

Python iteratorlari

Iterator - bu sanab bo'ladigan qiymatlarni o'z ichiga olgan ob'ektdur.

Iterator - bu takrorlanadigan ob'ekt, ya'ni siz barcha qiymatlar bo'ylab o'tishingiz mumkin (ya'ni barcha qiymatga alohida murojat qilishingiz mumkin hamda ular ustida bazi amallarni bajarishingiz mumkin).

Texnik jihatdan, Pythonda iterator - bu iterator protokolini amalga oshiradigan ob'ekt bo'lib, u usullar `iter()` hamda `next()` kalit so'zlari orqali yaratiladi.

Iterator va takrorlanuvchi

Ro'yxatlar, kortejlar, lug'atlar va to'plamlar takrorlanadigan ob'ektlardir. Ular iteratorni olishingiz mumkin bo'lgan takrorlanadigan konteynerlardir tushunarli qilib aytganda kodlar jamlanmasidagi kalit so'zlardir. Ushbu ob'ektlarning barchasi `iter()` iteratori olib ishlatish usulga ega.

Iteratorlar ko'p hollarda keying elementga o'tishda ishlatiladi. Sizlar bilan oddiy bir iteratorni ko'rishimiz mumkin **Kod_1**

Iterator parametri `iter()` metodi orqali olingan iterator obyektini ifodalaydi. `next()` metodi esa keyingi elementni qaytaradi.

Agar iteratorning oxiriga yetib ketsangiz, **StopIteration** xato tufayli xatolik yuz beradi. Shuning uchun, iterator obyektini to'xtatish uchun siz **StopIteration** xatolikni yozishingiz kerak. **Kod_2**

Python iteratori satrlar, shuningdek, belgilar ketma-ketligini o'z ichiga olgan takrorlanadigan metoddir: **Kod_3**

Iteratorni loop oilasida keng ko'lamda ishlatishimiz mumkin, ya'ni forda ham while da ham keng ko'lamda ishlatilgan python hech qanday xatolik qaytarmaydi: Bularni biz **Kod_4** hamda **Kod_5** da ko'rishimiz mumkin.

Loop for aslida iterator ob'ektini yaratadi va har bir tsikl uchun next() usulini bajaradi.

Ob'ektni sinfni iterator sifatida yaratish uchun siz `iter()` va `next()` ob'ektingizga amal qilishingiz kerak.

Python sinflari ob'ektlari bo'limida bilib olganingizdek, barcha sinflarda `__init__` ob'ekt yaratilayotganda bir oz ishga tushirishni amalga oshirish imkonini beruvchi funksiya mavjud.

Metod `iter()` shunga o'xshash ishlaydi, siz operatsiyalarni bajarishingiz mumkin (boshlash va h.k.), lekin har doim iterator ob'ektining o'zini qaytarishingiz kerak.

Metod `next()` shuningdek, operatsiyalarni bajarishga imkon beradi va keyingi elementni ketma-ketlikda qaytarishi kerak. **Kod_6**

Iterator ni to'xtatish:

Yuqoridagi misol, agar sizda yetarlicha `next()` iboralari bo'lsa yoki u tsiklda ishlatilsa, abadiy davom etadi.

Takrorlashning abadiy davom etishiga yo'l qo'ymaslik uchun biz `StopIteration` bayonotidan foydalanishimiz mumkin.

Usulda `next()`, agar takrorlash ma'lum bir necha marta bajarilsa, xatoni ko'tarish uchun tugatish shartini qo'shishimiz mumkin. **Kod_7**