3-dars.

Tarmoqlanish operatorlari. Mantiqiy operatorlar. Or, and, not operatorlari.

If,else,elif operatorlari.

## Python operatorlari

Operatorlar o'zgaruvchilar va qiymatlar bo'yicha operatsiyalarni bajarish uchun ishlatiladi.

Quyidagi misolda biz +ikkita qiymatni qo'shish uchun operatordan foydalanamiz :

Python operatorlarni quyidagi guruhlarga ajratadi:

* Arifmetik operatorlar
* Topshiriq operatorlari
* Taqqoslash operatorlari
* Mantiqiy operatorlar
* Identifikatsiya operatorlari
* A'zolik operatorlari
* Bitwise operatorlari

## Python arifmetik operatorlari

Arifmetik operatorlar umumiy matematik amallarni bajarish uchun raqamli qiymatlar bilan ishlatiladi:

|  |  |  |
| --- | --- | --- |
| **Operator** | **Nomi** | **Misol** |
| + | Addition | x + y |
| - | Subtraction | x - y |
| \* | Multiplication | x \* y |
| / | Division | x / y |
| % | Modulus | x % y |
| \*\* | Exponentiation | x \*\* y |
| // | Floor division | x // y |

## Python tayinlash operatorlari

Topshiriq operatorlari o'zgaruvchilarga qiymatlarni berish uchun ishlatiladi:

|  |  |  |
| --- | --- | --- |
| **Operator** | **Misol** | **Xuddi shundek** |
| = | x = 5 | x = 5 |
| += | x += 3 | x = x + 3 |
| -= | x -= 3 | x = x - 3 |
| \*= | x \*= 3 | x = x \* 3 |
| /= | x /= 3 | x = x / 3 |
| %= | x %= 3 | x = x % 3 |
| //= | x //= 3 | x = x // 3 |
| \*\*= | x \*\*= 3 | x = x \*\* 3 |
| &= | x &= 3 | x = x & 3 |
| |= | x |= 3 | x = x | 3 |
| ^= | x ^= 3 | x = x ^ 3 |
| >>= | x >>= 3 | x = x >> 3 |
| <<= | x <<= 3 | x = x << 3 |

REKLAMA

## Pythonni taqqoslash operatorlari

Taqqoslash operatorlari ikkita qiymatni solishtirish uchun ishlatiladi:

|  |  |  |
| --- | --- | --- |
| **Operator** | **Nomi** | **Misol** |
| == | Equal | x == y |
| != | Not equal | x != y |
| > | Greater than | x > y |
| < | Less than | x < y |
| >= | Greater than or equal to | x >= y |
| <= | Less than or equal to | x <= y |

## Python mantiqiy operatorlari

Mantiqiy operatorlar shartli gaplarni birlashtirish uchun ishlatiladi:

|  |  |  |
| --- | --- | --- |
| **Operator** | **Tasnifi** | **Misol** |
| and | Returns True if both statements are true | x < 5 and  x < 10 |
| or | Returns True if one of the statements is true | x < 5 or x < 4 |
| not | Reverse the result, returns False if the result is true | not(x < 5 and x < 10) |

## Python identifikatori operatorlari

Identifikatsiya operatorlari ob'ektlarni taqqoslash uchun ishlatiladi, agar ular teng bo'lsa emas, balki ular aslida bir xil ob'ekt bo'lsa va xotira joyi bir xil bo'lsa:

|  |  |  |
| --- | --- | --- |
| **Operator** | **Tasnifi** | **Misol** |
| is | Returns True if both variables are the same object | x is y |
| is not | Returns True if both variables are not the same object | x is not y |

## Pythonga a'zolik operatorlari

A'zolik operatorlari ob'ektda ketma-ketlik mavjudligini tekshirish uchun ishlatiladi:

|  |  |  |
| --- | --- | --- |
| **Operator** | **Description** | **Example** |
| in | Returns True if a sequence with the specified value is present in the object | x in y |
| not in | Returns True if a sequence with the specified value is not present in the object | x not in y |

## Python Bitwise operatorlari

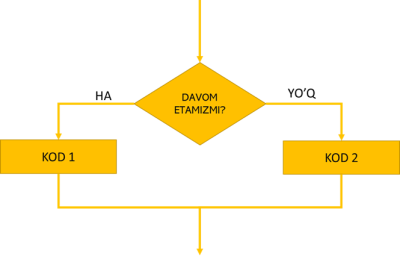
Bitwise operatorlari raqamlarni solishtirish uchun ishlatiladi (ikkilik):

|  |  |  |
| --- | --- | --- |
| **Operator** | **Name** | **Description** |
| & | AND | Sets each bit to 1 if both bits are 1 |
| | | OR | Sets each bit to 1 if one of two bits is 1 |
| ^ | XOR | Sets each bit to 1 if only one of two bits is 1 |
| ~ | NOT | Inverts all the bits |
| << | Zero fill left shift | Shift left by pushing zeros in from the right and let the leftmost bits fall off |
| >> | Signed right shift | Shift right by pushing copies of the leftmost bit in from the left, and let the rightmost bits fall off |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1.7. Shatr ifodalari** Shart ifodalarini bir qator amallar taqdim qiladi. Ushbu amallarning barchasi ikkita operand qabul qiladi va natija sifatida *boolean* turidagi mantiqiy qiymat qaytaradi. Faqatgina ikkita mantiqiy qiymat mavjud, ular *True* (ifoda rost) *False* (ifoda yolg‟on) qiymatlardir. **Taqqoslash amallari.** Eng sodda shart ifodalariga taqqoslash amallari misol bo‟lib, ular ikki qiymatni taqqoslash uchun ishlatiladi. Python quyidagi taqqoslash amallarini qo‟llab-quvvatlaydi: **==** - agar ikki operand teng bo‟lsa *True*, aks holda *False* qiymat qaytaradi; **!=** - agar ikki operand teng bo‟lmasa *True*, aks holda *False* qiymat qaytaradi; > (dan katta) – agar birinchi operand ikkinchisidan katta bo‟lsa *True*, aks holda *False* qiymat qaytaradi; < (dan kichik) – agar birinchi operand ikkinchisida kichik bo‟lsa True, aks holda False qiymat qaytaradi; >= (dan katta yoki teng) – agar birinchi operand ikkinchisidan katta yoki teng bo‟lsa *True*, aks holda *False* qiymat qaytaradi; <= (dan kichik yoki teng) – agar birinchi operand ikkinchisidan kichik yoki teng bo‟lsa *True*, aks holda *False* qiymat qaytaradi; **1.8. Mantiqiy amallar** Murakkab shartli ifodalarni yozish, odatda mantiqiy amallar yordamida amalga oshiriladi. Pythonda quyidagi mantiqiy operatorlar mavjud: ***and*** (mantiqiy ko‟paytirish). Murakkab ifodadagi biror bir qism ifodani qiymati *False* bo‟lsa, ifodaning yakuniy qiymati *False* , aks holda *True* qiymat qaytaradi. Masalan:  yoshi = 21 vazni = 72 natija = yoshi > 17 **and** vazni == 72 print(natija) *# True*  Yuqoridagi dasturda murakkab mantiqiy amal ikki qismdan  yoshi > 17 va vazni > 56  qismlardan tashkil topgan bo‟lib, ular ***and***mantiqiy operatori bilan birlashtirilgan. Agarda ikkala mantiqiy amal *True* qiymat qaytarsa ifodaning qiymati *True* bo‟ladi, aks holda *False* qiymat qaytaradi. Matriqiy ifodalarda faqatgina taqqoslash amallaridan foydalanish shart emas. Ixtiyoriy mantiqiy amal yoki *boolean* turidagi qiymatlar (*True*, *False*) ham ishlatilishi mumkin. Masalan:   |  |  | | --- | --- | | 1 2 3 4 5 | yoshi = 21 vazni = 72 t = **True** natija = yoshi > 17 **and** vazni > 56 **and** t print(natija) *# True* |   ***or*** (mantiqiy qo‟shish). Agarda ifodadagi biror bir qism ifoda *True* qiymat qaytarsa, yakuniy natija ham *True*, aks holda *False* bo‟ladi.   |  |  | | --- | --- | | 1 2 3 4 | yoshi = 21 t = **False** natija = yoshi > 17 **or** t print(natija) *# True* |   18 ***not*** (mantiqiy inkor). Ifodaning qiymatini *True* bo‟lsa, natija *False* va aksincha.   |  |  | | --- | --- | | 1 2 3 4 | yoshi = 21 t = **False** print(**not** yoshi > 17) *# False* print(**not** t) *# True* |   *and* operatorining biror bir operandi *False* qiymatga ega bo‟lsa, u holda boshqa operand qiymati tekshirib (hisoblanib) o‟tirilmaydi, har doim natija *False* bo‟ladi. Bunday xususiyat ish unumdorligini bir oz bo‟lsada oshirish imkonini beradi. Xuddi shunaqa xususiyat *or* operatori uchun ham o‟rinli. Ya`ni *or* operatorining biror bir operandi qiymati *True* qiymatga ega bo‟lsa, boshqa operandlar tekshirilmaydi, natija sifatida har doim *True* qiymati qaytariladi. Agar bitta ifodada bir nechta mantiqiy operatorlar qatnashgan bo‟lsa, u holda ularning ustunligiga (prioritetiga) alohida e`tibor qatarish kerak. Dastlab *not* operatori keyin *and* va eng so‟ngra *or* operatori bajariladi. Masalan:   |  |  | | --- | --- | | 1 2 3 4 5 | yoshi = 22 xolati = **False** vazni = 58 natija = vazni == 58 **or** xolati **and not** yoshi > 21 *# True* print(natija) |   Ushbu dasturda keltirilgan ifodadagi mantiqiy amallar kuyidagi ketma-ketlikda bajariladi: *1. not yoshi > 21* mantiqiy ifoda *False* qiymat qaytaradi; *2. xolati and False* (*not yoshi > 21*) esa *False* qiymat qaytaradi; *3. vazni == 58 or True* (*xolati and not yoshi > 21*) esa *True* qiymat qaytaradi. Shuni alohida ta`kidlash kerarki, mantiqiy ifodalarda mantiqiy amallarning bajarilish ketma-ketligini qavslar (, ) yordamida o‟zgartirish mumkin. |
|  |
|  |
|  |

**TARMOQLANISH**

Shu vaqtgacha yozgan dasturlarimizga e'tibor bersangiz, dasturimiz yuqoridan pastga qarab qatorma-qator bajarilib keldi. Bu chiziqli dastur deyiladi. Voqelikda esa aksar dasturlar ma'lum bir shart bajarilishi (yoki bajarilmaganiga) ko'ra kodning bir qismidan boshqa qismiga "sakrab" o'tishi tabiiy hol. Dasturlashda bu ***tarmoqlanish*** deb ataladi.



**Tarmoqlanishga misol**

Ushbu darsimizda biz **if** operatori yordamida shunday shartlarni yozishni, tekshirishni va tekshiruv natijasiga ko'ra kodning turli qismlarini bajarishni o'rganamiz.

**if OPERATORI**

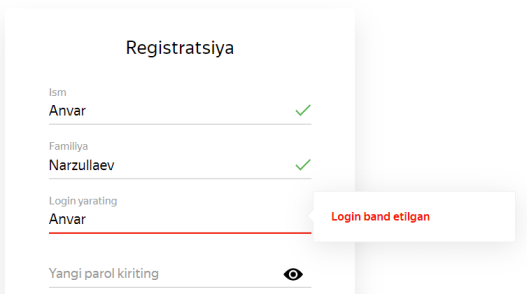
**if** so'zi ingliz tilidan **"agar"** deb tarjima qilinadi va deyarli barcha dasturlash tillarida shartlarni yozish uchun foydalaniladi.

Keling quyidagi misolni ko'ramiz.

# MATNLARNI SOLISHTIRISH

Aksar tizimlar foydalanuvchi kiritgan matnni ma'lum bir ko'rinishga keltirib oladi. Buning sababi, kompyuter uchun 'Ali', 'ALI', va 'ali' bu uchta turli hil ism. Ularni solishtirish uchun esa bir ko'rinishga keltirib olish kerak.

Tasavvur qiling siz yangi email manzil ochmoqchisiz, va o'zingizga yangi foydalanuvchi ismini tanlashingiz kerak. Kompyuter siz kiritgan foydalanuvchi ismini tizimdagi mavjud foydalanuvchilar bilan solishtiradi va agar ism band bo'lsa sizga boshqa ism tanlashni aytadi. Solishtirish jarayonida esa, siz tanlagan ismni kichik harflarga o'tkazib, boshqa ismlar bilan solishtiradi.



Yandex.uz sahifasida ro'yxatdan o'tish jarayoni

Yuqoridagi misolda, kimdur anvar@yandex.ru manzilini band qilgan, agarda men 'Anvar', yoki 'ANvar', yoki 'ANVAR' deb login tanlasam ham, anvar@yandex.ru band bo'gani sababli men so'ragan loginlar rad qilinaveradi.

Xo'sh, turli ko'rinishda yozilgan matnlarni qanday qilib solishtiramiz? Juda oddiy. Matnlarni solishtirishdan avval .lower() metodi yordamida kichik harflar ko'rinishiga keltirib olamiz:

ism = 'Ali'

ism.lower() == 'ali'

Natija: True

# QIYMATLARNING TENG EMASLIGINI TEKSHIRISH

Agar ikki qiymatning teng emasligini tekshirish talab qilinsa, != operatoridan foydalanilamiz.

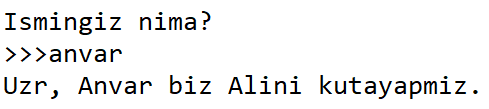
ism = input('Ismingiz nima?\n>>>') # Foydalanuvchi ismini so'raymiz

if ism.lower() != 'ali': # Agar ism Aliga teng bo'lmasa ...

print(f"Uzr, {ism.title()} biz Alini kutayapmiz.") # quyidagi xabar chiqadi

else:

print("Salom, Ali")



Natija

Demak yuqoridagi kodning 2-qatorida ism ichidagi qiymat 'ali' ga teng bo'lmasa "Uzr, {ism} biz Alini kutyapmiz" degan xabarni chiqar dedik. Aks holda (else), "Salom, Ali" degan xabar chiqadi.

Shartlarda else qismi bo'lishi majburiy emas. Bunga keyingi bo'limlarda tushunarliroq misollar ko'ramiz.

# SONLARNI SOLISHTIRISH

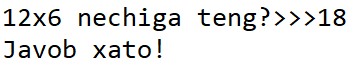
Sonlarni solishtirishda yuqoridagi teng (==) va teng emas (!=) shartlariga qo'shimcha ravishda quyidagi mantiqiy shartlar ham qo'shiladi:

* Kichik: a<b
* Kichik yoki teng: a<=b
* Katta: a>b
* Katta yoki teng: a>=b

javob = float(input("12x6 nechiga teng?>>>"))

if javob!=72:

print("Javob xato!")



Natija

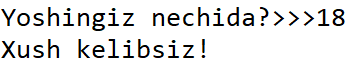
yosh = int(input("Yoshingiz nechida?>>>"))

if yosh>=18: # yosh 18 dan katta yoki teng bo'lsa

print('Xush kelibsiz!')

else: # ask holda

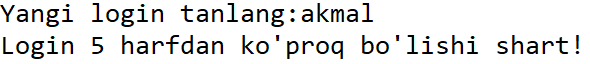
print('Kirish mumkin emas!')



login = input("Yangi login tanlang:")

if len(login)<=5: # login uzunligini tekshiramiz

print("Login 5 harfdan ko'proq bo'lishi shart!")



Natija

Sonlarni solishtirishda arifmetik ifodalar ham yozishimiz mumkin:

yil = int(input("Tug'ilgan yilingizni kiriting:"))

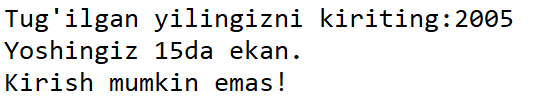
if 2020-yil<18: # foydalanuvchining yoshini hisoblaymiz

print(f"Yoshingiz {2020-yil}da ekan.")

print("Kirish mumkin emas!")

else:

print("Xush kelibsiz!")



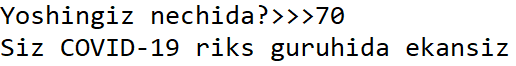
Natija

# BIR QATOR if/else

Qisqa kodlar uchun shart va uning badanini 1 qatorga jamlab yozishimiz ham mumkin:

yosh = int(input("Yoshingiz nechida?>>>"))

if yosh>65: print("Siz COVID-19 risk guruhida ekansiz")



Natija

Yoki:

x, y = 25, 50 # x=25 va y=50

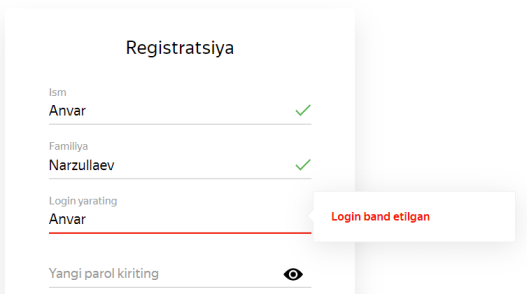
print("x>y") if x>y else print("x<y")

Natija: x<y

**MATNLARNI SOLISHTIRISH**

Aksar tizimlar foydalanuvchi kiritgan matnni ma'lum bir ko'rinishga keltirib oladi. Buning sababi, kompyuter uchun 'Ali', 'ALI', va 'ali' bu uchta turli hil ism. Ularni solishtirish uchun esa bir ko'rinishga keltirib olish kerak.

Tasavvur qiling siz yangi email manzil ochmoqchisiz, va o'zingizga yangi foydalanuvchi ismini tanlashingiz kerak. Kompyuter siz kiritgan foydalanuvchi ismini tizimdagi mavjud foydalanuvchilar bilan solishtiradi va agar ism band bo'lsa sizga boshqa ism tanlashni aytadi. Solishtirish jarayonida esa, siz tanlagan ismni kichik harflarga o'tkazib, boshqa ismlar bilan solishtiradi.



Yandex.uz sahifasida ro'yxatdan o'tish jarayoni

Yuqoridagi misolda, kimdur anvar@yandex.ru manzilini band qilgan, agarda men 'Anvar', yoki 'ANvar', yoki 'ANVAR' deb login tanlasam ham, anvar@yandex.ru band bo'gani sababli men so'ragan loginlar rad qilinaveradi.

Xo'sh, turli ko'rinishda yozilgan matnlarni qanday qilib solishtiramiz? Juda oddiy. Matnlarni solishtirishdan avval .lower() metodi yordamida kichik harflar ko'rinishiga keltirib olamiz:

ism = 'Ali'

ism.lower() == 'ali'

Natija: True

**QIYMATLARNING TENG EMASLIGINI TEKSHIRISH**

Agar ikki qiymatning teng emasligini tekshirish talab qilinsa, != operatoridan foydalanilamiz.

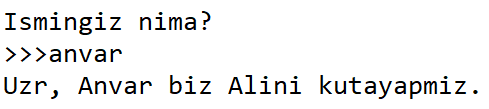
ism = input('Ismingiz nima?\n>>>') # Foydalanuvchi ismini so'raymiz

if ism.lower() != 'ali': # Agar ism Aliga teng bo'lmasa ...

print(f"Uzr, {ism.title()} biz Alini kutayapmiz.") # quyidagi xabar chiqadi

else:

print("Salom, Ali")



Natija

Demak yuqoridagi kodning 2-qatorida ism ichidagi qiymat 'ali' ga teng bo'lmasa *"Uzr, {ism} biz Alini kutyapmiz"* degan xabarni chiqar dedik. Aks holda (else), "Salom, Ali" degan xabar chiqadi.

Shartlarda else qismi bo'lishi majburiy emas. Bunga keyingi bo'limlarda tushunarliroq misollar ko'ramiz.

**SONLARNI SOLISHTIRISH**

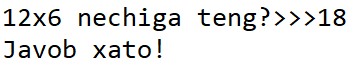
Sonlarni solishtirishda yuqoridagi teng (==) va teng emas (!=) shartlariga qo'shimcha ravishda quyidagi mantiqiy shartlar ham qo'shiladi:

* Kichik: a<b
* Kichik yoki teng: a<=b
* Katta: a>b
* Katta yoki teng: a>=b

javob = float(input("12x6 nechiga teng?>>>"))

if javob!=72:

print("Javob xato!")



Natija

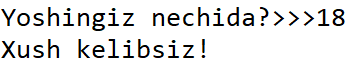
yosh = int(input("Yoshingiz nechida?>>>"))

if yosh>=18: # yosh 18 dan katta yoki teng bo'lsa

print('Xush kelibsiz!')

else: # ask holda

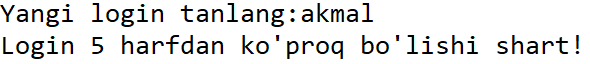
print('Kirish mumkin emas!')



login = input("Yangi login tanlang:")

if len(login)<=5: # login uzunligini tekshiramiz

print("Login 5 harfdan ko'proq bo'lishi shart!")



Natija

Sonlarni solishtirishda arifmetik ifodalar ham yozishimiz mumkin:

yil = int(input("Tug'ilgan yilingizni kiriting:"))

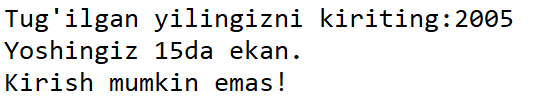
if 2020-yil<18: # foydalanuvchining yoshini hisoblaymiz

print(f"Yoshingiz {2020-yil}da ekan.")

print("Kirish mumkin emas!")

else:

print("Xush kelibsiz!")



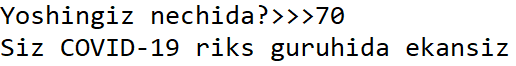
Natija

**BIR QATOR if/else**

Qisqa kodlar uchun shart va uning badanini 1 qatorga jamlab yozishimiz ham mumkin:

yosh = int(input("Yoshingiz nechida?>>>"))

if yosh>65: print("Siz COVID-19 risk guruhida ekansiz")



Natija

Yoki:

x, y = 25, 50 # x=25 va y=50

print("x>y") if x>y else print("x<y")

Natija: x<y

a = 200

b = 33

if b > a:

print("b is greater than a")

elif a == b:

print("a and b are equal")

else:

print("a is greater than b")