

PYTHON DASTURLASH TILIDAGI " RANDOM" MODULI TASODIFIY SONLARNI YARATISH

Jalolov Tursunbek Sadridinovich

Osiyo Xalqaro Universiteti assitenti

Odilova Fotima Oltiboyevna

Osiyo Xalqaro Universiteti talabasi

Annotatsiya.*Python dasturlash tilidagi " tasodifiy " moduli tasodifiy sonlarni yaratish va tasodifiylik bilan bog'liq turli operatsiyalarni bajarish uchun vositalarni taqdim etadi. Ushbu maqolada Python-dagi " tasodifiy" modulning roli va qo'llanilishi , uning funksiyalari, qo'llanilishi va modellashtirish, test qilish va o'yinni ishlab chiqish kabi turli sohalaridagi ahamiyati ko'rib chiqiladi. Shuningdek, biz soxta tasodifiy raqamlar va kriptografik tasodifiylik o'rtasidagi farqni ko'rib chiqamiz. Xulosa qilib, dasturlashda tasodifiy jarayonlar bilan ishlash uchun " tasodifiy " modulning ahamiyati va ko'p qirraliligi haqida xulosa chiqariladi.*

KIRISH

Tasodifiylik real hayotda ham, dasturlashda ham ko'plab jarayonlarning ajralmas jihati hisoblanadi. Python-da " tasodifiy " moduli psevdotasodifiy raqamlarni yaratish va tasodifiy tanlash yoki aralashtirishni talab qiladigan vazifalarni bajarish uchun funksiyalarni ta'minlaydi. Ushbu modul turli xil ilovalarda qo'llaniladi: test ma'lumotlarini yaratishdan o'yin algoritmlarini yaratish va murakkab tizimlarni modellashtirishgacha. Tasodifiy ma'lumotlardan foydalanishni talab qiladigan muammolar dasturlashning ko'plab sohalarida uchraydi. Dasturchilar oldindan aytib bo'lmaydigan o'yin mexanikasini yaratish, dasturiy ta'minot komponentlarini turli sharoitlarda sinab ko'rish va matematik modellashtirish uchun tasodifiy raqamlardan foydalanishlari mumkin. Bu barcha vazifalarning asosi psevdotasodifiy raqamlarni yaratish va ular bilan operatsiyalarni amalga oshirish funksiyasini amalga oshiradigan " tasodifiy " moduldir. Ushbu maqolaning maqsadi " tasodifiy " modulning imkoniyatlarini, uning ahamiyatini va zamonaviy dasturiy echimlarda qo'llanilishini ko'rib chiqish va Python dasturlash kontekstida psevdotasodifiylik va haqiqiy tasodifiylik o'rtasidagi farqni tushuntirishdir .

ASOSIY QISM

1. Pythonda psevdotasodifiylik

Python-da tasodifiy sonlarni yaratish deterministik algoritmlar yordamida yaratilgan psevdotasodifiy raqamlardan foydalanishga asoslangan. Garchi bunday raqamlar tasodifiy ko'rinsa-da, ular aslida boshlang'ich holati yoki urug' (urug') bilan takrorlanishi mumkin. Bu ularni dastur kodini sinash va disk raskadrovka qilish uchun qulay qiladi, chunki bir xil "tasodifiy" natijani qayta-qayta olish mumkin.

random.seed () funksiyasi tasodifiy sonlar generatorining urug'lik holatini o'rnatish imkonini beradi. Agar siz `seed` uchun bir xil qiymatni o'rnatmangiz, dastur har safar bajarilganda tasodifiy sonlar ketma-ketligi bir xil bo'ladi:

```
tasodifiy import qilish
```

```
tasodifiy.seed(10)
```

```
print(random.random()) # 0,5714025946899135
```

```
print(random.random()) # 0.4288890546751146
```

Shunday qilib, psevdotasodifiylik takrorlanuvchanlik va tasodifiy sonlar ustidan nazoratni kafolatlaydi, bu ayniqsa dasturlarni sinash va disk raskadrovka qilishda foydalidir.

2. "Tasodifiy" modulining asosiy funksiyalari

Tasodifiy moduli tasodifiy sonlar va ketma-ketliklar bilan ishlash uchun ko'plab funktsiyalarni taklif etadi. Keling, ulardan eng muhimlarini ko'rib chiqaylik.

a) Tasodifiy sonlarni hosil qilish

tasodifiy (): [0,0, 1,0] oraliqdagi tasodifiy suzuvchi nuqta sonini qaytaradi.

```
print(random.random()) # Masalan , 0,3745401188473625
```

randint (a, b): "a" dan "b" gacha bo'lgan tasodifiy butun sonni qaytaradi.

```
print(random.randint(1, 10)) # Masalan , 7
```

uniform (a, b): [a, b] oralig'ida tasodifiy suzuvchi nuqta sonini qaytaradi.

```
print(random.uniform(1.5, 2.5)) # Masalan , 2.1347
```

b) Ketma-ketliklar bilan ishlash

tanlov (seq): Berilgan ketma-ketlikdan tasodifiy elementni qaytaradi.

```
mevalar = ["olma", "banan", "gilos"]
```

```
print(random.choice(fruits)) # Masalan , 'banan'
```

aralash (seq): tartibini tasodifiy o'zgartirib, ketma-ketlikni joyida aralashtirib yuboradi.

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

```
random.shuffle(pastki)
```

```
print( deck ) # Masalan, [3, 1, 4, 5, 2]
```

namuna (populyatsiya , k) : Berilgan ketma-ketlikdan k noyob tasodifiy elementlar ro'yxatini qaytaradi.

```
raqamlar = list (diapazon (100))
```

```
print(random.sample(raqamlar, 5)) # Masalan , [5, 62, 23, 1, 99]
```

Bu xususiyatlar ma'lumotlar to'plamidan tasodifiy ravishda elementlarni tanlash yoki ob'ektlarni aralashtirish kabi murakkab vazifalarni amalga oshirishni osonlashtiradi.

3. Tasodifiylikning turli sohalarda qo'llanilishi

a) O'yin sanoati

Tasodifiylik - oldindan aytib bo'lmaydigan hodisalar va o'yin mexanikasini yaratish uchun o'yinlardagi asosiy element. Masalan, poker yoki blackjack kabi karta o'yinlarida kartalarni tasodifiy aralashtirish kerak:

```
pastki = list(diapazon(1, 53)) # Deck 52 ta karta
```

```
random.shuffle ( pastki )
```

```
print( pastki [:5]) # Beshta tasodifiy karta
```

Yoki zar o'ynash o'yinlarida rulonlarni simulyatsiya qilish uchun tasodifiylik ishlatiladi:

```
def roll_dice():
```

```
    qaytish random.randint(1, 6)
```

```
    print( roll_dice () ) # Masalan, 4
```

b) Tasodifiy jarayonlarni modellashtirish

Fizika, moliyaviy bozorlar yoki biologik tizimlardagi zarrachalar harakati kabi tasodifiy jarayonlarni modellashtirish real voqealarni simulyatsiya qilish uchun tasodifiylikdan foydalanishni talab qiladi. Python tasodifiy ma'lumotlarni yaratish orqali bunday jarayonlarni modellashtirishga imkon beradi:

```
def simulate_coin_flips(n):
```

```
    natijalar = [random.choice([" Heads ", " Tails "]) diapazondagi _ uchun(n)]
```

```
    natijalarni qaytarish
```

```
    print(simulate_coin_flips(10)) # Masalan , [ ' Heads ' , ' Tails ' , ' Heads ' , ' Tails ' , ' Heads ' ]
```

c) Sinov

Dasturiy ta'minotni ishlab chiqishda tasodifiy ma'lumotlar ko'pincha tizimlarni sinab ko'rish va ularni turli stsenariylarga chidamli qilish uchun ishlatiladi. Masalan, tartiblash algoritmini sinab ko'rish uchun tasodifiy ma'lumotlar qatorini yaratishingiz mumkin:

```
test_data = [random.randint(0, 1000) _ diapazondagi(100)]
```

Bu sizga algoritmning barqarorligini ta'minlab, turli xil kirishlar bilan qanday ishlashini ko'rish imkonini beradi.

4. Kriptografik tasodifiylik

Pseudorandom raqamlar har doim ham parollar yoki kriptografik kalitlarni yaratish kabi xavfsizlik bilan bog'liq vazifalar uchun mos kelmaydi. Bunday hollarda kriptografik jihatdan kuchli tasodifiy sonlarni yaratishni ta'minlaydigan " sirlar" kutubxonasidan foydalaniladi:

```
import sirlari
```

```
parol = ''.join(secrets.choice('abcdefghijklmnopqrstuvwxyz0123456789') i diapazonidagi(8))
```

```
print( parol ) # Masalan, '4f8zj9kq'
```

sirlar maksimal oldindan aytib bo'lmaydigan vazifalarni bajarish uchun ishonchli vositalarni taqdim etadi.

XULOSA

tasodifiy moduli turli xil dasturlash sohalarida ilovalarga ega tasodifiylik bilan ishlash uchun kuchli va moslashuvchan vositani taqdim etadi. O'yinlardan dasturiy ta'minotni sinovdan o'tkazishgacha murakkab tizimlarni simulyatsiya qilishgacha tasodifiylik moslashuvchan va interaktiv echimlarni yaratishda muhim rol o'ynaydi. " Tasodifiy " moduli tasodifiy ma'lumotlarni talab qiladigan ko'pgina vazifalar uchun

mos bo'lgan psevdotasodifiy raqamlarni yaratishni ta'minlaydi. Xavfsizlikka sezgir bo'lgan vazifalar uchun Python tasodifiy sonlarning kriptografik kuchini kafolatlaydigan "sirlar" kutubxonasini taqdim etadi.

Python tilidagi "tasodifiy" modulning roli oldindan aytib bo'lmaydigan jarayonlar bilan ishlash uchun keng imkoniyatlarni taqdim etishdan iborat bo'lib, uni dasturchilar uchun eng muhim vositalardan biriga aylantiradi.

FOYDALANILGAN ADABIYOTLAR RO'YXATI:

1. Sadriddinovich, J. T. (2024). BASICS OF PSYCHOLOGICAL SERVICE. *PSIXOLOGIYA VA SOTSIOLOGIYA ILMIY JURNALI*, 2(4), 61-67.
2. Jalolov, T. S. (2024). PYTHONNING MATEMATIK KUTUBXONALARINI O'RGANISH: KENG QAMROVLI QO'LLANMA. *BIOLOGIYA VA KIMYO FANLARI ILMIY JURNALI*, 2(5), 71-77.
3. Jalolov, T. S. (2023). PARALLEL PROGRAMMING IN PYTHON. *TECHNICAL SCIENCE RESEARCH IN UZBEKISTAN*, 1(5), 178-183.
4. Jalolov, T. S. (2023). PYTHON DASTUR TILIDADA WEB-ILOVALAR ISHLAB CHIQISH. *TECHNICAL SCIENCE RESEARCH IN UZBEKISTAN*, 1(5), 160-166.
5. Jalolov, T. S. (2023). PYTHON TILINING AFZALLIKLARI VA KAMCHILIKLARI. *TECHNICAL SCIENCE RESEARCH IN UZBEKISTAN*, 1(5), 153-159.
6. Dj. Forse, P. Bisseks, U. CHan - Django. Razrabotka veb-prilopeniy na Python, Simvol-Plyus, 2010.
7. Eric Matthes Python Crash Course, a hands-on, Project-Based, Introduction to Programming, ISBN-13: 978-1-59327-603-4, 2016, 562 p