

Python programming language

Week 9. Python functions

Alisher Ismailov

Lecturer

Department of Information technology
Andijan Branch of Tashkent Financial Institute

Email: alisherismailov534@gmail.com

Python dasturlash tili

9-Mavzu. Python funksiyalar

Ma'ruzachi: Alisher Ismoilov








Axborot texnologiyalari kafedrası

Toshkent moliya instituti Andijon filiali

Elektron pochta: alisherismailov534@gmail.com

9-Mavzu. Python functions

Reja:

-  Funksiya yaratish
-  Funksiyani chaqirish
-  Argumentlar
-  Standart parametr qiymati
-  List to'plamini argument sifatida foydalanish
-  Return (Qaytish) qiymatlari
-  Pass kalit so'zi

Python dasturlash tilida Funksiya

Funksiya kod bloki bo'lib, u faqat chaqirilganda ishlaydi. Funksiyani amaliy mashg'ulotlar orqali ko'rib chiqamiz

Python dasturlash tilida Funksiya

- Python dasturlash tilida funksiya **def** kalit soʻzi yordamida yaratiladi:

```
def Funksiya_Yaratish():  
    print("Funksiyadan Salom")
```

Python dasturlash tilida Funksiya

- Funksiyani chaqirish uchun funksiya nomidan keyin qavsdan funksiya_nomi() foydalanish kerak:

```
def Funksiya_Yaratish():  
    print("Funksiyadan Salom")
```

```
Funksiya_Yaratish()
```

Python dasturlash tilida Funksiya

- **Argumentlar**

- Axborot funksiyalarga argument sifatida uzatilishi mumkin. Bu argumentlar yoki parametr deb nomlanadi. Argumentlar funksiya nomidan keyin qavslar ichidagi o'zgaruvchi hisoblanadi. Funksiyaga xoxlagancha argumentlar qo'shishingiz mumkin, ularni vergul bilan ajratib qo'yish kerak. Quyidagi misolda bitta argument (ism) bilan funksiya yaratildi. Funksiya chaqirilganda, biz to'liq ismni chop etish uchun funksiya ichida ishlatiladigan ismni joylaymiz:

Python dasturlash tilida Funksiya

- **Argumentlar**

- Axborot funksiyalarga argument sifatida uzatilishi mumkin. Bu argumentlar yoki parametr deb nomlanadi. Argumentlar funksiya nomidan keyin qavslar ichidagi o‘zgaruvchi hisoblanadi. Funksiyaga xoxlagan cha argumentlar qo‘shishingiz mumkin, ularni vergul bilan ajratib qo‘yish kerak. Quyidagi misolda bitta argument (ism) bilan funksiya yaratildi. Funksiya chaqirilganda, biz to‘liq ismni chop etish uchun funksiya ichida ishlatiladigan ismni joylaymiz:

```
def ism_print(ism):  
    print("Ismingiz: "+ism)
```

```
ism_print("Alisher")
```

Python dasturlash tilida Funksiya

- **Argumentlar soni**

- Funksiya to‘g‘ri argumentlar soni bilan chaqirilishi kerak. Ya‘ni, agar sizning funksiyangizda 2 ta argument berilgan bo‘lsa, siz funksiyani 2 ta argument bilan chaqirishingiz kerak.

```
def Funksiya_ism(ism, familya):  
    print("Ismingiz: "+ism+"", Familyangiz:""+familya)
```

```
Funksiya_ism("Alisher", "Ismailov")
```

Python dasturlash tilida Funksiya

- **Argumentlar soni**

- Agar siz funktsiyani 1 ta yoki 3 ta argument bilan chaqirmoqchi bo'lsangiz, siz xatoga duch kelasiz

```
def Funksiya_Ism(ism, familya):  
    print("Ismingiz: "+ism+", Familyangiz: "+familya)
```

```
Funksiya_Ism("Lola")
```

```
-----  
TypeError                                Traceback (most recent call last):  
~\AppData\Local\Temp\ipykernel_6668\2930385202.py in <module>  
      2     print("Ismingiz: "+ism+", Familyangiz: "+familya)  
      3  
----> 4 Funksiya_Ism("Lola")
```

```
TypeError: Funksiya_Ism() missing 1 required positional
```

Python dasturlash tilida Funksiya

- **Ixtiyoriy argumentlar, *args**

- Agar funksiyangizga qancha argument o'tkazilishini bilmasangiz, funksiya ta'rifida parametr nomidan oldin * belgisini qo'shing.

```
def mening_funksiyam(*ism):  
    print("Do'stimning ismi: "+ism[2])
```

```
mening_funksiyam("Alisher", "Bobur", "Ulugbek")
```

Python dasturlash tilida Funksiya

- **Kalit soʻz argumentlari**

- Siz argumentlarni kalit = qiymat sintaksisi bilan ham yuborishingiz mumkin.

```
def mening_funksiyam(*ism):  
    print("Doʻstimning ismi: "+ism2)
```

```
mening_funksiyam(ism1 = "Alisher", ism2 = "Bobur", ism3 =  
"Ulugbek")
```

Python dasturlash tilida Funksiya

- **Ixtiyoriy kalit soʻz argumentlari, **kwargs**
- Agar funksiyangizga qancha kalit soʻz argumentlari oʻtkazilishini bilmasangiz, ikkita yulduzcha belgisini qoʻshing: ** funksiya taʼrifidagi parametr nomidan oldin.

Python dasturlash tilida Funksiya

- **Ixtiyoriy kalit soʻz argumentlari, **kwargs**
- Agar kalit soʻz argumentlari soni nomaʼlum boʻlsa, parametr nomidan oldin ikkita ** belgisini qoʻshing:

```
def mening_funksiyam(**ism):  
    print("Doʻstimning ismi: "+ism["ism3"])
```

```
mening_funksiyam(ism1 = "Alisher", ism2 = "Bobur", ism3 =  
"Ulugbek")
```

Python dasturlash tilida Funksiya

- **Standart parametr qiymati**

- Quyidagi misol standart parametr qiymatidan qanday foydalanishni ko'rsatadi.
- Agar funktsiyani argumentsiz chaqirsak, u standart qiymatdan foydalanadi:

```
def mening_vatanim(vatan = "O`zbekiston"):  
    print("Mening vatanim: "+vatan)
```

```
mening_vatanim()
```

Python dasturlash tilida Funksiya

- **List to'plamini argument sifatida foydalanish**

- Har qanday turdagi argumentlarni funksiyaga yuborishingiz mumkin va u funksiya ichida bir xil ma'lumotlar turi sifatida ko'rib chiqiladi.
- Masalan, Agar siz Listni argument sifatida yuborsangiz, u funksiyaga kelganda ham List bo'lib qoladi:

```
def Funksiya_D_Tillari(dasturlash):  
    for x in dasturlash:  
        print(x)
```

```
dasturlash_tili = ["Python", "Java", "C"]  
Funksiya_D_Tillari(dasturlash_tili)
```

Python dasturlash tilida Funksiya

- **Return (Qaytish) qiymatlari**
- Funksiyaga qiymat qaytarishiga ruxsat berish uchun return kalit soʻzidan foydalanish mumkin:

```
def return_qiymati(son):  
    return son * 5
```

```
print(return_qiymati(5))  
print(return_qiymati(3))
```

Python dasturlash tilida Funksiya

- **Pass kalit so'zi**

- Funksiya ta'riflari bo'sh bo'lishi mumkin emas, lekin agar sizda biron sababga ko'ra kontentsiz funksiya bo'lsa, xatolikka yo'l qo'ymaslik uchun pass kalit so'zini kiritish kerak.

```
def pass_qiymati():  
    pass
```

```
pass_qiymati()
```

Python dasturlash tilida Funksiya

- **Funksiyalarga amaliy mashg`ulotlar**
-
- 1-misol
- Maks qiymatni qaytradigan funksiya
-

```
def maks_qiymat(son1, son2)
    if son1 > son2:
        return son1
    else:
        return son2
```

Python dasturlash tilida Funksiya

- Funksiyalarga amaliy mashg`ulotlar
-
- 2-misol
- Minimum qiymatni qaytradigan funksiya

```
def min_qiymat(son1, son2):  
    if son1 < son2:  
        return son1  
    else:  
        return son2
```

Python dasturlash tilida Funksiya

- Funksiyalarga amaliy mashg`ulotlar
-
- 3-misol
- 3 ta son orasidan maks qiymatni qaytradigan funksiya

```
def maks_qiymat(son1, son2, son3):  
    if son1 > son2 and son1 > son3:  
        return son1  
    if son1 < son2 and son2 > son3:  
        return son2  
    else:  
        return son3
```

Python dasturlash tilida Funksiya

- Funksiyalarga amaliy mashg`ulotlar
-
- 4-misol
- Sonni 1 dan to`qqizgacha ko`paytirish

```
def kopaytma(son):  
    for x in range(1, 10):  
        print(son + " * " + str(x) + " = " + str(son * x))
```

Python dasturlash tilida Funksiya

- Funksiyalarga amaliy mashg`ulotlar
-
- 5-misol
- Hafta kunlarini ekranga chop etadigan funksiya

```
def haftaKunlari():
```

```
    haftaKun = ["Dushanba", "Seshanba", "Chorshanba",  
"Payshanba", "Juma", "Shanba", "Yakshanba"]
```

```
    for x in haftaKun:
```

```
        print(x)
```

Python dasturlash tilida Funksiya

- Funksiyalarga amaliy mashg`ulotlar
-
- 6-misol
- Hafta kunlarini ekranga chop etadigan funksiya

```
def haftaKunlari(kun):  
    for x in kun:  
        print(x)
```

```
haftaKun = ["Dushanba", "Seshanba", "Chorshanba", "Payshanba",  
"Juma", "Shanba", "Yakshanba"]
```

```
haftaKunlari(haftaKun)
```

Python dasturlash tilida Funksiya

- Funksiyalarga amaliy mashg`ulotlar
-
- 7-misol
- Oy nomlarini ekranga chop etadigan funksiya

```
def OyNomlari():
```

```
    oy = ["Yanvar", "Fevral", "Mart", "Aprel", "May",  
"Iyun", "Iyul", "Avgust", "Sentabr", "Oktabr", "Noyabr",  
"Dekabr"]
```

```
    for x in oy:
```

```
        print(x)
```

Python dasturlash tilida Funksiya

- Funksiyalarga amaliy mashg`ulotlar
-
- 8-misol
- Oy nomlarini ekranga chop etadigan funksiya

```
def OyNomlari(Oy):  
    for x in Oy:  
        print(x)
```

```
OyNomlari = ["Yanvar", "Fevral", "Mart", "Aprel", "May", "Iyun",  
"Iyul", "Avgust", "Sentabr", "Oktabr", "Noyabr", "Dekabr"]
```

```
OyNomlari(OyNomlari)
```

Python dasturlash tilida Funksiya

- Funksiyalarga amaliy mashg`ulotlar
-
- 9-misol
- Sonni juft yoki toq sonligini aniqlaydigan funksiya yaratish

```
def juft_toq(son):  
    if son % 2 == 0:  
        print("Juft son")  
    else:  
        print("Toq son")
```

Python dasturlash tilida Funksiya

- **Funksiyalarga amaliy mashg`ulotlar**
-
- 10-misol
- 1 dan 100gacha bo`lgan sonlarni ekranga chop etadigan funksiya yaratish

```
def yuzgacha():  
    for x in range(1, 101)  
        print(x)
```

Python dasturlash tilida Funksiya

- Funksiyalarga amaliy mashg`ulotlar
-
- 11-misol
- 2 ta sonni yig`indisini topadigan funksiya yaratish

```
def yigindi(son1, son2):  
    return son1+son2
```

```
yigindi(5, 6)
```

Foydalanilgan adabiyotlar

1. Mastering Object-Oriented Python: Build powerful applications with reusable code using OOP design patterns and Python 3.7, 2nd Edition, Steven F. Lott, Packt Publishing (June 14, 2019)
2. Learning Python, 5th Edition Fifth Edition, Mark Lutz , O'Reilly Media, June 12, 2013
3. Python Programming for Beginners: The Ultimate Guide for Beginners to Learn Python Programming: Crash Course on Python Programming for Beginners, AMZ Publishing, Independently published (July 13, 2021)
4. <https://www.python.org/>
5. <https://www.w3schools.com/>
6. <https://www.codecademy.com/catalog/language/python>
7. <https://realpython.com/>
8. <https://www.anaconda.com/>

**E'tiboringiz
uchun rahmat!**