

## لیست (List) در پایتون

در برنامه‌نویسی پایتون، نوع داده لیست با قرار دادن همه آیتم‌ها (عناصر) درون یک براکت یعنی [ ] ساخته می‌شود. عناصر یک لیست، با استفاده از علامت ویرگول، از هم جدا می‌شوند. لیست می‌تواند هر تعدادی عنصر داشته باشد و این عناصر ممکن است خود دارای انواع داده متفاوتی باشند. برای مثال، عناصر یک لیست ممکن است ترکیبی از نوع داده «صحیح» (integer)، «شناور» (float) و «رشته» (String) باشند. نوع یا type لیست‌ها از جنس list است.

```
mylist = ["ali", "hassan", 1, 2, True, "reza"]
```

لیست‌ها قابل تغییر هستند و همچنین مجاز به داشتن مقادیر تکراری می‌باشند. از دیگر صفات لیست‌ها مرتب بودن آنهاست یعنی مشخص است که داده جدید در کجای لیست وارد می‌شود و پیشفرض به انتهای لیست اضافه می‌شود هرچند با توابعی می‌توان این مکان را تغییر داد.

## دسترسی به مقادیر لیست :

هر یک از خانه‌های لیست دارای index و یا شناسه خاصی است. index لیست از شماره ۰ آغاز می‌شود یعنی خانه اول لیست دارای شناسه صفر است و به این شکل می‌توان به آن دسترسی داشت.

```
mylist[2]
```

علاوه بر مدل دسترسی بالا برای مشخص کردن دسترسی به یک بازه می‌توان از علامت : استفاده کرد و همچنین می‌توان برای انتخاب از انتهای لیست از اعداد منفی هم مانند نمونه زیر استفاده کرد

```
mylist[0:4]
```

```
mylist[-2]
```

## تغییر مقادیر لیست:

برای تغییر مقادیر لیست کفایت بصورت معمولی یکی از خانه‌ها را انتخاب کرده و مقدار آن را تغییر دهیم.

```
mylist[2] = "gholi"
```

## اضافه کردن مقدار به لیست:

اگر بخواهیم به انتهای لیست مقداری اضافه کنیم از تابع append() استفاده می‌کنیم.

```
mylist.append("taha")
```

برای اضافه کردن یک مقدار به خانه مشخص از دستور insert() به شکل زیر استفاده می‌کنیم که ابتدا index و سپس مقدار را وارد می‌کنیم.

```
mylist.insert(3, "taha")
```

برای اضافه کردن یک لیست به انتهای لیست فعلی از دستور extend() استفاده می‌کنیم.

```
mylist.extend(list1)
```

## حذف کردن از لیست:

برای حذف کردن یک مقدار مشخص از لیست از دستور remove() استفاده می‌کنیم.

```
mylist.remove("ali")
```

برای حذف کردن از انتهای لیست و یا مشخص کردن یک index برای حذف از دستور pop() استفاده می‌کنیم.

```
mylist.pop()
```

```
mylist.pop(1)
```

گاهی نیاز است لیستمان را خالی کنیم ولی آن را حذف نکنیم پس در این موارد از دستور clear() استفاده می‌کنیم.

```
mylist.clear()
```

و اگر خواستیم کل لیست را حذف کنیم به روش زیر از دستور del استفاده می‌کنیم.

```
del mylist
```

## مرتب سازی لیست:

برای اینکه لیست را مرتب کنیم میتوانیم از تابع `sort()` استفاده کنیم و برای اینکه این عملیات مرتب سازی معکوس انجام شود میتوان از فعال سازی `reverse` به شکل زیر استفاده کرد.

```
mylist.sort()
mylist.sort(reverse = True)
```

برای برعکس لیست از دستور `reverse()` استفاده میکنیم.

```
mylist.reverse()
```

گاهی نیاز است از لیستمان یک کپی بگیریم ، برای اینکار از دستور `copy()` استفاده میکنیم.

```
list1 = mylist.copy()
```

میتوان لیست ها را با هم جمع کرد و با این عملیات اتفاقی که میافتد این است که لیست دوم به انتهای لیست اول اضافه میشود.

```
list 2 = mylist + list1
```

برای شمارش تعداد تکرار مقادیر خاص در یک لیست از تابع `count()` استفاده میکنیم.

```
mylist.count("ali")
```

برای پیدا کردن `index` یک مقدار خاص در لیست از دستور `index()` استفاده میکنیم.

```
mylist.index("ali")
```

روش های پیمایش لیست میتواند به صورت های گوناگونی صورت پذیرد که در زیر سه روش آن با استفاده از حلقه ها آمده است:

### #The First Method

```
for x in list1:
    print(x)
```

### #The Second Method

```
for i in range(len(list1)):
    print(list1[i])
```

### #The Third Method

```
i = 0
while i < len(list1):
    print(list1[i])
    i = i + 1
```

## لیست های دو بعدی :

در پایتون برای ساخت لیست های دو بعدی کافیهست که داخل یک لیست ، لیست بسازیم

```
thislist = [ [1,2,3] , [4,5,6] , [7,8,9] ]
```

نحوه دسترسی به این لیست ها به این صورت است که ابتدا باید از براکت اول ، لیست داخلی را انتخاب کنیم و از براکت دوم به خانه های آن لیست داخلی دسترسی پیدا کنیم ، برای مثال برای دسترسی به شماره ۴ در لیست بالا باید اینچنین آن را صدا بزنیم:

```
thislist [1][0]
```