

Data Science & AI & NIC - Param

Python-For Data Science
List

Lecture No.- 01

By- Pankaj Sharma Sir



Recap of Previous Lecture



Topic

Strings Part-02



isalnum()
isalpha()
isdigit()
isspace()

upper()
lower()
swapcase()

Topics to be Covered



Topic

List Part-01



List

- * duplicate allowed ✓
- * Order preserve ✓
- * Slicing
- * index
- * mutable
- * Heterogeneous

create

- ① `l = []` # Empty list
- ② `l = [1, 2, 3, 4, 5]`
- ③ `l = []`
`for i in range(1, 6):`
`l.append(i)`
- ④ `l = list(range(1, 6))`
- ⑤ `l = eval(input("Enter a list"))`

String

`split` \Rightarrow return a list

`Eval` ✓
`append` ✓

Problem solving

space separated i/p

①

1 2 3 4 5

input() \Rightarrow "1 2 3 4 5"

s = input().split() \Rightarrow ["1", "2", "3", "4", "5"]

↓
string
↓
int

l = []

for element in s:

l.append(int(element))

[1 2 3 4 5] print(l)

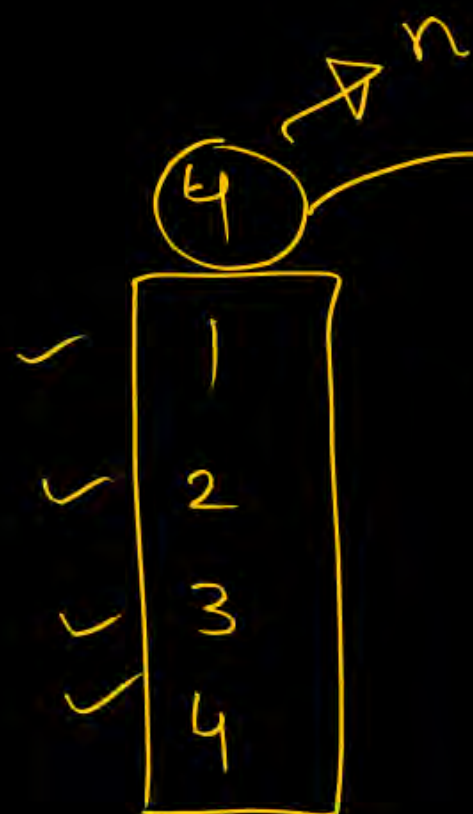
②

4 → no. of elements

1 2 3 4 →

```
n = int(input())
```


③



$n \rightarrow$ 4

$l = []$
 $n = \text{int}(\text{input}())$

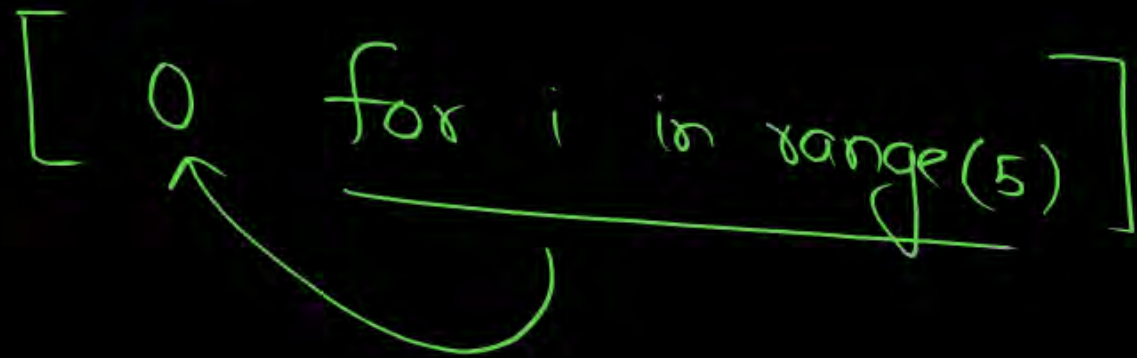
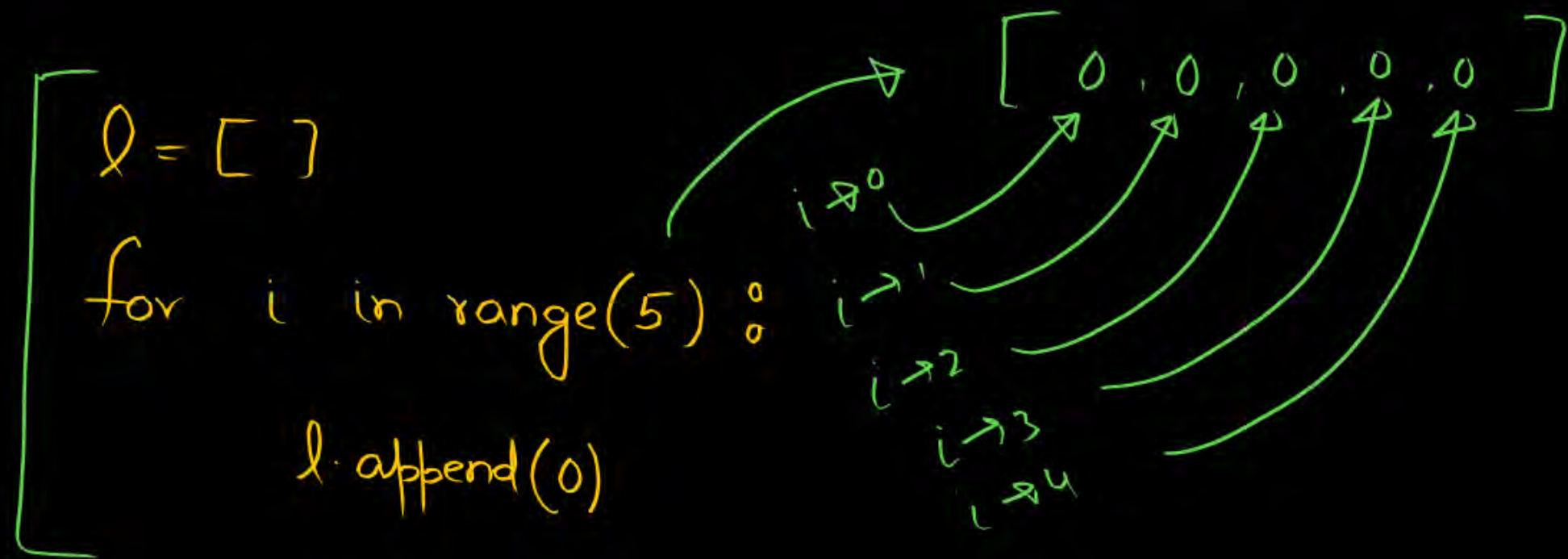
for i in $\text{range}(n)$:

$\text{inp} = \text{int}(\text{input}())$
 $l.append(\text{inp})$

$\text{print}(l)$

$\text{int}('1')$
 $\Rightarrow 1$

'1'



adding element

- ① `append(x)` : add element x at the end of the list
 - ② `insert(i, element)` : insert element at index i
 - ③ `extend`
-] \Rightarrow { only element }



To add all elements of a list
into another list.

$a = [1, 2, 3]$

$b = [4, 5, 6]$

`a.extend(b)`

delete

not
return anything

① remove

② pop → a)
→ b)

remove → any element

Pop(i) → index
any ele. ✓

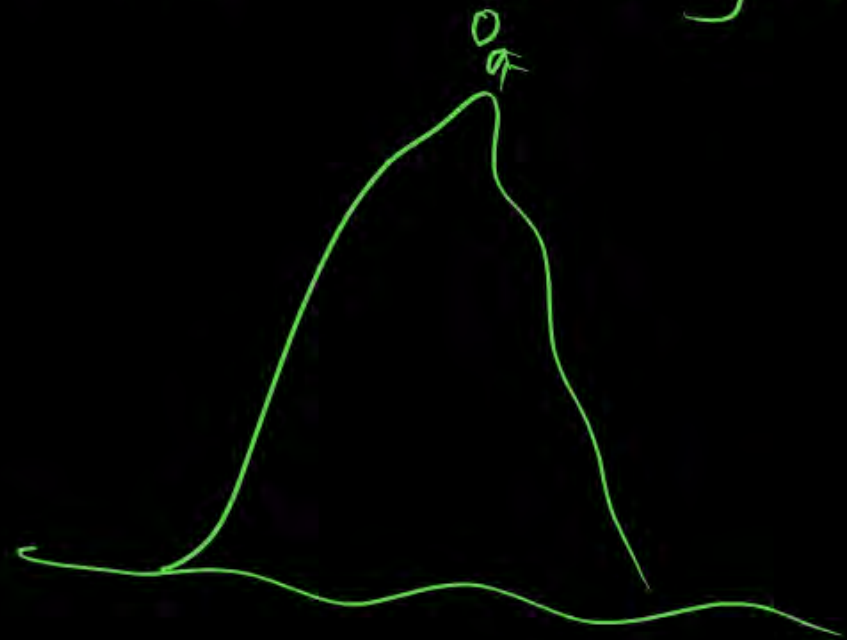
return

Pop() → last element

{ append()
Pop() } stack

Pop() \rightarrow ^{only} last elem

Pop(i) \rightarrow any element



① reverse()

② sort()

①

l = [1, 2, 3, 4]

l.reverse()

print(l) [4, 3, 2, 1]

l = ["Bam", "aam", "shyam"]

l.sort()

print(l)

['aam', 'Bam', 'shyam']

l = [10, 0, 50, -90]

l.sort()

print(l)

[-90, 0, 10, 50]

Day 12 Python List

```
In [1]: #creating a list
#Empty list
l=[]
print(l)
print(type(l))

[]
<class 'list'>
```

```
In [2]: #using List()
l=list(range(1,6))
print(l)

[1, 2, 3, 4, 5]
```

```
In [4]: #list()
l=list(range(1,6,2)) #==>1,3,5
print(l)

[1, 3, 5]
```

```
In [6]: #eval
l=eval(input("Enter a list"))
print(l)

Enter a list[1,2,3,4]
[1, 2, 3, 4]
```

```
In [7]: #using a for loop
l=[]
for i in range(1,6):
    l.append(i)    #add i to the end of list l
print(l)

[1, 2, 3, 4, 5]
```

```
In [8]: l=[1,2,"pankaj"]
print(l)

[1, 2, 'pankaj']
```

```
In [9]: #taking a space seperated seq of integers from user
s=input().split()
l=[]
for element in s:
    l.append(int(element))
print(l)

1 2 3 4 5
[1, 2, 3, 4, 5]
```

```
In [12]: n=int(input())
l=[]
for i in range(n):
    inp=int(input())
```

```
l.append(inp)
print(l)
```

```
6
10
20
30
40
50
60
[10, 20, 30, 40, 50, 60]
```

```
In [13]: #list comprehension
l=[ 0 for i in range(5)]
print(l)
```

```
[0, 0, 0, 0, 0]
```

```
In [14]: l=[ i for i in range(5) ] #0,1,2,3,4
print(l)
```

```
[0, 1, 2, 3, 4]
```

```
In [15]: l=[ 2*i for i in range(5)] #[0,2,4,6,8]
print(l)
```

```
[0, 2, 4, 6, 8]
```

```
In [16]: l=[i*i for i in range(10)]#0,1,4,9,16,25,36,49,64,81
print(l)
```

```
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
```

```
In [17]: #adding elements in list
#append ==>insert element at end of the list
l=[]
l.append(100)
print(l)
```

```
[100]
```

```
In [18]: l.append(200)
print(l)
```

```
[100, 200]
```

```
In [19]: l.append("pankaj")
print(l)
```

```
[100, 200, 'pankaj']
```

```
In [20]: #insert
l
```

```
Out[20]: [100, 200, 'pankaj']
```

```
In [24]: l.insert(0,"neeraj")#insert "neeraj" at index 0 ==>starting of the list
# 'neeraj' will be 1st element of the list
print(l)
print(id(l))
```

```
['neeraj', 'neeraj', 100, 9999, 200, 'pankaj', 'dheeraj']  
1928771787200
```

```
In [25]: l.insert(2,9999)#3rd element will be 9999  
#['neeraj',100,9999,200,'pankaj']  
print(l)  
print(id(l))
```

```
['neeraj', 'neeraj', 9999, 100, 9999, 200, 'pankaj', 'dheeraj']  
1928771787200
```

```
In [26]: l.insert(50,"dheeraj")#50>largest index possible  
print(l)  
print(id(l))
```

```
['neeraj', 'neeraj', 9999, 100, 9999, 200, 'pankaj', 'dheeraj', 'dheeraj']  
1928771787200
```

```
In [29]: #extend ==>  
a=[1,2,3]  
b=[4,5,6]  
a.extend(b)  
print(a)
```

```
[1, 2, 3, 4, 5, 6]
```

```
In [30]: l=[1,2,3,4,1,6,4,4]  
l.remove(2)#to remove specified element from the list  
print(l)
```

```
[1, 3, 4, 1, 6, 4, 4]
```

```
In [31]: l.remove(4)#remove only 1st occurrence of 4  
print(l)
```

```
[1, 3, 1, 6, 4, 4]
```

```
In [32]: l.remove(100)#100 is not present in the list
```

```
-----  
ValueError                                Traceback (most recent call last)  
Cell In[32], line 1  
----> 1 l.remove(100)  
  
ValueError: list.remove(x): x not in list
```

```
In [33]: #pop() ==>it remove and returns the last element from the list  
a=[1,2,3,4]  
l=a.remove(1)#remove does not returns anything  
print(l) #None
```

```
None
```

```
In [35]: a=[1,2,3,4]  
print("Original list is",a)  
l=a.pop()#it remove and return last element 4 from the list  
  
print("Element removed is",l)  
print("new list is ",a)
```


Original list is [1, 2, 3, 4]
 Element removed is 4
 new list is [1, 2, 3]

In [36]: `a=[]`
`a.pop()`

```
-----
IndexError                                Traceback (most recent call last)
Cell In[36], line 2
      1 a=[]
----> 2 a.pop()

IndexError: pop from empty list
```

In [38]: `a=[10,20,30,40]`
`print("Original list is ",a)`
#pop() can also be used to remove element at particular index
#pop(i) ==> it remove and return element at index i from the list
`l=a.pop(2) #index 2==>3rd element`
`print("the element removed is",l)`
`print("New list is ",a)`

Original list is [10, 20, 30, 40]
 the element removed is 30
 New list is [10, 20, 40]

In [39]: `a=[1,2,3,4]`
`print(a)`

[1, 2, 3, 4]

In [40]: `del a`

In [41]: `a`

```
-----
NameError                                Traceback (most recent call last)
Cell In[41], line 1
----> 1 a

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

In [42]: `a=[1,2,3,4]`
`a=None`

In [43]: `a`

In [44]: `a=[2,3]`

In [45]: `a`

Out[45]: [2, 3]

In [46]: `l=[1,2,3,4,"pankaj"]`
`print(len(l))`

5

```
In [47]: print(l.index("pankaj"))
```

```
4
```

```
In [48]: print(l.index("neeraj"))
```

```
-----  
ValueError                                Traceback (most recent call last)  
Cell In[48], line 1  
----> 1 print(l.index("neeraj"))  
  
ValueError: 'neeraj' is not in list
```

```
In [49]: print(l)
```

```
[1, 2, 3, 4, 'pankaj']
```

```
In [50]: #membership  
if "neeraj" in l:  
    i=l.index("neeraj")  
    print("Element is at position",i)  
else :  
    print("element is not present")
```

```
element is not present
```

```
In [51]: a=[1,2,3,4]  
a.reverse()  
print(a)
```

```
[4, 3, 2, 1]
```

```
In [52]: a=[10,"pankaj",2,"neeraj"]  
a.reverse()  
print(a)
```

```
['neeraj', 2, 'pankaj', 10]
```

```
In [53]: a=[1,2,30,-2,900,-1]  
a.sort()  
print(a)
```

```
[-2, -1, 1, 2, 30, 900]
```

```
In [54]: a=[1,2,"pankaj"] #element are not of same type  
a.sort()
```

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[54], line 2  
      1 a=[1,2,"pankaj"] #element are not of same type  
----> 2 a.sort()  
  
TypeError: '<' not supported between instances of 'str' and 'int'
```

```
In [55]: a=[1,2,3,-3,0,-900]  
a.sort(reverse=True)  
print(a)
```

```
[3, 2, 1, 0, -3, -900]
```

```
In [ ]: #next class : List cloning and aliasing
```


THANK - YOU