

- Бұл Питон тілінің ең жиі қолданылатын функциясы

- Тілдің құрамдас ішкі (стандартты) функциясы

- `print "Some Text"` – python 2.7.*
- `print("Another Text")` – python 3.6.*

- Аргументтері:

- `print(*object, sep="", end='\n', file=sys.stdout, flush=False)`

- Мысал:

```
print("Кім маған ", " 1000 теңге бере алар екен", sep="дәл қазір", end="?!")
```

Соңғы жол нәтижесі:

Кім маған дәл қазір 1000 теңге бере алар екен?!

Мәлімет шығару форматтары

- Сіз шығарылатын мәліметті дұрыстап рәсімдей аласыз.
- Оны қалай істеуге болар екен?
- Оның бірнеше тәсілі бар:
- 1-мысал:

```
name = "Армат"
```

```
print("Танысайық, менің атым - %s" % name)
```

Нәтижесі

```
... /PycharmProjects/bb/venv/btanis.py  
Танысайық, менің атым - Армат
```

```
Process finished with exit code 0
```

Мәлімет шығару форматтары

- 2-мысал:

```
name = "Армат"
```

```
surname = "Серікұлы"
```

```
print("Менің толық аты-жөнім - %s %s" % (surname, name))
```

Менің толық аты-жөнім – Серікұлы Армат

- 3-мысал:

```
a = 10
```

```
b = 20
```

```
print("%i және %i қосындысы - " % (a,b), a+b)
```

10 және 20 қосындысы - 30

Мәлімет шығару форматтары

- 4-мысал:

- **.format(...)** әдіс

Оқу сендерге болашақта пайдалы бола ма?

```
print("Оқу {1} болашақта {0} бола ма?".format("пайдалы ", "сендерге"))
```

- Тапсырма: **Кітаптардан .format()** туралы толық оқып, келесі сабақта айтып бересіңдер.

- Форматтардағы % таңбасы жайлы берілген түсініктер түйіні

- i – int # бүтін сан
 - d – signed int decimal # таңбалы ондық сан
 - c – character # таңба, символ
 - f – float # нақты, жылжымалы нүктелі сан
 - s – string # сөз тіркесі (строка - тіркес)

% таңбасы арқылы түрлендіру кестесі

Түрлендіру таңбасы	Түсініктемесі
d	Таңбалы ондық сан
i	Таңбалы ондық сан
o	Таңбасыз сегіздік сан
u	'd' таңбасының бұрынғы ескі эквиваленті, яғни таңбалы ондық сан
x	Таңбасыз он алтылық сан (кіші әріптер, төменгі регистр)
X	Таңбасыз он алтылық сан (бас әріптер, жоғарғы регистр)
e	Жылжымалы үтірлі экспоненциал формат (кіші әріптер)
E	Жылжымалы үтірлі экспоненциал формат (бас әріптер)
f	Жылжымалы нүктелі ондық сан форматы
F	Жылжымалы нүктелі ондық сан форматы
g	Егер дәреже -4-тен артық болса немесе дәлдігі төмен болса, «e» сияқты, әйтпесе «f» форматы секілді
G	Егер дәреже -4-тен артық болса немесе дәлдігі төмен болса, «E» сияқты, әйтпесе «F» форматы секілді
c	Бір символ (бүтін санды немесе символдардың бір жолын қабылдайды
r	repr () -ді пайдаланып, кез келген объектіні түрлендіреді
s	str () -ді пайдаланып, кез келген объектіні түрлендіреді
%	Ешқандай аргумент түрлендірілмейді, нәтижесінде «%» символы алынады

Мәліметтер арасын ажырату символдары

- 5-мысал: жай бос орын таңбасымен бөлу
`print('Бір', 'Екі', 'Үш')`
- 6-мысал: нүктелі үтір (;) таңбасымен бөлу
`print('Бір', 'Екі', 'Үш', sep=';')`
- 7-мысал: басқаша (~) таңбалармен бөлу
`print('Бір', 'Екі', 'Үш', sep='~')`
- 8-мысал: көлденең табуляция (\t) арқылы бөлу
`print('Бір\tЕкі\tҮш')`
- 9-мысал: келесі жолдарға көшіру (\n)
`print('Бір\nЕкі\nҮш')`

Нәтижелер:

Бір Екі Үш

Бір;Екі;Үш

Бір~Екі~Үш

Бір Екі Үш

Бір
Екі
Үш

Сандарды форматтап шығару

- Форматсыз шығару

```
jp = 5450000.0  
print('Жылдық пайда:',jp)  
ap = jp/12.0  
print('Айлық пайда:',ap)
```

Нәтижесі:

Жылдық пайда: 5450000.0

Айлық пайда: 454166.6666666667

- Форматпен шығару, '2f' – формат спецификаторы : бөлшегі 2 цифр

```
a = 54321.6789  
print(format(a, '.2f'))  
print(format(a, '.1f'))
```

54321.68

54321.7

- Форматпен шығару, 'e' – формат спецификаторы : экспоненциалды сан

```
print(format(54321.6789, 'e'))  
print(format(54321.6789, '.2E'))
```

5.432168e+04

5.43E+04

Сандарды форматтап шығару

- Форматпен валютаны шығару, бүтін бөліктің әрбір 3 цифрын үтірмен бөліп жазу

```
at = 3500.0 # айлық табыс  
jt = at*12  # жылдық табыс
```

```
print('Сіздің жылдық табысыңыз $',  
      format(jt, ',.2f'), sep='')
```

Сіздің жылдық табысыңыз \$42,000.00

- Форматпен шығару, '10.2f' – формат спецификаторы : ені – 10, бөлшегі 2 цифр

```
a = 54321.6789  
print('a санының мәні = ',  
      format(a, '10.2f'))
```

a санының мәні = 54321.68

- Формат арқылы бүтін санның енін (14) беріп, әрбір 3 цифрды үтірмен бөліп жазу

```
print(format(123456789, '14,d'))
```

123,456,789

Сандарды форматтап шығару

- Келесі программада санды валюта түрінде шығару көрсетілген.

Бөлшегі бар нақты санды шығару.

```
amount_due = 5000.0
```

```
monthly_payment = amount_due / 12
```

```
print('Әрбір айлық төлем = ', format(monthly_payment, '.2f'))
```

- Программа нәтижесі:

Әрбір айлық төлем = 416.67

Сандарды форматтап шығару

- Келесі программада санды валюта түрінде шығару көрсетілген.

Бөлшегі бар нақты санды шығару.

```
amount_due = 5000.0
```

```
monthly_payment = amount_due / 12
```

```
print('Әрбір айлық төлем = ', format(monthly_payment, '.2f'))
```

- Программа нәтижесі:

Нәтижесі:

Әрбір айлық төлем = 416.67

Сандарды форматтап шығару

- Келесі программада санды валюта түрінде шығару көрсетілген.

```
# Нақты санды валюта түрінде шығару  
monthly_pay = 30000.0  
annual_pay = monthly_pay * 12  
print('Сіздің жылдық табысыңыз $',  
      format(annual_pay, ',.2f'), sep=' ')
```

Нәтижесі:

Сіздің жылдық табысыңыз \$360,000.00

Сан өрісі енін көрсету

```
>>> print('Сан = ', format(12345.6789,'12,.2f')) Enter
```

Нәтижесі:
Сан = 12,345.68

```
>>> print('Сан = ', format(12345.6789,'12.2f')) Enter
```

Нәтижесі:
Сан = 12345.68

Бүтін сан енін көрсету

- `format` функциясын бүтін сандарға да қолдана аламыз. Мұнда екі түрлі мүмкіндік бар:
 - тип көрсеткіші ретінде `d` символы қолданылады:
 - дәлдік көрсетілмейді.
- Төменде 123456 саны форматсыз шығарылған:

```
>>> print(format(123456, 'd'))
```

Нәтижесі: 123456
- Келесі жолда 123456 үтір арқылы ажыратылып жазылады:

```
>>> print(format(123456, ',d'))
```

Нәтижесі: 123,456
- Төменде санның ені 10 орынмен берілген:

```
>>> print(format(123456, '10,d'))
```

Нәтижесі: 123,456

Пайыз (%) таңбасын шығару

- Пайыз таңбасын шығару үшін санның типін көрсететін f символы орнына % таңбасын қою керек:

```
>>> print(format(0.5, '%'))
```

Нәтижесі: 50.000000%

- Егер бүтін сан түрінде бергіміз келсе, дәлдігін 0 етіп береміз:

```
>>> print(format(0.5, '.0%'))
```

Нәтижесі: 50%

```
>>>
```

If-else нұсқаулары (шартты өрнектер)

- **if-else** нұсқауларының ішкі операторларын (блоктары) шегініспен жазу керек, ол үшін мыналарды есте сақтау қажет:
 - **if - else** нұсқаулары шегініссіз тураланып, бір деңгейде жазылады;
 - **if** және **else** нұсқауларынан кейін тұратын блоктар (ішкі операторлар) шегініспен жазылады. Блоктардың сол жақ шеттен бірдей шегініспен (3-5 орын) жазылуын мұқият қадағалау керек (тек босорын не не тек **Tab**).Мысалы:

```
temperatura = -3
if temperatura < 5:
    print('Күн салқын, киініп алыңыз.')
else:
    print('Күн жақсы, қыдыруға болады.')
```

Нәтижесі: Күн салқын, киініп алыңыз.

if-else нұсқаулары (шартты өрнектер)

- Келесі код фрагменті **!=** операторын пайдаланып, **month** айнымалысы көрсетіп тұрған мәнді **'Қаңтар'** тексереді де, нәтиже береді:

```
month = 'Қараша'
```

```
if month != 'Қаңтар':
```

```
    print('Жаңа жыл мерекесі тойланбайды!')
```

```
    print('Шырша да безендірілмейді!')
```

```
else:
```

```
    print('Жаңа жыл мерекесін қарсы аламыз !')
```

```
    print('Би билейміз, ән саламыз!')
```

Жаңа жыл мерекесі тойланбайды!
Шырша да безендірілмейді!

if-else нұсқаулары (шартты өрнектер)

- Программалаудағы **else** бөлігі болмайтын сәттер туралы білетін боларсыздар
- **if** қысқаша нұсқауы (else бөлігі жоқ) және оның мысалы:
- 1-мысал:

```
a = 5
```

```
if a < 10:
```

```
    print ("{0} саны {1} санынан кіші".format("a",10))
```

Нәтижесі:

a саны 10 санынан кіші

if-else нұсқаулары (шартты өрнектер)

- 2-мысал: Қабаттасқан (біріне бірі кірістірілген) if операторы
- Мұнда келесі if алдыңғы if-тен оңға қарай шегініп жазылады

```
a = 5
```

```
b = 6
```

```
c = 100
```

```
if a>4:
```

```
    if b>a:
```

```
        if c>b:
```

```
            print("%2d < %2d < %2d" % (a,b,c))
```

Нәтижесі:

5 < 6 < 100

if-else нұсқаулары (шартты өрнектер)

- If + else нұсқаулары: Сіз мұнда "else" нұсқауын тек "if" нұсқауынан кейін қоя аласыз

- 3-мысал

```
a=71
```

```
if a>50:
```

```
    print("Сіз %d бағасын алдыңыз" %a)
```

```
else:
```

```
    print("Сіз жазғы семестрге қалдыңыз")
```

Нәтижесі:

Сіз 71 бағасын алдыңыз

if-else нұсқаулары (шартты өрнектер)

- If + elif + else нұсқаулары: elif C#, Java тілдеріндегі else-if нұсқауы сияқты

- 4-мысал

a=50

b=85

if a>=50:

print("Сіз %d бағасымен өттіңіз" %a)

elif b>=80 and b<=100:

print("Емтиханнан өзі өтіпті, бірақ досының бағасы {} болыпты".format(a))

else:

print("Басқалар да мықты емес, бірақ Сіз жазғы семестрге қалдыңыз")

Нәтижесі (a=50 , b=85):

Сіз 50 бағасымен өттіңіз

Нәтижесі (a=49, b=85):

Емтиханнан өзі өтіпті, бірақ досының бағасы 49 болыпты

if-else нұсқаулары (шартты өрнектер)

- Күрделі if нұсқаулар: бірнеше шарттар бар

- 5-мысал

a=50

b=70

if a>=50 and b<75:

print("Сіздің бағаңыз %d мен %d аралығында" % (a, b))

else:

print("Сіз жазғы семестрге қалдыңыз")

Нәтижесі (a=42):

Сіз жазғы семестрге қалдыңыз

Нәтижесі (a=50):

Сіздің бағаңыз 50 мен 75 аралығында

if-else нұсқаулары (шартты өрнектер)

- Күрделі if нұсқаулар: бірнеше шарттар бар
- 6-мысал. x айнымалысының таңбасын анықтау:

$\text{sign}(x) =$ 1, егер $x > 0$
0, егер $x = 0$
-1, егер $x < 0$

```
x = 5
if x > 0:
    print (1)
elif x < 0:
    print (-1)
else:
    print (0)
```

```
x = -3
if x > 0:
    print (1)
elif x < 0:
    print (-1)
else:
    print (0)
```

```
x = 0
if x > 0:
    print (1)
elif x < 0:
    print (-1)
else:
    print (0)
```

Н а т и ж е л е р і

1

-1

0

if-else нұсқаулары (шартты өрнектер)

- **7-мысал** Күрделі if нұсқаулар: бірнеше шарттар бар

```
print ("Сәлем!")  
tauar1 = 5000  
tauar2 = 4225  
if tauar1+ tauar2 > 10000 :  
    print ("Ақша жетпейді")  
else:  
    print ("Ақша төленді")  
print ("Көріскенше, күн жақсы...")
```

Сәлем!
Ақша төленді
Көріскенше, күн жақсы...

```
print ("Сәлем!")  
tauar1 = 5500  
tauar2 = 6225  
if tauar1+ tauar2 > 10000 :  
    print ("Ақша жетпейді")  
else:  
    print ("Ақша төленді")  
print ("Көріскенше, күн жақсы...")
```

Сәлем!
Ақша жетпейді
Көріскенше, күн жақсы...

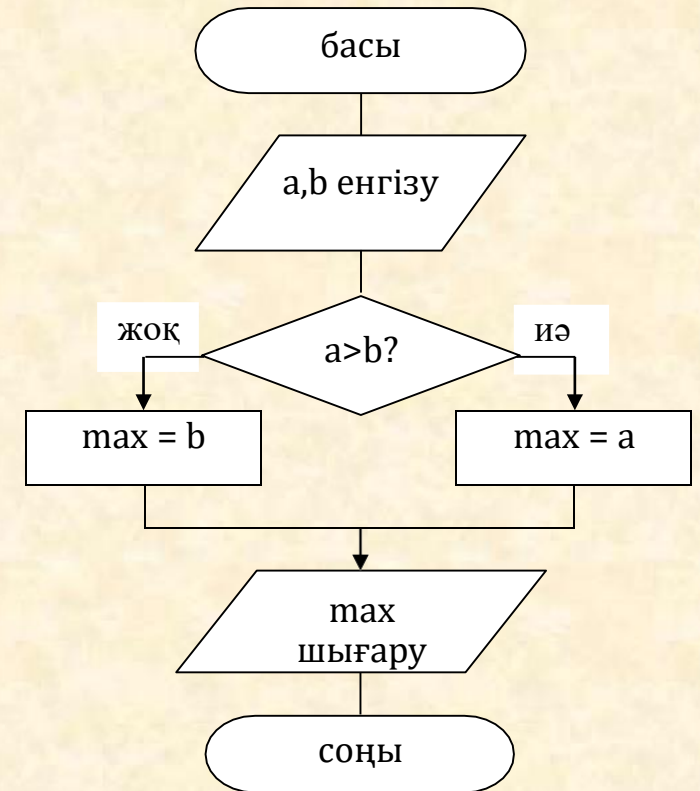
if-else нұсқаулары (шартты өрнектер)

```
a=71; b=115
if a>b:
    print("max =",a)
else:
    print("max =",b)
```

Нәтижесі:
max = 115

```
a=71; b=15
if a>b:
    print("max =",a)
else:
    print("max =",b)
```

Нәтижесі:
max = 71



Екі санның үлкенін
табу алгоритмі

if-else нұсқаулары (шартты өрнектер)

- Нақты a, b, c сандары берілген. Солардың $\max$ -ын анықтайтын есептің алгоритм схемасы мен программасын құру керек.

$a = 5.7; b = 8.12; c = -3.25$

$\max = a;$

if($b > \max$):

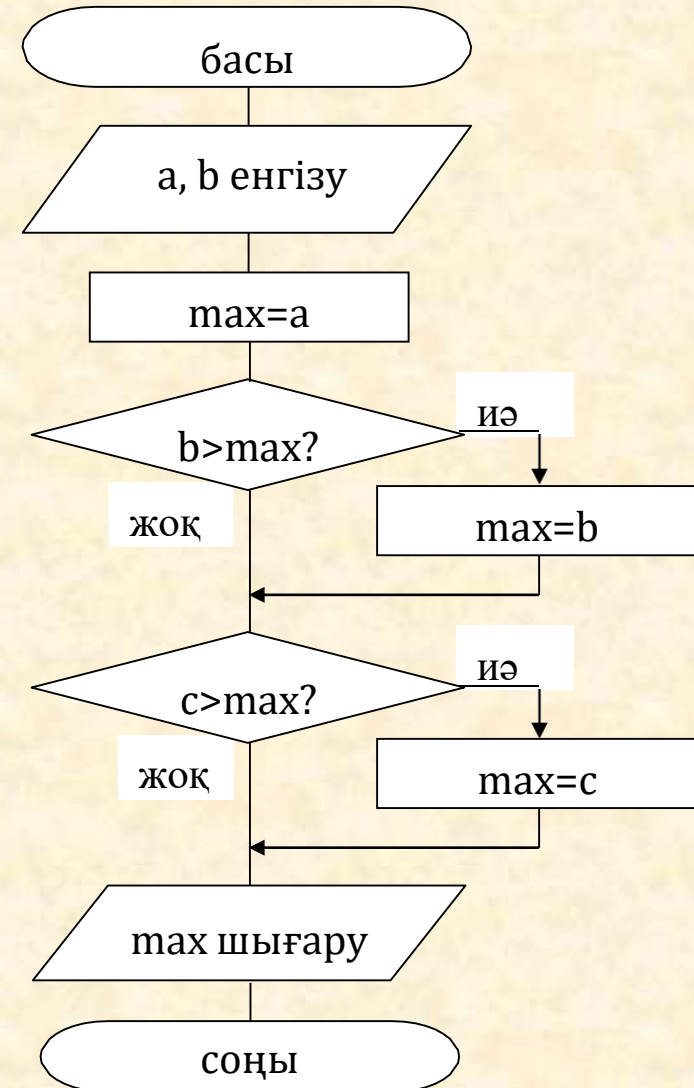
$\max = b;$

if($c > \max$):

$\max = c;$

print("max =", $\max$)

Нәтижесі:
max = 8.12



if-else нұсқаулары (шартты өрнектер)

Келесі есепті қарастырайық: $y = \begin{cases} \sin\sqrt{x}, & \text{егер } x \geq \pi \\ x \cdot e^{x+1}, & \text{егер } x < \pi \end{cases}$

```
import math          # math кітапханасы іске қосылады
```

```
x = 2.78
```

```
if x >= math.pi:    # math.pi = 3,1415926
```

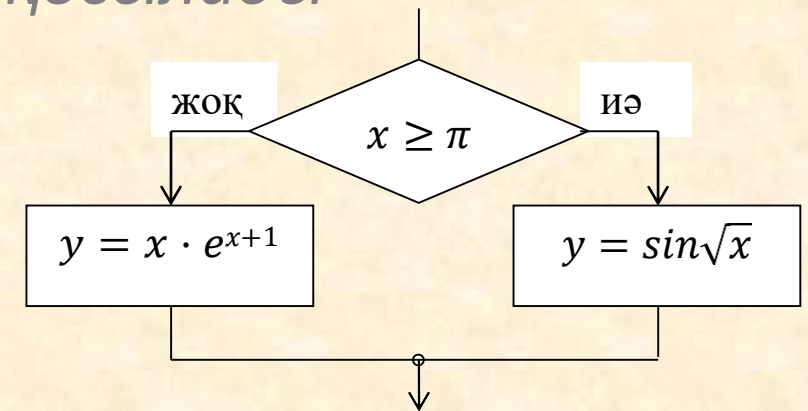
```
    y = math.sin(math.sqrt(x));
```

```
else
```

```
    y = x*math.exp(x+1);
```

Нәтижесі:

y = 121.81



Тармақты алгоритм схемасы:

if-elif нұсқаулары (шартты өрнектер)

```
A = 90; B = 70
C = 60; D = 50
# Қолданушы балл енгізеді
ball = int(input('Балл мөлшерін енгізіңіз: '))
# Баллдың әріптік баламасын анықтау
if ball >= A:
    print('Әріптік баға - A')
elif ball >= B:
    print('Әріптік баға - B')
elif ball >= C:
    print('Әріптік баға - C')
elif ball >= D:
    print('Әріптік баға - D')
else:
    print('Әріптік баға - F')
```

Балл мөлшері	Әріптік бағасы
90 – 100	A
70 – 89	B
60 – 69	C
50 – 59	D
0 – 49	F

Практикалық жұмыс

- 1. екі айнымалыға сандық мәндер меншіктеледі;
- 2. егер бірінші айнымалының мәні екіншісінен артық болса, онда айнымалылар мәндерінің айырмасын (біріншісінен екіншісін азайту) табыңыздар, нәтижесін үшінші айнымалыға меншіктеңіздер;
- 3. егер бірінші айнымалының мәні екіншісінен кіші болса, онда үшінші айнымалы екеуінің қосындысына тең болады;
- 4. қалған барлық жағдайларда, үшінші айнымалыға бірінші айнымалы мәнін меншіктеңіз;
- 5. экранға айнымалылар мәндерін шығарыңыздар.

Практикалық жұмыс

- A және B бүтін сандары берілген. Егер $A < B$ болса, A -дан B -ға дейінгі барлық сандарды өсу ретімен, кері жағдайда, оларды кему ретімен экранға шыарыңыз.
- A және B бүтін сандары берілген, $A > B$. A -дан B -ға дейінгі барлық тақ сандарды кему ретімен экранға шығарыңыз. Бұл есепті `if` операторын қолданбай шығаруға болады.
- 10 бүтін сан берілген. Солардың қосындысын табу керек. Осы программаны айнымалылардың ең аз санын қолдана отырып шығаруға тырысыңыз.
- Берілген n натурал санының факториалын ($n!$) табу қажет. Мұнда математикалық `math` кітапханасын қолдануға болмайды.

Практикалық жұмыс

Берілген **n** натурал саны бойынша келесі қосындыны

$$s = 1!+2!+3!+\dots+n!1!+2!+3!+\dots+n!$$

табу керек. Бұл есепте тек бір ғана циклді қолдануға болады.
Мұнда **math** кітапханасын қолдануға да болмайды.

Математикалық операторлар

Операциясы	Синтаксисі	Функциясы
Қосу	$a + b$	<code>add(a, b)</code>
Біріктіру	$seq1 + seq2$	<code>concat(seq1, seq2)</code>
Containment Test	$obj \text{ in } seq$	<code>contains(seq, obj)</code>
Бөлу	a / b	<code>truediv(a, b)</code>
Бөлу (бүтін бөлінді алу)	$a // b$	<code>floordiv(a, b)</code>
ЖӘНЕ (биттер үшін)	$a \& b$	<code>and_(a, b)</code>
Bitwise Exclusive Or	$a \wedge b$	<code>xor(a, b)</code>
ТЕРІСТЕУ (биттер үшін)	$\sim a$	<code>invert(a)</code>
НЕМЕСЕ (биттер үшін)	$a b$	<code>or_(a, b)</code>
Дәрежелеу	$a ** b$	<code>pow(a, b)</code>
Identity	$a \text{ is } b$	<code>is_(a, b)</code>
Identity	$a \text{ is not } b$	<code>is_not(a, b)</code>
Тізім элементін өзгерту	$obj[k] = v$	<code>setitem(obj, k, v)</code>
Тізім элементін өшіру	$\text{del } obj[k]$	<code>delitem(obj, k)</code>
Тізім элементін индекстеу	$obj[k]$	<code>getitem(obj, k)</code>
Солға жылжыту (биттер)	$a \ll b$	<code>lshift(a, b)</code>
Қалдық табу	$a \% b$	<code>mod(a, b)</code>

Математикалық операторлар (жалғасы)

Операциясы	Синтаксисі	Функциясы
Көбейту	$a * b$	<code>mul(a, b)</code>
Матрицалық көбейту	$a @ b$	<code>matmul(a, b)</code>
Терістеу (арифметикалық)	$-a$	<code>neg(a)</code>
Терістеу (Логикалық)	<code>not a</code>	<code>not_(a)</code>
Оң таңбалы	$+a$	<code>pos(a)</code>
Оңға жылжыту	$a \gg b$	<code>rshift(a, b)</code>
Тізім бөлігін орнату	<code>seq[i:j] = values</code>	<code>setitem(seq, slice(i, j), values)</code>
Тізім бөлігін өшіру	<code>del seq[i:j]</code>	<code>delitem(seq, slice(i, j))</code>
Тізім бөлігін қиып алу	<code>seq[i:j]</code>	<code>getitem(seq, slice(i, j))</code>
Тіркесті форматтау	$s \% obj$	<code>mod(s, obj)</code>
Азайту	$a - b$	<code>sub(a, b)</code>
Дұрыстығын тесеру	<code>obj</code>	<code>truth(obj)</code>
Реттеу	$a < b$	<code>lt(a, b)</code>
Реттеу	$a \leq b$	<code>le(a, b)</code>
Тең екенін анықтау	$a == b$	<code>eq(a, b)</code>
Тең емес екенін анықтау	$a \neq b$	<code>ne(a, b)</code>
Реттеу	$a \geq b$	<code>ge(a, b)</code>

Мәліметтерді енгізу

- Мәліметтерді енгізу үшін ішкі `input` деп аталатын функцияны қолдануға болады:
- Мысалы:
 - `a = input("Қолданушыдан бір нәрсе енгізуді сұраңыз ")` (python 3.*)
 - Тек сөз тіркесін (string) қайтарады
 - `b = raw_input("Қолданушыдан бір нәрсе енгізуді сұраңыз ")` (python 2.*)
 - Бұл да тек сөз тіркесін (string) қайтарады
 - `b = input("Қолданушыдан бір нәрсе енгізуді сұраңыз ")` (python 2.*)
 - Енгізілген мәліметтер типін қайтарады

Мәліметтерді енгізу

Атың кім? Бекен
Сәлем, Бекен
Python үйренгің келе ме? (Иә/Жоқ) Иә
онда дос боламыз, программа құрамыз...

- Пернелерден мәліметтер енгізу - `input("Бірдеңе енгіз:")`
- Программа жұмысы тоқталады, сол сәтте бір мән енгізіп, **Enter** бассaq, жұмыс ары қарай жалғасады.

- Мысал:

```
name = input("Атың кім?")
```

```
print('Сәлем, ' + name)
```

```
jaуap = input(" Python үйренгің келе ме? (Иә/Жоқ) ")
```

```
if jaуap == "Иә" :
```

```
    print(" онда дос боламыз, программа құрамыз...")
```

```
elif jaуap == "Жоқ" :
```

```
    print(" онда 'Қош бол!' үйіңе қайт...")
```

```
else :
```

```
    print("Не деп отырсың? Тұр орныңнан...")
```

Атың кім? Секен
Сәлем, Секен
Python үйренгің келе ме? (Иә/Жоқ) Жоқ
онда 'Қош бол!' үйіңе қайт...

Атың кім? Төкен
Сәлем, Төкен
Python үйренгің келе ме? (Иә/Жоқ)
Білмедім
Не деп отырсың? Тұр орныңнан...

Кодқа комментарийлер қосу

- Кодқа комментарий қосу программалаудың жақсы тәсілі
- Комментарийлер сіздің кодыңызды басқалардың (бірігіп жұмыс істейтін мамандардың) түсінуін жеңілдетеді
- Кодқа комментарийлер қосу үшін келесі тәсілдерді қолданыңыз:
 - # - осы таңбадан (шарп) басталатын комментарий жол соңында тұрады
 - ''' сөйлемдер ''' – үш жалқы тырнақшамен басталып, үш жалқы тырнақшамен аяқталатын күрделі комментарийлер, бұлар бірнеше жолдардан тұратын түсініктеме мәтіндер түрінде болады

Сөз тіркестері

- Python-да сөз тіркестерін (тіркестерді) де салыстыруға болады.
Мысалы:

```
name1 = 'Mary'
```

```
name2 = 'Mark'
```

```
if name1 == name2:
```

```
    print ('Аттар бірдей.')
```

```
else:
```

```
    print ('Аттар бірдей ЕМЕС.')
```

'Mary' мен 'Mark' бірдей емес, сондықтан **else** бөлігі 'Аттар бірдей ЕМЕС.' деген нәтиже береді.

Сөз тіркестері

- Келесі программа құпиясөз (пароль - password) енгізуді сұрайды, сонан соң оның бұрыннан сақталған **'koseu'** сөзімен бірдей екендігі анықталады.

```
# Қолданушыдан құпиясөз енгізуді сұрау
password = input('Құпиясөз енгізіңіз: ')
if password == 'koseu':
    print ('Бәрі дұрыс. ')
else:
    print ('Құпиясөз дұрыс емес.')
```

Нәтижесі:

Құпиясөз енгізіңіз: koseu
Бәрі дұрыс.

Сөз тіркестері

```
A = 90; B = 70; C = 60; D = 50
# Қолданушы балл енгізеді
ball = int(input('Балл мөлшерін енгізіңіз: '))
if ball >= A:
    print('Әріптік баға - A')
elif ball >= B:
    print('Әріптік баға - B')
elif ball >= C:
    print('Әріптік баға - C')
elif ball >= D:
    print('Әріптік баға - D')
else:
    print('Әріптік баға - F')
```

Нәтижесі:

Балл мөлшерін енгізіңіз: 85

Әріптік баға - B

Сөз тіркестері

- ASCII кодтары символдарды реттеп орналастырады. "А" символы "В" символының алдында тұрады (А – 65, В – 66), ал ол "С" символының алдында (С - 67), т.с.с.
- Программа символдарды салыстырғанда, олардың кодтарын салыстырады, мысалы, мынадай if:

```
if 'a' < 'b': print ('a әрпі b әрпінен кіші ')
```

дұрыс жазылған шарт, яғни 'a' – 97, ал 'b' – 98 болып табылады.

A-Z кодтары 65 пен 90 аралығында,

ал a-z 97 мен 122 аралығында орналасқан.

Мысалдар орындау

```
a = 54321.6789
print('a санының мәні = ',format(a,'10.2f')) # 10.2
# a санының мәні = 54321.68 ұзындығы – 10, бөлшегі - 2
print(format(123456789,'14,d')) # ұзындығы – 14 цифр
# 123,456,789
```

Нәтижесі: a санының мәні = 54321.68
123,456,789

Мысалдар орындау

```
# Төмендегі нақты (бөлшек) сандар  
# нүкте бойынша тураланып, бір бағанаға шығарылады.  
num1 = 127.899  
num2 = 3465.148  
num3 = 3.776  
# Сандар ені - 7 таңба, дәлдігі, яғни бөлшегі 2 - таңба  
print(format(num1, '7.2f'))  
print(format(num2, '7.2f'))  
print(format(num3, '7.2f'))
```

Нәтижелері:

127.90

3465.15

3.78

Мысалдар орындау

- Бүтін сандарды форматтау
- `format` функциясы бүтін сандарды да форматтайды. Оның форматын жазудың екі ерекшелігі бар:
 - тип көрсеткіші ретінде **d** символы жазылады;
 - дәлдігі көрсетілмейді.

123456 санын форматсыз жазайық:

```
print(format(123456, 'd'))
```

Нәтижесі: 123456

Ал енді сол санның мыңдықтарын үтірмен бөліп жазайық:

```
print(format(123456, ',d'))
```

Нәтижесі: 123,456

Мысалдар орындау

- Нақты сандарды пайыз (%) түрінде форматтап шығару
- Мұнда **f** орнына тип көрсеткіші ретінде **%** символы қолданылады. **%** символы санды 100-ге көбейтіп, оның соңына **%** таңбасын қосып жазады. Төменде мысал келтірілген:

```
print(format(0.5, '%'))
```

Нәтижесі: 50.000000%

- Дәлдігін нөл арқылы берсек, нәтижесі бүтін сан болады:

```
print(format(0.5, '.0%'))
```

Нәтижесі: 50%

