






Python Lists


Python list ek ordered aur mutable collection hoti hai jo multiple items ko ek variable mein store karne ke liye use hoti hai. Yeh square brackets [] ke andar define hoti hai aur isme different types ke data store kiye ja sakte hain.

-  Ordered hoti hai (sequence maintain hota hai)
-  Mutable hoti hai (modify kar sakte hain)
-  Duplicate values allow karti hai
-  Different data types store kar sakti hai
- Syntax dekh niche list ka:

```
my_list = [1, 2, 3, "hello", 5.5]
print(my_list)
```

-  `append()` → List ke end me ek element add karta hai


```
my_list = [1, 2, 3]
my_list.append(4)
print(my_list) # Output: [1, 2, 3, 4]
```

-  `extend()` → Ek list ke andar doosri list ke elements add karta hai

```
list1 = [1, 2, 3]
list2 = [4, 5, 6]
list1.extend(list2)
print(list1) # Output: [1, 2, 3, 4, 5, 6]
```

-  `insert()` → Specific index pe element insert karta hai

```
my_list = [1, 2, 4]
my_list.insert(2, 3) # Index 2 pe '3' insert karega
print(my_list) # Output: [1, 2, 3, 4]
```

-  `remove()` → Diye gaye element ko list se hata deta hai

```
my_list = [1, 2, 3, 4]
my_list.remove(2)
print(my_list) # Output: [1, 3, 4]
```

-  `pop()` → Last element ya specific index ka element remove karta hai

```
my_list = [1, 2, 3, 4]
my_list.pop()
print(my_list) # Output: [1, 2, 3]
```

- **6** index() → Diye gaye element ka index return karta hai

```
my_list = [10, 20, 30, 40]
print(my_list.index(30)) # Output: 2
```

- **7** count() → Kisi element ki list me kitni baar occurrence hui, yeh count karta hai

```
my_list = [1, 2, 2, 3, 2, 4]
print(my_list.count(2)) # Output: 3
```

- **8** sort() → List ko ascending order me sort karta hai

```
my_list = [4, 1, 3, 2]
my_list.sort()
print(my_list) # Output: [1, 2, 3, 4]
```

- **9** reverse() → List ke elements ka order ulta kar deta hai

```
my_list = [1, 2, 3, 4]
my_list.reverse()
print(my_list) # Output: [4, 3, 2, 1]
```

- **10** copy() → List ki ek nayi copy return karta hai

```
my_list = [1, 2, 3]
new_list = my_list.copy()
print(new_list) # Output: [1, 2, 3]
```

- **11** clear() → List ke saare elements hata deta hai

```
my_list = [1, 2, 3]
my_list.clear()
print(my_list) # Output: []
```