

List Comprehension

List comprehension nədir?

Python-da yeni siyahı yaratmağın **qısa, səliqəli və sürətli üsuludur**.

Adi dövr (for loop) ilə etdiyimiz əməliyyatı **tək sətirə** yazı bilərik.

List comprehension necə işləyir?

Bir list comprehension 3 hissədən ibarətdir:

1. Expression (çıxış hissəsi):

Yeni siyahıya hansı dəyəri daxil edəcəyik?

Məsələn `x*2`, yaxud `x.upper()` və s.

2. For döngəsi

Hansı siyahı üzərində dövr edirik?

3. Şərt (optional)

İstəsən filtr də əlavə edə bilərsiniz.

Məsələn: `if x % 2 == 0`

Ümumi forma:

```
[new_value for element in list if condition]
```

List comprehension harada istifadə olunur?

❖ Filtrləmə

(şərt verib yalnız bəzi elementləri seçmək)

❖ Dəyərləri dəyişərək yeni siyahı yaratmaq

❖ Məlumatı təmizləmək

(boşluqları, hərflər olmayanları çıxarmaq və s.)

❖ String → rəqəm, rəqəm → string çevirmək

Python-da siyahı ilə bağlı olan hər işin qısa versiyası list comprehension ilə olur.

Sadə nümunələrlə izah

1. Rəqəmlərin kvadratlarını yaratmaq

```
kv = [x**2 for x in range(1, 6)]  
print(kv)
```

```
[1, 4, 9, 16, 25]
```

2. Tək ədədləri seçmək

```
nums=[1,2,3,4,5]  
tek = [x for x in nums if x % 2 != 0]  
print(tek)
```

```
[1, 3, 5]
```

3. Cümlədən yalnız hərfləri çıxarmaq

```
cumle="salam"  
herfler = [i for i in cumle if i.isalpha()]  
print(herfler)
```

```
['s', 'a', 'l', 'a', 'm']
```

4. Sözləri böyük hərflə yazmaq

```
cumle="Hello World"  
sozler = [s.upper() for s in cumle.split()]  
print(sozler)
```

```
['HELLO', 'WORLD']
```

5. Təkrarlanan elementləri çıxarmaq

```
lst = [1,1,2,3,4,5]  
unikal = list({x for x in lst})  
print(unikal)
```

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

List comprehension niyə yaxşıdır?

✓ Qısa görünür

3-4 sətirlik kod 1 sətir olur.

✓ Daha oxunaqlıdır

Struktur aydındır: çıxış → dövr → şərt.

✓ Daha sürətlidir

Python-un daxili mexanizmləri loop-dan daha sürətli işləyir.