

PYTHON FUNKTSIYALARI: DASTURLASHDA YORDAMCHI ASOSLAR

Aslonov Qodir Ziyodulayevich

Osiyo halqaro universiteti kafedra assistenti

Saydumarov Ibaydullo Ikromjon o'gli

Osiyo xalqaro universiteti talabasi

Annotatsiya: Ushbu maqolada Python dasturlash tilida funktsiyalar qanday ishlaydi, ularning turlari, foydalari va foydalanish usullari haqida ma'lumot beriladi. Maqola dasturchilar va o'qituvchilar uchun muhim tushunchalarni taqdim etadi, shuningdek, amaliy misollar bilan boyitiladi.

Kalit so'zlar: Python, funktsiyalar, dasturlash, kodni qayta ishlatish, parametrlar, qaytish qiymati.

KIRISH

Python dasturlash tilida funktsiyalar — bu kodning qayta ishlatilishi va modularizatsiyasini ta'minlaydigan muhim vositadir. Funktsiyalar yordamida dasturchilar o'z kodlarini yanada tushunarli va samarali qilishlari mumkin. Ushbu maqolada funktsiyalarning asosiy tushunchalari va ulardan qanday foydalanish haqida batafsil ma'lumot beramiz.

Asosiy Qism

1. Funktsiya Nima?

Funktsiya — bu ma'lum bir vazifani bajarish uchun yozilgan kodning blokidir. Funktsiyalar kodni modularizatsiya qilish va uni qayta ishlatish imkonini beradi.

2. Funktsiya Yaratish

Funktsiya yaratish uchun def kalit so'zidan foydalanamiz. Misol:

```
def greet(name):  
    return f"Salom, {name}!"
```

3. Funktsiyalarning Turlari

- **Oddiy funktsiyalar:** Faqatgina bir vazifani bajaradi.
- **Parametrlil funktsiyalar:** Funktsiyaga ma'lum parametrlar o'tkaziladi.
- **Qaytish qiymatiga ega funktsiyalar:** Funktsiya bajarilgandan so'ng, natijani qaytaradi.

4. Parametrlar va Argumentlar

Funktsiyalarga beriladigan ma'lumotlar parametrlar deb ataladi. Argumentlar esa bu parametrlar uchun haqiqiy qiymatlardir. Misol:

```
def add(a, b):  
    return a + b  
result = add(5, 3) # result = 8
```

5. Funktsiyalarning Foydalari

- **Kodning takrorlanishini kamaytiradi.**
- **Kodning o'qilishi va tushunarli bo'lishini oshiradi.**

- **Kodni parchalash orqali murakkab vazifalarni soddalashtiradi.**

6. Lambda Funktsiyalari

Python'da qisqa funktsiyalar yaratish uchun lambda kalit so'zidan foydalaniladi.

Misol:

```
square = lambda x: x ** 2
print(square(4)) # Natija: 16
```

7. Funktsiyani yaratish

Pythonda funktsiya **def** kalit so'zi yordamida aniqlanadi:

Misol [O'zingizning Python serveringizni oling](#)

```
def my_function():
    print("Hello from a function")
```

8. Argumentlar

Axborot funktsiyalarga argument sifatida uzatilishi mumkin. Argumentlar funktsiya nomidan keyin qavslar ichida ko'rsatiladi. Siz xohlaganicha argumentlar qo'shishingiz mumkin, ularni vergul bilan ajratib qo'ying. Quyidagi misolda bitta argument (fname) bilan funktsiya mavjud. Funktsiya chaqirilganda, biz to'liq ismni chop etish uchun funktsiya ichida ishlatiladigan ismni o'tkazamiz:

Misol

```
def my_function(fname):
    print(fname + " Refsnes")
my_function("Emil")
my_function("Tobias")
my_function("Linus")
```

9. Argumentlar soni

Odatiy bo'lib, funktsiya to'g'ri argumentlar soni bilan chaqirilishi kerak. Ya'ni, agar sizning funktsiyangiz 2 ta argumentni kutsa, siz funktsiyani 2 ta argument bilan chaqirishingiz kerak, ko'p emas va kam emas.

Misol

Bu funktsiya 2 ta argumentni kutadi va 2 ta argumentni oladi:

```
def my_function(fname, lname):
    print(fname + " " + lname)
my_function("Emil", "Refsnes")
```

Funksiya faqat chaqirilgan vaqtda ishlaydigan kod blogi hisoblanadi. Funktsiyaga parametrlar sifatida ma'lumotlarni uzatish mumkin. Funktsiyalar muayyan bir vazifani bajarish uchun ishlatiladi. Kod yozish jarayonida yaratgan funktsiyadan bir yoki bir nechta marta ishlatish yoki umuman ishlatmaslik imkoniyatiga egasiz.

Funksiya Yaratish

Python dasturlash tilida funktsiya yaratish uchun **def** kalit so'zi yordamida yaratiladi. quyidagi misolga qarang.

```
def my_function():
    print("Hello UzbekDevs")
```

Funksiyani chaqirish

Funksiyani chaqirishda uning nomidan foydalanamiz agar parametrlar bolsa parametrlarni junatamiz aks holda bo'sh () qavsdan foydalanamiz.

```
def my_function():  
    print("Hello UzbekDevs")
```

```
my_function()
```

Argumentlar

Funksiya beriladigan argumentlar funksiya nomi yonidagi qavs ichida keltirilib o'tilishi zarur. bunda siz belgilangan argumentlarni funksiyaga junatish imkoniyatiga ega bo'lasiz. Quyidagi misolda **name** argument junatib keltirilgan ismni "developer" degan so'zni qo'shib chiqaradigan funksiya yaratamiz.

```
def my_function(name):  
    print(name + " Developer")
```

```
my_function("Farrukh")  
my_function("Qudrat")  
my_function("Khushnud")
```

XULOSA

Python funktsiyalari dasturlash jarayonida muhim ahamiyatga ega. Ular kodni soddalashtirish, qayta ishlatish va tuzilmaviy yondashuvni ta'minlaydi. Ushbu maqolada funktsiyalarning asosiy tushunchalari va amaliy misollar yordamida ular qanday ishlashini ko'rsatdik. Python dasturlash tilini o'rganayotganlar uchun funktsiyalarni tushunish juda muhimdir.

FOYDALANILGAN ADABIYOTLAR:

1. Python Documentation.(n.d.) .Retrieved from <https://docs.python.org/3/tutorial/controlflow.html#defining-functions>
2. Lutz, M. (2013). Learning Python. O'Reilly Media.
3. Sweigart, A. (2019). Automate the Boring Stuff with Python. No Starch Press.
4. Jalolov, T. S. (2024). YUQORI HAJMLI MA'LUMOTLARNI QAYTA ISHLASHDA PYTHON KUTUBXONALARI. *MASTERS*, 2(5), 121-128.
5. Jalolov, Tursunbek Sadridinovich. "YUQORI HAJMLI MA'LUMOTLARNI QAYTA ISHLASHDA PYTHON KUTUBXONALARI." *MASTERS* 2.5 (2024): 121-128.
6. Jalolov T. S. YUQORI HAJMLI MA'LUMOTLARNI QAYTA ISHLASHDA PYTHON KUTUBXONALARI //MASTERS. – 2024. – T. 2. – №. 5. – C. 121-128.