

КОММЕРЦИЯЛЫҚ ЕМЕС АКЦИОНЕРЛІК ҚОҒАМЫ
«ӘБІЛҚАС САҒЫНОВ АТЫНДАҒЫ ҚАРАҒАНДЫ ТЕХНИКАЛЫҚ
УНИВЕРСИТЕТІ»

Киберқауіпсіздік және жасанды интеллект кафедрасы

Аубакирова А.Е.

«PYTHON ОБЪЕКТІГЕ БАҒЫТТАЛҒАН БАҒДАРЛАМАЛАУ»
ПӘНІ БОЙЫНША
ДӘРІСТЕРГЕ
ӘДІСТЕМЕЛІК НҰСҚАУЛАР

Қарағанды 2026

КОММЕРЦИЯЛЫҚ ЕМЕС АКЦИОНЕРЛІК ҚОҒАМЫ
«ӘБІЛҚАС САҒЫНОВ АТЫНДАҒЫ ҚАРАҒАНДЫ ТЕХНИКАЛЫҚ
УНИВЕРСИТЕТІ»

Киберқауіпсіздік және жасанды интеллект кафедрасы

Аубакирова А.Е.

«PYTHON ОБЪЕКТІГЕ БАҒЫТТАЛҒАН БАҒДАРЛАМАЛАУ»
ПӘНІ БОЙЫНША
ДӘРІСТЕРГЕ
ӘДІСТЕМЕЛІК НҰСҚАУЛАР

6B06105 «Data Science» білім беру бағдарламасы

Қарағанды 2026

ДӘРИС КОНСПЕКТІЛЕРІ

Тақырып 5. Python тіліндегі тармақталу құрылымы.

Жұмыстың мақсаты: if тармақталу құрылымын зерттеу. Тармақталу құрылымын қолдана отырып, сандармен және жолдармен жұмыс істеуді үйреніңіз.

Теориядан қысқаша мәліметтер

Python-дағы логикалық мәндер екі логикалық тұрақты шын (шын) және жалған (жалған) болып табылады. Логикалық мәндер логикалық операциялар мен логикалық жағдайларды есептеу нәтижесінде алынады. Негізгі логикалық амалдар мен өрнектер 1-кестеде келтірілген.

Кесте 1.

Шарт	Сипаттамасы
>	$a > b$ екенін тексеріңіз
<	$a < b$ екенін тексеріңіз
==	Біз a-ның b-қа тең екенін тексереміз
!=	a-ның b-ға тең емес екенін тексеріңіз
not x	Бас тарту (x шарты орындалмайды)
x and y	Логикалық "ЖӘНЕ". X and y шарттарын орындау үшін x және y шарттары бір уақытта орындалуы керек.
x or y	Логикалық "НЕМЕСЕ". X or y шартын орындау үшін шарттардың кем дегенде біреуі орындалуы керек.

If тармақтау операторы кейбір логикалық жағдайларға байланысты белгілі бір нұсқаулар жиынтығын орындауға мүмкіндік береді. Келесі пайдалану жағдайлары мүмкін.

1. If құрылымы

If операторының синтаксисі келесідей:

if логикалық шарт:

команда_1

команда_2

...

команда_n

Айта кету керек, if блогындағы әрбір Нұсқаулық if операторынан бірдей шегініске ие. C, C++, Java және PHP сияқты көптеген тілдер блогтың басы мен

соңын анықтау үшін жақшаларды қолданады, бірақ Python бұл үшін шегіністерді қолданады.

Мысал:

```
if number > 10:  
    print ("10-нан үлкен сан")
```

2. If-else құрылымы

Бағдарламаны орындаудың балама нұсқасын қарастыру қажет болған жағдайлар болады. Шартты орындау кезінде сіз бір нұсқаулық жиынтығын, әйтпесе басқа жиынтығын орындауыңыз керек. Ол үшін if-else конструкциясы қолданылады.

If-else операторының синтаксисі келесідей:

if логикалық шарт:

```
    команда_1  
    команда_2  
    ...  
    команда_n  
else:  
    команда_1  
    команда_2  
    ...  
    команда_n
```

Мысал 1:

```
radius = int(input ("радиусты енгізіңіз:"))  
if radius >= 0:  
    print ("шеңбер Ұзындығы=", 2 * 3.14 * radius)  
    print ("аудан=", 3.14 * radius * * 2)  
else:  
    print ("оң санды енгізіңіз")
```

Мысал 2:

```
құпия сөз = енгізу ("парольді енгізіңіз:")  
if password == "12s54h":  
    print ("қош келдіңіз")  
else:  
    print ("кіруге тыйым салынады")
```

3. if – elif – else құрылымы

"Шығу" мүмкін " егер-Элиф-басқа. Синтаксистік оператор-if-Elif-else
выгля:

егер логикалық шарт 1 болса:

```
    команда_1
```

```
команда_2
...
команда_n
```

elif логикалық терминдер_2:

```
команда_1
команда_2
...
команда_n
```

elif логикалық терминдер_3:

```
команда_1
команда_2
```

...

```
commanda_n
```

тағы:

```
команда_1
команда_2
...
команда_n
```

If-elif-else нұсқаулығы орындалған кезде, ең алдымен, `_1` шарты тексеріледі. Егер шарт дұрыс болса, онда `if` Нұсқаулық блогы орындалады. Келесі шарттар мен нұсқаулар жіберілмейді және басқару `if-elif-else` блогынан кейін операторға өтеді.

Егер `_1` шарты жалған болса, онда Басқару келесі `elif` шартына өтеді және `_2` шарты тексеріледі. Егер бұл рас болса, онда `elif` блогындағы нұсқаулар орындалады. Келесі шарттар мен нұсқаулар жіберілмейді және басқару `if-elif-else` блогынан кейін операторға өтеді.

Бұл процесс `elif` шарты болғанша қайталанады, ол дұрыс болып шығады. Егер бұл болмаса, онда `else` блогы ең соңында орындалады.

Мысал:

```
score = int (input ("ұпайларыңызды енгізіңіз:"))
```

```
if score >= 90:
```

```
    print("керемет! Сіздің бағалауыңыз а")
```

```
elif score >= 75:
```

```
    print("жарайды! Сіздің бағаңыз-В")
```

```
elif score >= 65:
```

```
    print("қанағаттанарлық! Сіздің бағаңыз-С")
```

```
elif score >= 50:
```

```
    print("сіздің бағалауыңыз - D. материалды қайталауға тұрарлық.")
```

```
else:
```

```
    print("Сіз емтихан тапсырған жоқсыз")
```

Жұмысты орындау тәртібі

1. Теориялық материалды зерттеу.
2. Тапсырмаларды орындау.
3. Бақылау сұрақтарына жауап беріңіз.

1 тапсырма

1. Бағдарлама пайдаланушыдан екі санды сұрайды, содан кейін оларды салыстырады және егер сандар тең болса, онда тиісті хабарлама көрсетіледі.
2. Бағдарлама пайдаланушыдан екі санды сұрайды, содан кейін оларды салыстырады және үлкен санды көрсетеді.
3. Бағдарлама пайдаланушыдан пароль сұрайды, содан кейін тиісті хабарлама береді.

2 тапсырма

1. Үш сан берілген. Олардың ең кішісін тауып, экранға шығарыңыз.
2. $Ax^2 + Bx + c = 0$ түріндегі квадрат теңдеуді шешу бағдарламасын жасаңыз. А, В, С коэффициенттері пернетақтадан орнатылады.
3. Төрт сан берілген. Олардың ең үлкенін тауып, экранға шығарыңыз.

3 тапсырма

1. Жеңілдікті ескере отырып, сатып алу құнын есептеу бағдарламасын жазыңыз. 5% жеңілдік Егер сатып алу сомасы 2000 теңгеден артық болса, 8% -ға-егер сома 5000 теңгеден артық болса беріледі.
2. Осы санға (1-12) сәйкес келетін айдың атауын ағылшын тілінде көрсететін бағдарлама жазыңыз.
3. X, y, z жақтары бар тікбұрышты үшбұрыштың бар-жоғын анықтаңыз.

Есеп мазмұны

1. Тақырып парағы.
2. Орындалған жұмыстардың нәтижелері.
3. Бақылау сұрақтарына жауаптар.

Бақылау сұрақтары

1. Бағдарламаның мысалын if – else операторымен жазыңыз.
2. If – elif – else операторы қашан қолданылады?
3. If операторында қандай шарттарды қолдануға болады?
4. X және y шарттарын қолдануға мысал келтіріңіз.

5. Тегін тексеру бағдарламасын жазыңыз.

Python тілінің синтаксисі

Python тілінің синтаксисі, тілдің өзі сияқты, өте қарапайым. Онда күрделі құрылымдар жоқ, сондықтан оны үйрену өте оңай.

Python тілінің синтаксисінің негізгі принциптері:

1. Жолдың соңы - Нұсқаулықтың соңы болып табылады (үтір қажет емес).

Мысалы:

```
a = 5
```

```
b = 3
```

```
print(a + b)
```

2. Еңгізілген нұсқаулар шегініс өлшемі бойынша блоктарға біріктіріледі. Шегініс кез-келген болуы мүмкін, бастысы, бір кірістірілген блоктың ішінде шегініс бірдей болады.

Мысалы:

```
if a == 5:
```

```
    print('yes')
```

```
    a += 1
```

```
y=a*12
```

Негізгі нұсқаулық қос нүктемен аяқталған кезде, одан кейін негізгі нұсқаулық жолының астында шегінісі бар кірістірілген код блогы орналасқан.

Командаларды бір жолға жазып, оларды ";" арқылы бөлу мүмкіндігі бар.

Алайда, бұл әдісті жиі қолдануға болмайды, бұл көрінуді азайтады.

```
a = 1; b = 2; print(a)
```

```
print (a + b)
```

Python тіліндегі деректер түрлері

Python тілінде көптеген мәліметтер бар: бүтін сандар (int), өзгермелі нүктелер (float), жолдар (str), логикалық тип (Boolean), тізім (List) және т. б.

Әр айнымалының түрі бағдарламаның орындалу барысында динамикалық түрде өзгеруі мүмкін. Type () командасын қолдана отырып, айнымалының типін анықтауға болады.

Мысалы:

```
a = 72
```

```
print(a, "is of type", type(a))
```

Шығады

```
72 is of type <class 'int'>
```

```
a = 2.0
```

```
print(a, "is of type", type(a) )
```

```
2.0 is of type <class 'float'>
```

Бүтін санды сақтау үшін компьютердің жадында 4 байт қолданылады. Нақты Сан-бұл өзгермелі нүкте саны, оны сақтау үшін компьютердің жадында 8 байт қолданылады. Мысалы, 2.0 немесе 58.241. Жолдар Юникодты кодтаудағы таңбалар жиынтығын білдіреді.

Сандарды нақты заттардан бүтін санға және керісінше Python-да түрлендіру үшін `int()` және `float ()` функциялары анықталған. Мысалы, `int(12.6)` нәтижесінде 12, ал `float (12)` нәтижесінде 12.0 (ондық бөлгіш - нүкте) береді.

Бүтін сандардың негізгі операциялары

$A + B$ -сомма;

$A - B$ -айырмашылық;

$A * B$ -шығарма;

A / B — бөлу, (бұл әрекеттің нәтижесі нақты сан болып табылады, тіпті егер A B -ге бөлінсе де);

$A \% B$ - A -ның B -ға бөлінуінен қалған қалдық;

$A // B$ - A -ның B -ға бөлінуінің тұтас бөлігі

$A ** B$ — дәрежеге аудару.

Сонымен қатар, Python-да сандық операциялар үшін функциялар қолданылады:

`abs()` - абсолюттік мәнді есептеу;

`pow ()` - дәрежеге ауыстыру, `pow (2,3) = 8`;

`div mod ()` - бүтін санды бөлу мен қалдықтың нәтижесін есептеу, `div mod(17,5)` аламыз (3,2);

`round ()` - дөңгелектеу (шамамен), `round(5.7) = 6`.

Бұл мүмкіндіктер "кіріктірілген", яғни оларды пайдалану үшін қосымша кітапханаларды қосудың қажеті жоқ.

Мысалы:

```
x = 17
```

```
y=divmod(x,4)
```

```
print("бүтін бөлік",y[0])
```

```
print("қалдық", y[1])
```

Шығады

бүтін бөлік 4

қалдық 1

Квадраттық түбір, синус, тангенс және т.б. есептеу сияқты сандармен жұмыс істеуге арналған барлық басқа математикалық функциялар **math** кітапханасын қосуды қажет етеді. Бұл модульмен жұмыс істеу үшін оны алдымен **import math** командасымен импорттау керек.

Мысалы:

```
import math
```

```
x=16
```

```
y=math.log2(x)
```

```
print(y)
```

Әдепкі бойынша, стандартты сандар ондық жүйеде сандар ретінде қарастырылады. Бірақ Python екілік, сегіздік және он алтылық жүйелердегі сандарды да қолдайды.

Сан екілік жүйені білдіретінін көрсету үшін санның алдына 0b префиксі қойылады:

```
a = 0b11
```

```
b = 0b1011
```

```
c = 0b100001
```

```
print(a) # 3 ондық жүйеде
```

```
print(b) # 11 ондық жүйеде
```

```
print(c) # 33 ондық жүйеде
```

Python-бұл объектіге бағытталған, түсіндірілетін, жоғары деңгейлі портативті тіл. Жинақта Python аудармашысымен бірге IDLE (интеграцияланған даму ортасы) бар.

IDLE көмегімен сіз интеграцияланған орта үшін әдеттегі тапсырмаларды орындай аласыз: Python бағдарламаларын қарау, өңдеу, іске қосу, күйін келтіру. Код редакторы синтаксисті бөлектеуді қолданады. IDLE Python іске қосылғаннан кейін бағдарлама терезесі пайда болады. Бұл терезеде қарапайым командаларды орындауға және тіпті оны калькулятор ретінде пайдалануға болады. Файл мәзірі арқылы сіз Python-да жаңа бағдарлама жасай аласыз немесе бар бағдарламаны аша аласыз.

Қазіргі уақытта тілдің екі нұсқасы белсенді қолданылады-ескі 2x нұсқасы және қазіргі заманғы 3x нұсқасы. 2x нұсқасы енді дамымайды, бірақ әлі де қолданылады, өйткені көптеген бағдарламалар мен кітапханалар дәл 2x нұсқасы үшін жасалған. 2-ші нұсқадағы Python бағдарламасы 3-ші нұсқада жұмыс істемеуі мүмкін және керісінше, бірақ олар өте ұқсас.

Деректердің негізгі типтері

1. Сандар

- int - бүтін сан
- float-өзгермелі нүкте саны

2. Жолдар

- str

3. Логикалық түрі

- boolean

4. Тізім

- List

Деректерді енгізу және шығару

Деректерді енгізу input пәрмені (енгізу тізімі) арқылы жүзеге асырылады:

```
a = input()
```

Input () пәрмені әдепкі бойынша енгізулерді таңбалар жолы ретінде қабылдайды, сондықтан бүтін санды енгізу үшін int () деректер түрін көрсету керек:

```
a = int (input())
```

Нақты сандарды енгізу үшін пәрмен қолданылады

```
a=float(input())
```

Деректерді шығару print(шығару тізімі) пәрмені арқылы жүзеге асырылады:

```
a = 1
```

```
b = 2
```

```
print(a)
```

```
print(a + b)
```

```
print('сума = ', a + b )
```

Әдетте Python бағдарламасы келесі бөліктерден тұруы керек:

- деректерді оқу,

- мәселені шешу,

- нәтижені шығару.

Мысалы, екі санның қосындысын есептейтін бағдарлама келесі формада болуы мүмкін:

```
a = int(input())
```

```
b = int(input())
```

```
sum = a + b
```

```
print(sum)
```

Сандар бойынша математикалық операциялар

$x + y$	Қосу
$x - y$	Азайту
$x * y$	Көбейту
x / y	Бөлу
$x // y$	Бөлінуден бүкіл бөлікті алу
$x \% y$	Бөлінуден қалған қалдық
$-x$	Сан белгісін ауыстыру
$abs(x)$	Сандар модулі
$x ** y$	Дәрежеге көтеру

Python стандартты жеткізіліміне math кітапханасы кіреді, оның құрамында жиі қолданылатын математикалық функциялар көп. Бұл модульмен жұмыс істеу үшін оны алдымен **import math** командасымен импорттау керек.

Math Модулінің ең көп қолданылатын функциялары

math.ceil(x)	X-тен үлкен бүтін санды қайтарады
math.fabs(x)	X санының абсолютті мәнін қайтарады
math.factorial(x)	Факториалды x есептейді
math.floor(x)	X-тен кіші бүтін санды қайтарады
math.exp(x)	E^{**x} есептейді
math.log2(x)	2 негіз бойынша Логарифм
math.log10(x)	10 негіз бойынша Логарифм
math.log(x[, base])	Әдепкі бойынша, ол логарифмді e негізінде есептейді, қосымша Логарифм негізін көрсетуге болады
math.pow(x, y)	X мәнін у дәрежесінде есептейді
math.sqrt(x)	X-тен квадрат түбір

Python түрлендіру

Егер жол сандардан тұрса, онда сіз бұл жолды сан түрінде ұсына аласыз, сонда онымен арифметикалық амалдарды орындауға болады. Ол үшін int функциясы қолданылады. Кері түрлендіру үшін str функциясы қолданылады.

Мысалы, int ('123') 123 бүтін санды қайтарады, str (123) '123'жолын қайтарады. Int функциясы бөлшек бөлігін алып тастау арқылы нақты санды бүтін санға айналдыруға көмектеседі: int(12.3) = 12, int(-12.3) = -12.

Жұмысты орындау тәртібі

1. Теориялық материалды зерттеу.
2. Тапсырмаларды орындау.
3. Бақылау сұрақтарына жауап беріңіз.

1 тапсырма

Пайдаланушыдан x және y екі бүтін санды сұрайтын бағдарлама жасаңыз; Осыдан кейін ол экранға шығады:

- x және y сандарының қосындысы;
- x және y сандарының көбейтіндісі;
- X / y сандарының бөлінуі;
- x // y бөлінуінен толық бөлігін алу;
- X % y бөлуден қалған қалдық;
- x санының квадраты.

2 тапсырма

Пайдаланушыдан x бүтін санын сұрайтын бағдарлама жасаңыз.

Осыдан кейін ол экранға шығады:

- x санының 2 негізіндегі логарифм;
- x санының 10 негізіндегі логарифм;
- 3 дәрежелі x мәні;
- x -тен квадрат түбір;
- 4 дәрежесіндегі x мәні.

3 тапсырма

Пайдаланушыдан нақты x санын сұрайтын бағдарлама жасаңыз.

Осыдан кейін ол экранға шығады:

- ең жақын бүтін сан x -тен үлкен;
- ең жақын бүтін сан x -тен аз;
- x санының екі еселенген мәні;
- x санының жартысы.

4 тапсырма

1. $Y = (5+x)*(12-x)$ түріндегі теңдеуді шешу бағдарламасын жасаңыз. Айнымалы x пернетақтадан орнатылады.
2. Үшбұрыштың ауданын үш жағынан есептеу бағдарламасын жасаңыз. Тараптардың мәндері пернетақтадан орнатылады.
3. Пифагор теоремасына сәйкес дұрыс үшбұрыштың гипотенузасын есептеу бағдарламасын жасаңыз. Аяқ мәндері пернетақтадан орнатылады.
4. Формула бойынша сақинаның ауданын есептеу бағдарламасын жасаңыз $S=3,14 (R^2-r^2)$. Шеңбер Радиусы пернетақтадан орнатылады.

5 тапсырма

1. Пайдаланушыдан төрт санды сұрайтын бағдарлама жасаңыз. Содан кейін ол бірінші екеуін және екінші екеуін бөлек қосады. Содан кейін ол бірінші соманы екінші сомаға бөледі. Жауапта үтірден кейін екі сан болуы үшін нәтижені көрсетеді.
2. Үш таңбалы санның бүтін сандарын қосу бағдарламасын жасаңыз. Нөмір пернетақтадан енгізіледі.

6 тапсырма

1. Екілік санды ондық жүйеге аудару бағдарламасын жасаңыз.
2. Ондық санды екілік жүйеге аудару бағдарламасын жасаңыз.
3. Ондық санды он алтылық жүйеге аудару бағдарламасын жасаңыз.

Есеп мазмұны

1. Титулдық парақ.
2. Орындалған жұмыстардың нәтижелері.
3. Бақылау сұрақтарына жауаптар.

Бақылау сұрақтары

1. IDLE дегеніміз не?
2. Python-да бүтін санды нақты санға қалай түрлендіруге болады.
3. Бүтін сандарды енгізуді қай команда жүзеге асырады?
4. Экранға деректерді шығару командасы мысал келтіріңіз.
5. Қандай кітапханада математикалық функциялар бар?