Lakshmi','haasitha','supriya','alekhya','madevi','anil','ranga']
```

In [82]:

```
1 stds.insert(2,'tejasri')
```

In [83]:

```
1 stds
```

Out[83]:

```
['Lakshmi',
 'haasitha',
 'tejasri',
 'supriya',
 'alekhya',
 'madevi',
 'anil',
 'ranga']
```

In [84]:

```
1 stds.append('hema')
```

In [85]:

```
1 stds
```

Out[85]:

```
['Lakshmi',
 'haasitha',
 'tejasri',
 'supriya',
 'alekhya',
 'madevi',
 'anil',
 'ranga',
 'hema']
```

In [86]:

```
1 # extend
2 stds.extend()
```

In [87]:

```
1 stds
```

Out[87]:

```
['Lakshmi',
 'haasitha',
 'tejasri',
 'supriya',
 'alekhya',
 'madevi',
 'anil',
 'ranga',
 'hema',
 'S',
 'a',
 'i']
```

In [88]:

```
1 stds.append(['Gopi',"ranga","himavanth",'Kumar'])
```

In [89]:

```
1 stds
```

Out[89]:

```
['Lakshmi',
 'haasitha',
 'tejasri',
 'supriya',
 'alekhya',
 'madevi',
 'anil',
 'ranga',
 'hema',
 'S',
 'a',
 'i',
 ['Gopi', 'ranga', 'himavanth', 'Kumar']]
```

In [90]:

```
1 stds.extend(['Gopi',"ranga","himavanth",'Kumar'])
```

In [91]:

```
1 stds
```

Out[91]:

```
['Lakshmi',
 'haasitha',
 'tejasri',
 'supriya',
 'alekhya',
 'madevi',
 'anil',
 'ranga',
 'hema',
 'S',
 'a',
 'i',
 ['Gopi', 'ranga', 'himavanth', 'Kumar'],
 'Gopi',
 'ranga',
 'himavanth',
 'Kumar']
```

In [92]:

```
1 stds.remove(['Gopi', 'ranga', 'himavanth', 'Kumar'])
```

In [93]:

```
1 stds
```

Out[93]:

```
['Lakshmi',
 'haasitha',
 'tejasri',
 'supriya',
 'alekhya',
 'madevi',
 'anil',
 'ranga',
 'hema',
 'S',
 'a',
 'i',
 'Gopi',
 'ranga',
 'himavanth',
 'Kumar']
```

In [94]:

```
1 stds.pop()
```

Out[94]:

```
'Kumar'
```

In [95]:

```
1 stds.pop(8)
```

Out[95]:

```
'hema'
```

In [96]:

```
1 stds
```

Out[96]:

```
['Lakshmi',
 'haasitha',
 'tejasri',
 'supriya',
 'alekhya',
 'madevi',
 'anil',
 'ranga',
 'S',
 'a',
 'i',
 'Gopi',
 'ranga',
 'himavanth']
```

In [97]:

```
1 stds.count('ranga')
```

Out[97]:

2

In [98]:

```
1 std2 = stds.copy()
```

In [99]:

```
1 std2
```

Out[99]:

```
['Lakshmi',
 'haasitha',
 'tejasri',
 'supriya',
 'alekhya',
 'madevi',
 'anil',
 'ranga',
 'S',
 'a',
 'i',
 'Gopi',
 'ranga',
 'himavanth']
```

In [100]:

```
1 stds
```

Out[100]:

```
['Lakshmi',
 'haasitha',
 'tejasri',
 'supriya',
 'alekhya',
 'madevi',
 'anil',
 'ranga',
 'S',
 'a',
 'i',
 'Gopi',
 'ranga',
 'himavanth']
```

In [101]:

```
1 stds.reverse()
```

In [102]:

```
1 stds
```

Out[102]:

```
['himavanth',
 'ranga',
 'Gopi',
 'i',
 'a',
 'S',
 'ranga',
 'anil',
 'madevi',
 'alekhya',
 'supriya',
 'tejasri',
 'haasitha',
 'Lakshmi']
```

In [103]:

```
1 stds.sort()
```

In [104]:

```
1 stds
```

Out[104]:

```
['Gopi',
 'Lakshmi',
 'S',
 'a',
 'alekhya',
 'anil',
 'haasitha',
 'himavanth',
 'i',
 'madevi',
 'ranga',
 'ranga',
 'supriya',
 'tejasri']
```

max() -->maximum min()-->minimum value

tuple

- It is immutable
- It is ordeded
- it is declare with ()

In [105]:

```
1 t1 = (1,8,6,4,2)
```

In [106]:

```
1 t1
```

Out[106]:

```
(1, 8, 6, 4, 2)
```

In [107]:

```
1 t1[0]
```

Out[107]:

```
1
```

In [108]:

```
1 t1[0:]
```

Out[108]:

```
(1, 8, 6, 4, 2)
```

In [109]:

```
1 t1.count('5')
```

Out[109]:

```
0
```

In [110]:

```
1 t1.index(5)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-110-82da154b8472> in <module>
----> 1 t1.index(5)

ValueError: tuple.index(x): x not in tuple
```

In [111]:

```
1 t1.index(1)
```

Out[111]:

```
0
```

1. take the string and print the ascii values in list
2. Take paragraph and after space string must be in capital letter

3. find the number of duplicates in the string
4. take one list with stds names
    - Starting char must be capital letter
    - print only last char as "A"

In [ ]:

```
1
```