

TAKRORLASH OPERATORLARIDA BREAK VA CONTINUE BUYRUQLARIDAN FOYDALANISH

Tojimamatov Israiljon Nurmamatovich

Farg'ona davlat universiteti katta o'qituvchisi

israiltojimatov@gmail.com

Buzurukova Xumoraxon G'ayratjon qizi

Farg'ona davlat universiteti talabasi

khumorakhonb@gmail.com

Annotatsiya

Ushbu mavzu takrorlash operatorlarida (sikl operatorlarida) break va continue buyruqlarining ahamiyati va qo'llanilishi haqida ma'lumot beradi. Break operatori siklni to'liq to'xtatish uchun ishlatiladi va ma'lum shart bajarilganda sikldan chiqishni ta'minlaydi. Continue operatori esa joriy iteratsiyani yakunlab, siklning keyingi qadamiga o'tishni ta'minlaydi, shu bilan sikl tanasining ma'lum qismini o'tib ketishga yordam beradi. Ushbu maqola misollar orqali tushuntirilgan bo'lib, dasturlashda break va continue operatorlarining to'g'ri va samarali qo'llanilishi ko'rsatib o'tilgan.

Kalit so'zlar: Takrorlash operatorlari, sikl, break, continue, siklni boshqarish, shartli operatorlar, siklni to'xtatish, iteratsiya, dasturlash, Python, siklni nazorat qilish, kod optimallashtirish.

Аннотация

В этой теме представлена информация о важности и использовании команд break и continue в операторах цикла (операторах цикла). Оператор break используется для полной остановки цикла и выхода из цикла при выполнении определенного условия. Оператор continue завершает текущую итерацию и переходит к следующему шагу цикла, тем самым помогая пропустить определенную часть тела цикла. Эта статья объясняется на примерах и показывает правильное и эффективное использование операторов break и continue в программировании.

Ключевые слова: Операторы повторения, цикл, break, continue, управление циклом, условные операторы, остановка цикла, итерация, программирование, Python, управление циклом, оптимизация кода.

Annotation

This topic provides information about the importance and use of break and continue commands in loop operators (loop operators). The break operator is used to completely stop a loop and exits the loop when a certain condition is met. The continue operator completes the current iteration and moves to the next step of the loop, thus helping to skip a certain part of the loop body. This article is explained through

examples and shows the correct and effective use of break and continue operators in programming.

Keywords: Repetition operators, loop, break, continue, loop control, conditional operators, loop stopping, iteration, programming, Python, loop control, code optimization.

Kirish

Takrorlash operatorlarida break va continue buyruqlari sikl boshqaruvi uchun ishlatiladi. Break operatori sikl ichida ma'lum shart bajarilganda siklni to'liq to'xtatadi va sikldan chiqadi. Masalan, for yoki while siklida if sharti bajarilganda break yordamida sikl to'xtatiladi. Bu operator cheksiz sikllardan chiqish yoki kerakli natijaga erishilganda siklni to'xtatish uchun qo'llaniladi.

Continue operatori esa joriy iteratsiyani tugatib, siklning keyingi iteratsiyasiga o'tishni ta'minlaydi. Ya'ni, ma'lum shart bajarilganda sikl tanasining qolgan qismi bajarilmaydi va sikl keyingi qadamga o'tadi. Masalan, juft sonlarni o'tkazib yuborib, faqat toq sonlarni chiqarishda continue ishlatiladi.

1. for i in range(11):

if i == 5:

break # i 5 ga teng bo'lsa sikl to'xtaydi

print(i)

Natija: 0, 1, 2, 3, 4

2. for i in range(11):

if i % 2 == 0:

continue # juft sonlarda print qismi o'tkazib yuboriladi

print(i)

Natija: 1, 3, 5, 7, 9

Break va continue operatorlari dastur oqimini boshqarishda muhim bo'lib, siklni samarali boshqarish va kerakli shartlarda siklni to'xtatish yoki davom ettirish imkonini beradi. Shartlarni to'g'ri yozish zarur, aks holda sikl muammolari, masalan, cheksiz sikl yuzaga kelishi mumkin.

Break va continue operatorlari nima?

☞ Break operatori – break operatori takrorlash operatori (for yoki while) ichida ishlatiladi va belgilangan shart bajarilganda takrorlashni to'xtatadi. Bu operator yordamida takrorlashni erta to'xtatish mumkin, bu esa resurslarni tejash va kodni samarali qilishga yordam beradi.

☞ Continue operatori – takrorlashning joriy iteratsiyasini o'tkazib yuboradi va keyingi iteratsiyaga o'tadi. Bu operator yordamida ma'lum shartlarda kodning qolgan qismini bajarishni o'tkazib yuborish mumkin.

Qaysi sohalarda qo'llaniladi?

1. Ma'lumotlar tahlili (Data Analysis)	Katta hajmdagi ma'lumotlarni tahlil qilishda, ma'lum shartlarga javob bermaydigan ma'lumotlarni o'tkazib yuborish (continue) yoki kerakli ma'lumot topilgach takrorlashni to'xtatish (break) uchun ishlatiladi.
2. Veb dasturlash (Web Development):	Foydalanuvchi so'rovlarini qayta ishlashda, noto'g'ri yoki keraksiz so'rovlarni o'tkazib yuborish uchun continue, kerakli ma'lumot topilgach takrorlashni to'xtatish uchun break operatorlari qo'llaniladi.
3. Sun'iy intellekt va mashinani o'rganish (AI & Machine Learning)	Modelni o'qitish jarayonida, ma'lum shartlar bajarilganda takrorlashni to'xtatish (break) yoki ma'lum holatlarda iteratsiyani o'tkazib yuborish (continue) uchun ishlatiladi.
4. Avtomatlashtirilgan testlash (Automated Testing):	Test skriptlarida, ma'lum shartlar bajarilganda testni to'xtatish yoki o'tkazib yuborish uchun break va continue operatorlari ishlatiladi.

□ *Sun'iy intellekt (SI) sohasida qo'llanilishi*

🌐 Dunyo tajribasi – Sun'iy intellekt algoritmlari milliardlab ma'lumotlar ustida ishlaydi. SI modellarini o'qitish jarayonida ba'zida model belgilangan sifatga erishgach, darhol siklni to'xtatish kerak bo'ladi – bu holatda break printsiplari ishlatiladi. Ma'lumotlar orasida noto'g'ri, bo'sh yoki keraksiz yozuvlar bo'lishi mumkin. Ularni tahlil qilishning hojati yo'q – continue printsiplari orqali ular e'tiborsiz qoldiriladi.

- ✓ Resurslarni tejaydi (vaqt, energiya, hisoblash quvvati).
- ✓ Modellarning aniqligini oshiradi.
- ✓ Katta hajmdagi ma'lumotlar bilan samarali ishlashga imkon beradi.

🎓 *Ta'lim sohasida qo'llanilishi*

🌐 Global qo'llanilishi – Zamonaviy o'quv platformalari (Khan Academy, Coursera, DataCamp, Code.org) Python orqali o'quvchilarga dasturlashni o'rgatadi. Bu platformalarda o'quvchining noto'g'ri javobi bo'lsa — tizim keyingi savolga o'tadi (continue printsiplari). To'g'ri javob berilgach — test tugatiladi (break printsiplari).

- ✓ O'quvchi faqat kerakli bilimni oladi, keraksiz mashqlar bilan charchamaydi.
- ✓ Har bir o'quvchi o'z tempida o'rganadi.
- ✓ Shaxsiylashtirilgan o'quv yondashuvi shakllanadi.
- ✓ Muhim loyiha:
- ✓ Squirrel AI (Xitoy) — o'quvchilarni sun'iy intellekt asosida test qiladigan

platforma bo'lib, noto'g'ri bilim sohalarini aniqlab, ularni takrorlatadi.

✚ Tibbiyot sohasida qo'llanilishi

🌐 Global tajriba – Tibbiy AI tizimlari, millionlab rentgen, ultratovush va qon natijalari ustida ishlaydi. Har bir tahlilni tekshirish uchun ortiqcha vaqt sarflamaslik uchun — yaroqsiz yoki keraksiz holatlar avtomatik tashlab yuboriladi (continue). Bemor hayotiga xavf soladigan alomat aniqlansa — tizim tahlilni darhol to'xtatadi va shifokorni ogohlantiradi (break).

- ✓ Vaqtni tejaydi.
- ✓ Inson xatolarini kamaytiradi.
- ✓ Shoshilinch tibbiy holatlarga tezkor javob beradi.

Haqiqiy tizimlar – Google Health AI, IBM Watson Health, Babylon Health, Ada Health kabi sun'iy intellekt tizimlari bu yondashuvlardan foydalanmoqda.

Natija

Break va continue operatorlari Python tilida sikllarni samarali boshqarish vositasi bo'lib, dasturning ishlash tezligi va aniqligini oshiradi. Ular sun'iy intellekt, ta'lim va tibbiyot kabi zamonaviy sohalarda katta hajmdagi ma'lumotlar bilan ishlashda qo'llanilib, jarayonlarni soddalashtiradi. Bu operatorlar yordamida resurslar tejaladi, xatoliklar kamayadi va tizimlar inson fikrlashiga yaqinroq ishlaydi. Dasturchilar uchun bu operatorlarni chuqur tushunish – zamonaviy texnologik muammolarga samarali yechimlar yaratishning kalitidir.

Xulosa

Python dasturlash tilidagi break va continue operatorlari dasturlash jarayonini samarali boshqarishda muhim ahamiyatga ega. Ular takrorlanuvchi jarayonlarni nazorat qilish, ortiqcha yoki noto'g'ri ma'lumotlarni chetlab o'tish va kerakli paytda jarayonlarni to'xtatish imkonini beradi. Bugungi kunda bu operatorlar sun'iy intellekt, ta'lim va tibbiyot kabi ko'plab zamonaviy sohalarda keng qo'llanilmoqda. Ular yordamida tizimlar yanada tezkor, ishonchli va aqlli bo'lib, inson omilini kamaytirish orqali yuqori samaradorlikka erishmoqda.

Foydalanilgan adabiyotlar ro'yxati

1. Dorfman, R., Samuelson, P.A., & Solow, R.M. (1958). Linear Programming and Economic Analysis. McGraw-Hill.
2. Taha, H.A. (2017). Operations Research: An Introduction (10th ed.). Pearson.
3. Hillier, F.S., & Lieberman, G.J. (2015). Introduction to Operations Research (10th ed.). McGraw-Hill.
4. T.I.Nurmamatovich, N.A.Azizjon o'g'li - American Journal of Open, 2024 - sciencebulletin.com
5. Nurmamatovich, T. I. (2024). MY SQL MISOLIDA LOYIHA YARATISH. Ta'limda raqamli texnologiyalarni tadbiq etishning zamonaviy tendensiyalari va rivojlanish omillari, 31(2), 82-90.

6. Tojimamatov, I. (2024). BERILGANLARNI FIZIK MODELLASHTIRISH. *Development and innovations in science*, 3(5), 98-103.
7. Tojimamatov, I., & Abdulxayev, U. (2024). BERILGANLARNI KONSEPTUAL MODELINI YARATISH. *Current approaches and new research in modern sciences*, 3(6), 11-14.
8. Tojimamatov, I. (2024). BERILGANLARNI O'ZGARTIRISH OPERATORI. *Science and innovation in the education system*, 3(6), 74-78.
9. Nurmamatovich, T. I. (2024). BERILGANLARNING TARMOQ MODELLARI: ODDIY VA MURAKKAB TARMOQ TUZILISHLARI.
10. Tojimamatov, I., & Jo'rayeva, M. (2024). BERILGANLAR TUZILISHINING UCH DARAJALI ARXITEKTURASI. *Current approaches and new research in modern sciences*, 3(4), 194-197.
11. Тожимаматов, И. Н. (2023). Задачи Интеллектуального Анализа Данных. *Pedagog*, 6(4), 514-516.
12. Tojimamatov, I., & Mirsiddiqova, M. (2025). BERILGANLAR BAZASIDA HAYOTIY SIKL. *Модели и методы в современной науке*, 4(6), 66-70.
13. Tojimamatov, I., & Abduvaliyev, X. (2025). KO 'P FOYDALANUVCHILI BBBT ARXITEKTURASI. *Инновационные исследования в науке*, 4(5), 16-22.
14. Tojimamatov, I., & Siddiqova, G. (2025). TRANZAKSIYALARNI TAQSIMLANGAN TARZDA QAYTA ISHLASH MODELLARI. *Современные подходы и новые исследования в современной науке*, 4(6), 30-35.
15. Нурмаматович, Т. И., & Рахила, А. (2025). НА ОСНОВЕ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ ПОВЫШЕНИЕ УСТОЙЧИВОСТИ К ПОЛОМКАМ И АВАРИЯМ. *YANGI O 'ZBEKISTON, YANGI TADQIQOTLAR JURNALI*, 2(8), 197-204.
16. Тоджиматов И., Холмуродов А. О. Х. (2025, май). SQL-СЕРВЕРДА ЧЕКЛАШЛАР. В *КОНФЕРЕНЦИИ ПО СОВРЕМЕННОЙ НАУКЕ И ПЕДАГОГИКЕ* (Том 1, № 1, стр. 409-413).
17. Tojimamatov, I., & Ahmataliyeva, S. (2025). BERILGANLARNI MARKAZLASHGAN TARZDA BOSHQARISH TAMOYILLARI. *Академические исследования в современной науке*, 4(21), 59-64.
18. Tojimamatov, I., & Marifjonova, M. (2025). BERILGANLARNI MARKAZLASHGAN TARZDA BOSHQARISH PRINSIPLARI. *Академические исследования в современной науке*, 4(21), 65-70.
19. Tojimamatov, I., & Adxamova, C. (2025). AMALIY TIZIMLARDA BERILGANLAR BAZASINI BOSHQARISH TIZIMLARI O 'RNI. *Академические исследования в современной науке*, 4(21), 77-82.

20. Tojimamatov, I., & Fazliddinov, X. (2025). BERILGANLAR BAZASI ADMINISTRATORI VA UNING XUSUSYATLAR. *Академические исследования в современной науке*, 4(21), 90-95.
21. Tojimamatov, I. (2025). BERILGANLAR BAZASIDA AXBOROT ALMASHINISH SXEMASI HAMDA TURLARI. *Академические исследования в современной науке*, 4(21), 71-76.
22. Tojimamatov, I., & Zaylobiddinova, M. (2025). AUDIO MA'LUMOTLAR TAQDIMOT. *TRANSFORMING EDUCATION THROUGH SCIENTIFIC DISCOVERY*, 1(1), 38-44.
23. Tojimamatov, I., & Ismoilov, J. (2025). KRIPTOGRAFIK KALITLARNING KELAJAGI: XAVFSIZLIKNI TA'MINLASHDA YANGI YO'NALISHLAR. *BRIDGING THE GAP: EDUCATION AND SCIENCE FOR A SUSTAINABLE FUTURE*, 1(1), 56-63.
24. Tojimamamtov, I. (2025). RAQAMLI IKTISODIYETNING IJTIMOIY-IKTISODIY JARA'ENLARINI BOShQARIsh VA REJALAShtAGI MOxIYATI. *DIGITAL TRANSFORMATION AND ARTIFICIAL INTELLIGENCE*, 3(1), 229-233.
25. Tojmamatov, I., Sharofutdinov, I., & Saydolimova, G. (2025). BILIMLAR OMBORI. BILIMLAR OMBORINING TUZILISHI. *EDUCATION AND RESEARCH IN THE ERA OF DIGITAL TRANSFORMATION*, 1(1), 431-437.
26. Tojimamatov, I., & Metinboyeva, F. (2025). TESKARI MUNOSABATDA MUNOSABATLAR USTIDAGI AMALLAR. *BRIDGING THE GAP: EDUCATION AND SCIENCE FOR A SUSTAINABLE FUTURE*, 1(1), 202-214.