

Python funksiyasında arqumentlər

Funksiyaya arqument vermək üçün adətən belə yazırıq:

```
def salam(ad, yas):  
    print(f"Salam {ad}, sən {yas} yaşındasın.")  
  
salam("Aysel", 18)
```

Burada ad və yas → **müəyyən edilmiş arqumentlər** (obligator)

Amma bəzən biz **dəyişən sayda arqument** göndərmək istəyirik

*args nədir?

- *args → **dəyəri mövqeyə görə (positional) verilən dəyişən sayda arqumentləri** qəbul edir
- Nəticədə bu arqumentlər **tuple** (dəyişməz siyahı) şəklində saxlanılır.

Niyə tuple?

Tuple **dəyişməzdır**, ona görə də funksiyada arqumentlərin orijinal dəyərləri dəyişmir.

```
def toplama(*sayilar):  
  
    cəm = sum(sayilar)  
  
    print("Cəm:", cəm)  
  
  
toplama(1, 2, 3)  
  
toplama(5, 10, 15, 20)
```

**kwargs nədir?

kwargs → **adlı (keyword) arqumentləri qəbul edir.

Nəticədə bunlar **dictionary** şəklində saxlanılır.

```
def məlumat(**data):  
    for key, value in data.items():  
        print(key, ":", value)  
  
məlumat(ad="Aysel", yaş=18, şəhər="Bakı")
```

*args və **kwargs eyni funksiyada

```
def fun(a, b, *args, **kwargs):  
    print("a:", a)  
    print("b:", b)  
    print("args:", args)  
    print("kwargs:", kwargs)  
  
fun(1, 2, 3, 4, 5, x=100, y=200)
```

Vacib Qaydalar

1. ***args həmişə obligator və ya müvəqqəti arqumentlərdən sonra** yazılır
2. ****kwargs həmişə *args-dən sonra** yazılır
3. **_ dəyişəni for _ in range(n) kimi hallarda istifadə oluna bilər,** əgər dəyişən lazım deyilsə.