



Муаллифони маводи таълимӣ

Ҳакимов Раҳматҷон	Холматов Эркин
Азимов Парвиз	Кабулов Муродалӣ
Ќӯраев Ҳамзилло	Абдуллоев Донёр
Ўлмасов Муҳаммадюсуф	

Нашри навтарини маводи таълимӣ дар ин ҷо:

<https://docs.google.com/presentation/d/1ltaZqXnTp2EHMPIJFkARXBtN3bMcfJp4wX6i0XGj6Xg>

Коди намунаҳо дар ин ҷо:

<https://github.com/Oftobcom/python-3-examples>



Мавзӯи 1

1. Алгоритм
2. Псевдокод
3. Забонҳои барномасозӣ
4. Забони барномасозии Python 3
5. Насби муҳити барномасозии Python 3

Алгоритм



Мафҳуми алгоритм – яке аз мафҳумҳои асосии барномасозӣ аст. Алгоритм – ин пайдарпаии фармонҳо ба иҷрокунанда, ки дар натиҷаи иҷрои ин пайдарпай иҷрокунанда масъалаи гузошташударо ҳал мекунад. Алгоритм дар забони расмие, ки ба гуногунфаҳмӣ роҳ наметаҳад, навишта мешавад.

Иҷрокунанда – одам, компютер, дастгоҳи автоматӣ ва ғайра буда метавонад. Иҷрокунанда бояд механикӣ ва бе баҳсу мунозира ҳамаи фармонҳои дар таркиби алгоритм бударо иҷро карда тавонад.

Алгоритм



Сабти алгоритм дар забони расмӣ барнома (program) номида мешавад. Дар баъзе мавридҳо худӣ мафҳуми алгоритм бо сабти он иваз карда мешавад. Яъне калимаҳои «алгоритм» ва «барнома» қариб синоним ҳастанд.

Бо ин калимаи «алгоритм» одатан ғояи асосии бунёди алгоритмро, ки барои ҳамаи забонҳои алгоритмӣ умумӣ аст, дар назар доранд. Калимаи «барнома» бошад фақат ба сабти алгоритм дар ягон забони расмӣ мушаххас алоқаманд аст.

Мафҳуми псевдокод



Алгоритми ба намуди псевдокод навишташуда бисёр возеҳ ва фаҳмотар аст. Вай имконият медиҳад, ки ба таври озод ҳалли масъала тасвир карда шуда, баъд муфассал баён шавад.

Барои мутахассисон навиштани барнома дар дилхоҳ забони барномасозӣ аз рӯи псевдокод ягон мушкилӣ надорад.

Дар зери мафҳуми «код» матни барномаи компютерӣ фаҳмида мешавад.

Намунаи псевдокод

Дохил кардани кимати a .

Агар $a > 0$ бошад, он гоҳ $a = a + 4$.

Вагарна $a = a - 4$.

Чоп кардани кимати a .

Итмом.

$$F(a) = a + 4 \text{ агар } a > 0$$

$$F(a) = a - 4 \text{ агар } a \leq 0$$

Намунаи код

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

```
if a > 0:
```

```
    a = a + 4
```

```
else:
```

```
    a = a - 4
```

```
print (a)
```

$F(a) = a + 4$ агар $a > 0$

$F(a) = a - 4$ агар $a \leq 0$

Забонҳои барномасозӣ




Имрӯз дар ҷаҳон бештар аз 2000 забони барномарезӣ мавриди истифода қарор дорад.

Ҳар гуна забони барномарезӣ ба мисли забонҳои муқаррарии гуфтугӯ аз алифбо, синтаксис ва семантика иборат аст.

Синтаксис ин маҷмуи қоидаҳост, ки бо ёрии алифбои забон сохтори барнома (блокҳо, ифодаҳо, операторҳо, зербарномаҳо ва ғайра) эҷод мешавад.

Мисолҳои забонҳои барномасозӣ: Python, goLang, Pascal, PHP, ..., ғуруҳи забонҳои C (C, C++, C#)

Забони барномасозии Python 3

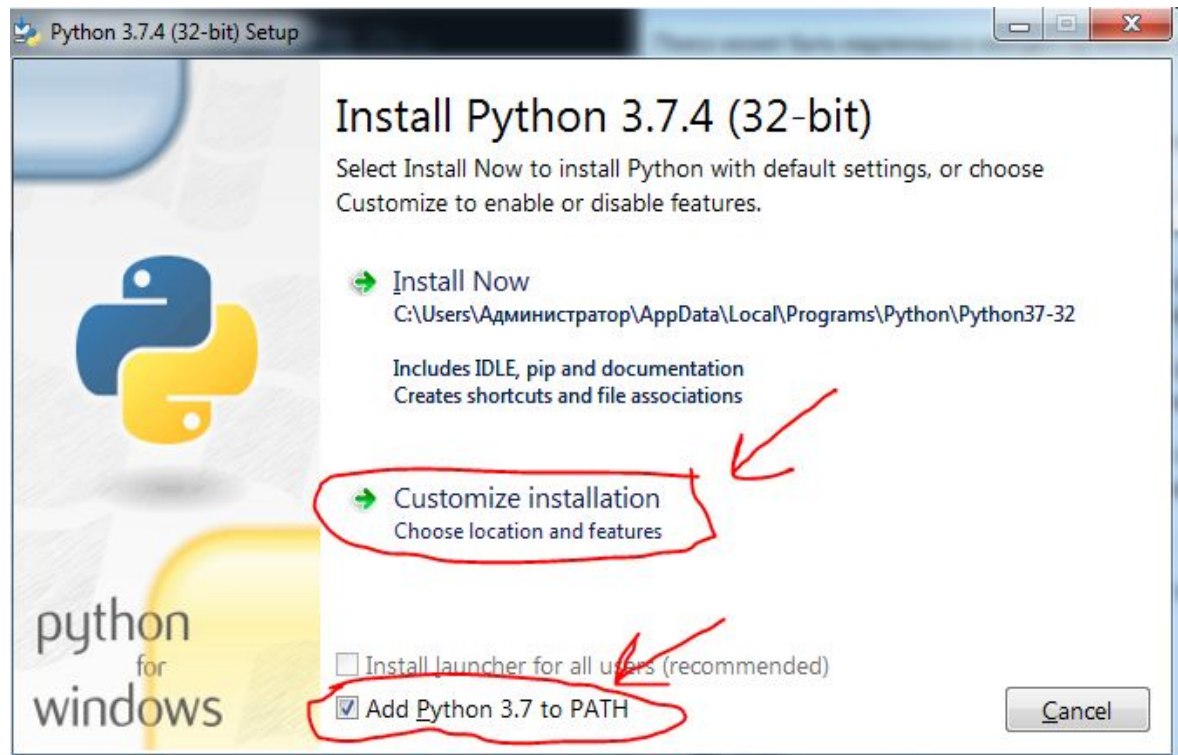


Гидо ван Россум ба кор дар Python дар охири солҳои 80 ҳамчун вориси забони барномасозии ABC оғоз кард ва бори аввал онро соли 1991 ҳамчун Python 0.9.1 нашр кард.

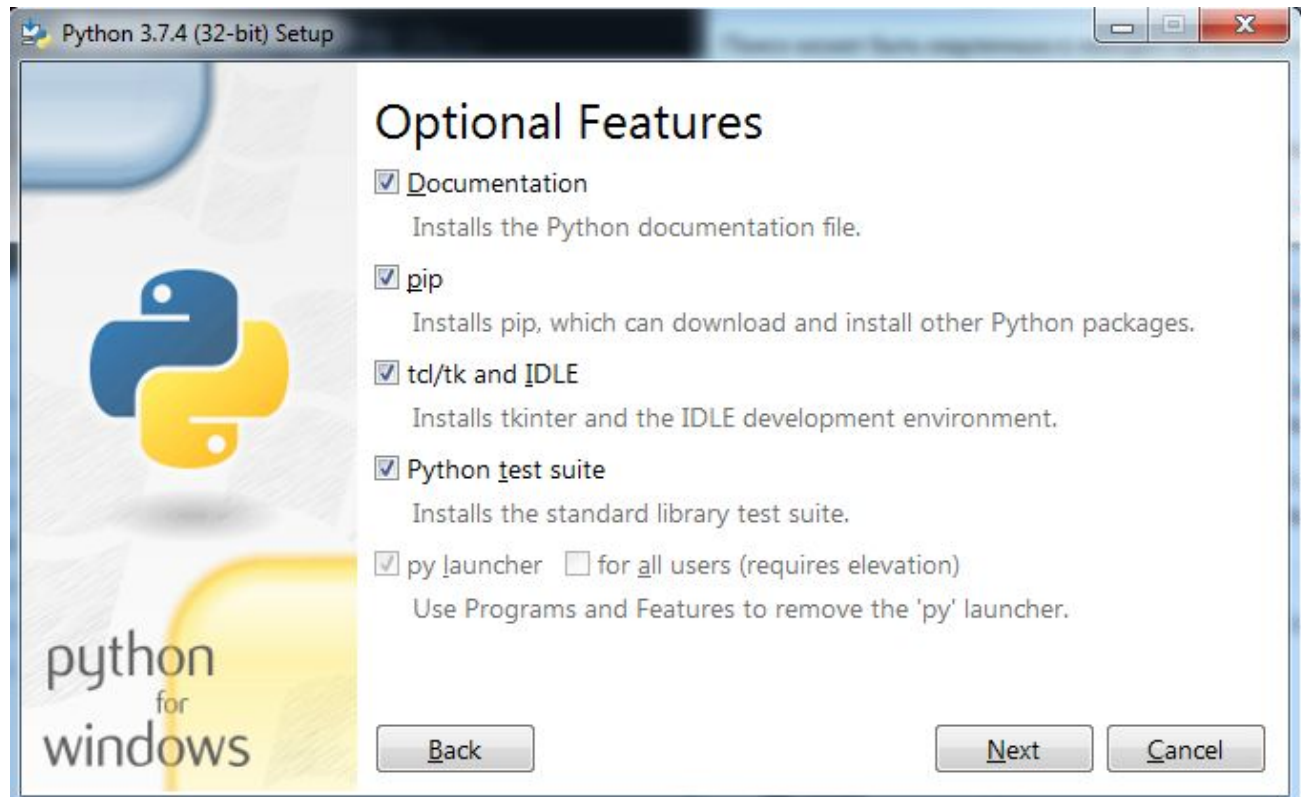
Python 2.0 дар соли 2000 нашр шуд ва хусусиятҳои навро, аз қабили фаҳмидани рӯйхат ва ҳисобкунии истинод ба партовҳоро ҷорӣ кард ва бо версияи 2.7.18 дар соли 2020 қатъ карда шуд.

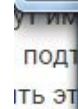
Python 3.0 дар соли 2008 нашр шуд ва ислоҳи асосии забон буд, ки ба Python 2 комилан мувофиқ нест.

Насби забони барномасозии Python 3



Насби забони барномасозии Python 3





Саволҳои санҷишӣ



1. Алгоритм гуфта дар барномарезӣ чиро меноманд?
2. Код чист?
3. Псевдокод чист?

Вазифаи хонагии № 1



Псевдокоди ҳалли масъалаҳои зерин навишта шавад.

Begin1. Тарафи квадрат дода шудааст. Периметри он $p = a * 4$ ёфта шавад.

Begin3. Тарафҳои росткунча a ва b дода шудаанд. Масоҳати он $s = a * b$ ва периметри он $p = 2 * (a + b)$ ёфта шаванд.

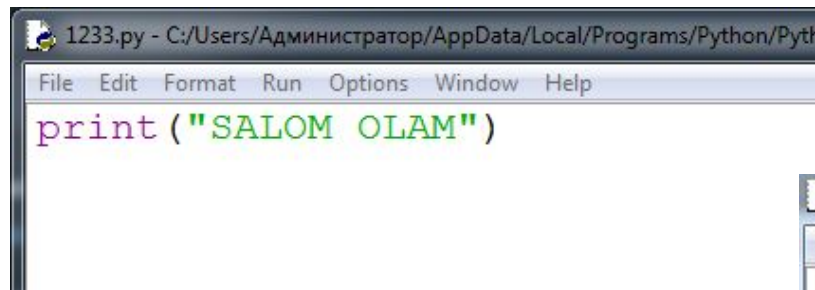
Аз вазифаи хонагии №2 сар карда ҳалли ҳамаи вазифаҳои хонагӣ дар намуди коди забони барномасозии Python 3 бояд иҷро карда шавад. Ба ғайр аз ҳолатҳое, ки алоҳида қайд мешаванд.



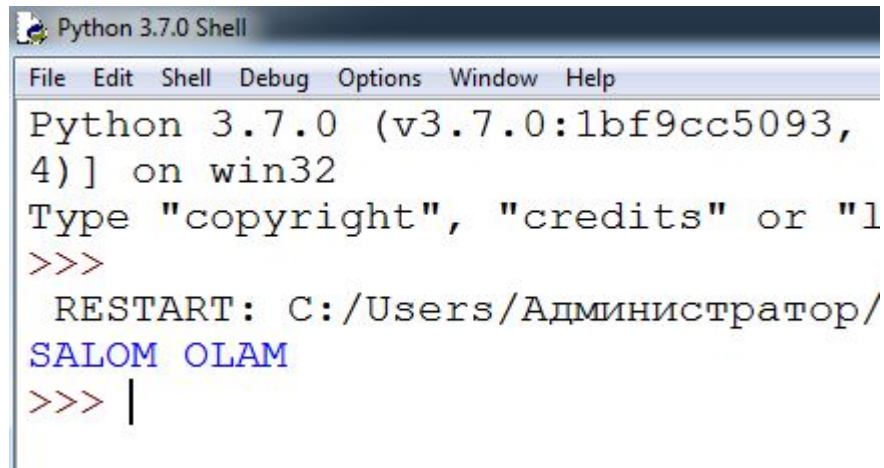
Мавзӯи 2

1. Барномаи аввалин
2. Намудҳои маълумот
3. Тағйирёбандаҳо
4. Амалҳои арифметикӣ

Барномаи аввалин



A screenshot of a Python IDE window. The title bar reads "1233.py - C:/Users/Администратор/AppData/Local/Programs/Python/Pyth...". The menu bar includes "File", "Edit", "Format", "Run", "Options", "Window", and "Help". The code editor contains a single line of Python code: `print("SALOM OLAM")`.



A screenshot of the "Python 3.7.0 Shell" window. The menu bar includes "File", "Edit", "Shell", "Debug", "Options", "Window", and "Help". The output text shows the Python version and platform: "Python 3.7.0 (v3.7.0:1bf9cc5093, 4)] on win32". It then prompts the user: "Type 'copyright', 'credits' or '1'". After the user enters a command (likely the filename), the output shows "RESTART: C:/Users/Администратор/" followed by the printed result "SALOM OLAM" in blue text. The prompt ">>>" is followed by a vertical cursor bar.

Тағйирёбандаҳо



Тағйирёбандаҳо барои нигоҳ доштани ягон қимат ба монанди матн ё адад ва ё ягон объекти мураккабтар пешбинӣ шудаанд. Тағйирёбанда дар рафти иҷрои скрипт метавонад якчанд маротиба истифода шавад ва қимати худро иваз кунад. Дар Python тағйирёбанда дар чунин намуд муайян карда мешавад

Номи_Тағйирёбанда = Қимати_Тағйирёбанда

Мисол:

a = 24

Тағйирёбандаҳо

Тағйирёбандаҳо дар Python эълон карда намешаванд. Танҳо номи тағйирёбанда ва аломати бахшиши қиматро (=) мегузорем ва ба он қимати заруриро мебахшем

Мисол

```
File Edit Format Run Op
a = 24
print(a)
```

Натиҷа

```
///
=====
24
>>>
```

Тағйирёбандаҳо

Ин қоидае, ки дар боло дидем барои сатрҳо ё рамзҳо низ истифода мешавад.

Мисол

```
File Edit Format Run Options
#барои сатрҳо
a = "Салом"
print(a)
#барои рамзҳо
b = "с"
print(b)
```

Натиҷа

```
=====
Салом
с
>>>
```

Навъҳои тағйирёбандаҳо



- **int** (бутун)
- **float** (ҳақиқӣ)
- **list** (Массив)
- **str** (аломатҳои сатри)
- **bool** (ададҳои мантиқӣ)



Намуна

Ададҳои бутун

Int (адад)

55

157

1235456

Ададҳои ҳақиқӣ

float(адад)

5.5

6.7

5.0

Сатр

str(сатр)

"Salom"

"Салом python "

Ададҳои мантиқӣ

Bool(мантиқӣ)

True

False



Амалҳои арифметикӣ

`+, -, *, /, %, =`

Амалҳои муносибат (муқоиса)

`<, >, <=, >=, !, !=, ==`

Амалҳо

Ранг	Амалҳо	Ассотсиативият
1	() [] {}	→
2	* / % (амалҳои бинарӣ)	→
3	+ - (амалҳои бинарӣ)	→

Саволҳои санчишӣ



1. Тағйирёбанда чист?
2. Тағйирёбандаро барои чӣ истифода мебаранд?
3. Қоидаҳои эълон кардани номи тағйирёбандаро номбар кунед.

Вазифаи хонагии № 2



Аз вазифаи хонагии №2 сар карда ҳалли ҳамаи вазифаҳои хонагӣ дар намуди коди забони барномасозии Python 3 бояд иҷро карда шавад. Ба ғайр аз ҳолатҳое, ки алоҳида қайд мешаванд.

Integer4. Ададҳои бутуни мусбати a ва b дода шудаанд ($a > b$). Дар порчаи дарозиаши a миқдори максималии имконпазири порчаҳои дарозиашон b ҷойгир карда шудаанд (болои ҳам нахобанда). Амали гирифтани бақияи бутун аз ҳосили тақсими истифода бурда, дарозии қисми ишғолнашудаи порчаи a -ро ёбед.

Integer6. Адади дурақама дода шудааст. Аввал рақами чапи онро (даҳиҳо) ва сипас рақами ростии онро (воҳидҳо) ҷоп кунед. Барои ёфтани даҳиҳо амали тақсим, барои ёфтани воҳидҳо — амали гирифтани бақия аз ҳосили тақсим истифода бурда шавад.



Мавзӯи 3

1. Ифодаҳои мантиқӣ
2. Операторҳои AND, OR, NOT
3. Оператори if-else

Python 3 барои навкорон

Оператор	Тавсиф	Мисол
==	Баробар будани ҳарду операндро месанҷад. Агар ҳа, пас шарт дуруст мешавад.	5 == 5 ба True оварда, мерасонад, True == False ба False оварда мерасонад
!=	Баробар будани ҳарду операндро месанҷад. Агар не, пас шарт дуруст мешавад.	12 != 5 ба True оварда мерасонад. False != False ба False оварда мерасонад.
<>	Баробар будани ҳарду операндро месанҷад. Агар не, пас шарт дуруст мешавад.	12 <> 5 ба True оварда мерасонад. Ба монанди оператори !=
>	Месанҷад, ки оё қимати операнди чап аз қимати ростии он калонтар аст. Агар ҳа, пас шарт дуруст мешавад.	5 > 2 ба True оварда мерасонад. True > False ба True оварда мерасонад. "A" > "B" ба False оварда мерасонад.

Python 3 барои навкорон

<	Месанҷад, ки оё қимати операнди чап аз қиммати рости он хурдтар аст. Агар ҳа, пас шарт дуруст мешавад.	3 < 5 ба True оварда мерасонад. True < False ба False оварда мерасонад. "A" < "B" ба True оварда мерасонад.
>=	Месанҷед, ки оё қимати операнд аз чап ба қиммати рости он калон аст ё баробар аст. Агар ҳа, пас шарт дуруст мешавад.	1 >= 1 ба True оварда мерасонад. 2 >= 3.2 ба True оварда мерасонад. "C" >= "D" ба False оварда мерасонад.
<=	Аз арзиши операнди рост камтар ё баробар ба арзиши операнди чапро месанҷад. Агар ҳа, пас шарт дуруст мешавад.	4 <= 5 ба True оварда мерасонад. 0 <= 0.0 ба True оварда мерасонад. -0.001 <= -36 ба False оварда мерасонад.

Мисолҳо



```
x = 5
```

```
y = 8
```

```
print("x == y:", x == y)
```

```
print("x != y:", x != y)
```

```
print("x < y:", x < y)
```

```
print("x > y:", x > y)
```

```
print("x <= y:", x <= y)
```

```
print("x >= y:", x >= y)
```

Натижа:

x == y: False

x != y: True

x < y: True

x > y: False

x <= y: True

x >= y: False

Оператори or



Агар ҳадди ақалл яке аз операндҳо рост бошад, он гоҳ тамоми ифода дуруст хоҳад буд.

Мисол:

A	B	A or B
True	True	True
True	False	True
False	True	True
False	False	False

Оператори and

Шарт дуруст хоҳад буд, агар ҳарду операнд рост бошанд.

Мисол:

A	B	A and B
True	True	True
True	False	False
False	True	False
False	False	False

Оператори not

Қиммати мантиқии операндро баръакс мекунад.

A	Not A
True	False
False	True

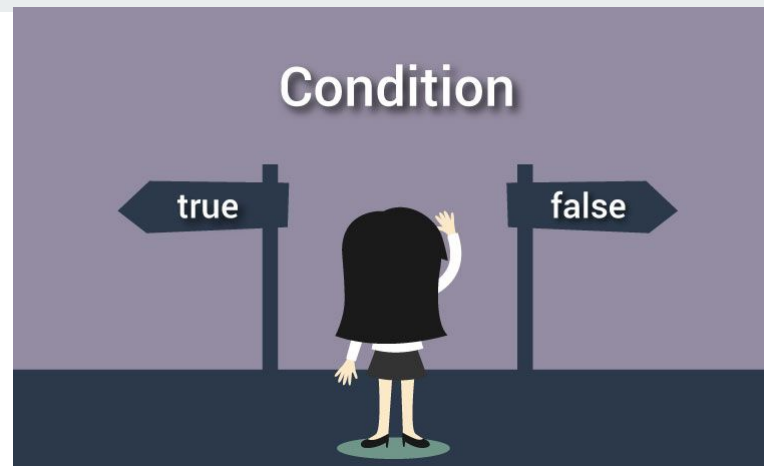
Оператори if-else

Синтаксиси оператори if else чунин аст:

```
if ифодаи_шартӣ:  
    оператори_1  
else :  
    оператори_2
```

Дар ин ҷо, ифодаи_шартӣ бояд доимӣ ё тағйирёбандаи ададӣ бошад. Агар қимати он ғайринулӣ бошад, онгоҳ оператори_1 иҷро мешавад. Дар акси ҳол оператори_2 иҷро карда мешавад. Мисол:

```
a = int(input("Адади бутун дохил кунед: "))  
#a = 5  
if a > 0:  
    print("Адади дохил шуда, адади мусбат")  
else:  
    print("Адади дохил шуда, адади манфӣ")
```



Саволҳои санҷишӣ

1. Оператори if-ро барои чӣ истифода мебаранд?
2. Қимати ин ифодаҳоро ёбед. TRUE ё FALSE?
A. $(4 * 5 == 22)$
B. $(48 / 6 != 6)$

Вазифаи хонагии № 3



If4(Abramyan). Се ададҳои бутун дода шудаанд. Миқдори ададҳои мусбат дар байни онҳо ёфта шавад.

If5(Abramyan). Се ададҳои бутун дода шудаанд. Миқдори ададҳои мусбат ва манфӣ дар байни онҳо ёфта шавад.



Мавзӯи 4

1. Сикли for
2. Сикли while

Мафҳуми цикл



Агар лозим шавад, ки ягон қисми томи код якчанд маротиба такрор ёбад операторҳои сиклиро истифода мебаранд. Дар Python ду намуди операторҳои сикли мавҷуданд: `for`, `while`. Барои операторҳои сиклиро хуб истифода бурда тавоништан мафҳумҳои `TRUE` ва `FALSE`-ро доништан зарур аст.

Сикли for



Сикли for як итератори универсалӣ дар Python аст. For дар сатрҳо, рӯйхатҳо, кортежҳо ва дигар такроршавандаҳои сохта, инчунин объектҳои нави муайянкардаи корбар кор мекунад. Синтаксиси сикли for дар Python чунин аст:

for мақсад in объект:

оператори_1

Else:

оператори_2

Мисолҳо

1

```
for i in [1,2,3,4]:
```

```
    print(i)
```

```
Else:
```

```
    print('Сикл ба охир расид.')
```

Натиҷа:

1

2

3

4

Сикл ба охир расид

2

```
for i in range(1, 5):
```

```
    print(i)
```

```
Else:
```

```
    print('Сикл ба охир расид.')
```

Натиҷа:

1

2

3

4

Сикл ба охир расид

Мисолҳо

3

```
for i in 'abcd':
```

```
    print(i, end = " ")
```

Натиҷа:

a b c d

4

```
for i in range(1,11):
```

```
    if n % 2 == 0:
```

```
        print(i, end = " ")
```

Натиҷа:

2 4 6 8 10

Сикли while



Сикли while ба шумо имкон медиҳад, ки то даме ки шарт санҷидашаванда дуруст аст, ҳамон як пайдарпаии амалҳоро иҷро кунад. Синтаксиси while чунин аст

```
while (ифодаи_шартӣ):  
    Тани_сикл
```

Ҳангоми иҷрои сикли while аввал шарт тафтиш карда мешавад. Агар он иҷро нашавад (нодуруст бошад), пас иҷрои сикл қатъ карда мешавад ва ба дигар сохтори баъд аз сикли while гузаронида мешавад. Агар шарт дуруст бошад, пас тани сикл иҷро карда мешавад, ки пас аз он шарт боз санҷида ва тани сикл боз иҷро карда мешавад.

Мисолҳо



```
n = 1
while n <= 5:
    print("Сикл ",n, "маротиба иҷро шуд")
    n += 1
```

Натиҷа:

```
Сикл 1 маротиба иҷро шуд
Сикл 2 маротиба иҷро шуд
Сикл 3 маротиба иҷро шуд
Сикл 4 маротиба иҷро шуд
Сикл 5 маротиба иҷро шуд
```


Бози “Ёфтани адади тасодуфӣ” бо сикли while

```
import random as r
a = r.randrange(1,100)
n = int(input())
while n != a:
    if n > a:
        print("Калон")
    Else:
        print("Хурд")
    n = int(input())
else :
    print("Шумо ададро ёфтед!")
```

Натиҷа:
1
хурд
98
калон
50
калон
25
калон
12
хурд
Шумо ададро ёфтед!

Саволҳои санҷишӣ



1. Оператори синкли барои чӣ истифода бурда мешавад?
2. Фарқи синкли for аз while дар чӣ аст?

Вазифаи хонагии № 4



For2. Ду ададҳои бутуни a ва b ($a < b$) дода шудаанд. Бо тартиби афзуншавӣ ҳамаи ададҳои бутуни дар байни a ва b (аз он ҷумла a ва b) мавҷуд буда ва миқдори ин ададҳо n -ро аз ҷоп бароред.

While4. Адади бутуни n (> 0) дода шудааст. Агар он дараҷаи адади 3 бошад, онро `true`, вагарна `false` ҷоп карда шавад.



Мавзӯи 5

1. Рӯйхатҳо (list)
2. Индексҳо
3. Бурриш (slice)

Рӯйхатҳо (list)



Рӯйхат ин сохтори маълумот барои нигоҳдории объектҳои намудҳои гуногун мебошад.

Андозаи рӯйхат статикӣ нест, онро тағир додан мумкин аст. Рӯйхат аз рӯи табиати худ як навъи тағирёбанда мебошад.

Тағирёбандае, ки ҳамчун рӯйхат муайян карда мешавад, дорои ишора ба сохторе дар хотира мебошад, ки дар навбати худ ишораҳо ба ягон объект ё сохтори дигар дорад.

Намуна

- Объекти намудашон якхела

```
list1 = ["apple", "banana", "cherry"]
```

```
list2 = [1, 5, 7, 9, 3]
```

```
list3 = [True, False, False]
```

- Объекти намудашон гуногун

```
list4 = ["abc", 34, True, 40, "male"]
```

Индексҳо



- Функцияи index методи дохилии рӯйхат мебошад, ки имкон медиҳад индекс ё мавқеи элементро дар пайдарпай донем.
- Ба ибораи дигар, ин метод аъзои рӯйхатро ҷустуҷӯ мекунад ва индекси онро бар мегардонад.

```
myList = ['1', '11', 1, 'a', 'x', 1.1]
```

```
print(myList.index(1))
```

```
=====RESTART: .....name.py =====
```

```
2
```

```
>>>
```

Бурриш (slice)




Функсияи `slice()` объектро (парчаеро) бармегардонад, ки метавонад барои бурида гирифтани қисми сатр, рӯйхат, `tuple` (кортеж) ва ғ. истифода шавад. Дар ин ҷо `slice` (буриш, бурида гирифтани) ба маънои бурида гирифтани ё ки ҷудо карда гирифтани ягон қисми (парчаи) объекти ибтидоӣ истифода мешавад. Дар ин маврид объекти ибтидоӣ тағйир намеёбад. Яъне сохтор ва мӯҳтавои объекти ибтидоӣ пештара боқӣ мемонад ва объекти нав сохта мешавад.

Тарзи дигари гирифтани `slice` ё ки бурида гирифтани қисми объект чунин аст:

`объект[ибтидо:интиҳо:қадам]`

Намуна



```
py_string = 'Python'
```

```
slice_object = slice(3)
```

```
print(py_string[slice_object])
```

```
slice_object = slice(1, 6, 2)
```

```
print(py_string[slice_object])
```

```
=====RESTART: .....name.py =====
```

```
Pyt
```

```
yhn
```

```
>>>
```

Саволҳои санҷишӣ



1. Мақсад аз истифодаи функцияҳо дар чист?
2. List - ро барои чӣ истифода мебаранд?
3. Функцияи slice() чӣ вазифаро иҷро мекунад?

Вазифаи хонагии № 5

Array4(Abramyan). Адади бутуни n (> 1), инчунин аъзои якум a_1 ва махраҷ d -и *прогрессияи геометрӣ* дода шудаанд. Массиви андозааш n , ки аз n аъзои аввали ин прогрессия: иборат аст, ташкил карда шуда, аз чоп бароварда шавад.

Array7(Abramyan). Массиви андозааш n дода шудааст. Элементҳои он аз баръаксаш аз чоп бароварда шавад.



Мавзӯи 6

1. Луғат (dictionary)
2. Методҳои кор бо луғат

Луғат (dictionary)

Луғатҳо барои нигоҳ доштани “калидҳо” ва қиматҳои додаҳо дар намуди ҷуфтҳои истифода мешаванд. Дар мисоли поёни калид “brand” ва қимати он “Ford” аст.

Намуна:

```
thisdict = {  
    "brand": "Ford",  
    "electric": False,  
    "year": 1964,  
    "colors": ["red", "white",  
    "blue"]  
}
```

Методҳои кор бо луғат



- Шумораи аъзоҳо `len()`
- Навъи маълумот `type()`

Методҳои кор бо луғат

Шумораи элементҳои луғатро чоп кунед:

```
thisdict = {  
    "brand": "Ford",  
    "model": "Mustang",  
    "year": 1964  
}
```

```
print(len(thisdict))
```

```
=====RESTART: .....name.py =====
```

```
3
```

```
>>>
```

Методҳои кор бо луғат

Навъи тағйирёбандаро чоп кунед:

```
thisdict = {  
    "brand": "Ford",  
    "model": "Mustang",  
    "year": 1964  
}
```

```
print(type(thisdict))
```

```
=====RESTART: .....name.py =====
```

```
<class 'dict'>
```

```
>>>
```


Саволҳои санҷишӣ

1. Методҳои кор бо луғатро номбар кунед.
2. Функсия `len()` чиро бармегардонад?

Вазифаи хонагии № 6

Case4(Abramyan). Рақами тартибии моҳ – адади бутуни аз 1-12 (1 – январ, 2 – феврал ва ғайра) дода шудааст. Миқдори рӯзҳо дар ин моҳ барои соли кабиса (высококосный) набуда муайян карда шавад.


Case5(Abramyan). Амалҳои арифметикӣ бо ададҳо чунин рақамгузорӣ карда шудаанд: 1 – ҷамъ, 2 – фарқ, 3 – зарб, 4 – тақсим. Рақами тартибии амал n дода шудааст (адади бутун аз 1-4) ва ададҳои ҳақиқии a ва b (b ба 0 баробар нест) дода шудаанд. Бо ин адаҳо амали нишондодашударо иҷро намуда, қимати ҳосилшударо аз ҷоп бароред.



Мавзӯи 7

1. Маҷмӯъ (set)
2. Методҳои кор бо маҷмӯъ

Маҷмӯъ чист?



Пеш аз омӯхтани маҷмӯъҳо дар Python, биёед аввал мафҳуми маҷмӯъҳоро дар математика ба ёд орем. Ба таври оддӣ карда гӯем, маҷмӯъ "як даста унсурҳои муайян" мебошад. Мо метавонем Диаграммаи Веннро барои ифодаи муносибати байни маҷмӯаҳо истифода барем.

Дар Python маҷмӯъ - ин пайдарпайии такрорнаёбандаи элементҳо (унсурҳо) мебошад, яъне ҳар як элементи `set` такрор намеёбад аст ва онро тағир дода намешавад.

Аммо, худ маҷмӯа (`set`) тағирёбанда аст. Ба ибораи дигар, мо метавонем аъзоҳоро дар маҷмӯъ илова ва хориҷ кунем. Python бисёр усулҳои (методҳои) дарунсохти идораи маҷмӯаҳоро фароҳам меорад, мо баъдтар ин усулҳоро меомӯзем.

Маҷмӯъ чист?

Маҷмӯаҳо пайдарпаии нумураттаб ва ғайри индексатсияшаванда мебошанд. Дар Python, маҷмӯаҳо дар қавсҳои чингила(фигурави) навишта мешаванд.

```
thisset = {"set", "list", "tuple"}
```

```
print(thisset)
```

Ҷавоб: {'set', 'tuple', 'list'}

Дастрасӣ ба унсурҳо (элементҳо)

Шумо ба унсурҳои маҷмӯъҳо аз рӯи индекс дастрасӣ пайдо карда наметавонед. Аммо шумо метавонед бо истифода аз `for`(сикл) аз маҷмӯъ гузаред ё фаҳмед, ки оё қимат дар маҷмӯъ бо истифода аз `in` оператор аст.

Биёед ҳар як унсури маҷмӯъро нишон диҳем:

```
thisset = {"set", "list", "tuple"}  
for x in thisset:  
    print(x)
```

Ҷавоб:

```
set  
list  
tuple
```

Методҳои кор бо маҷмӯъ



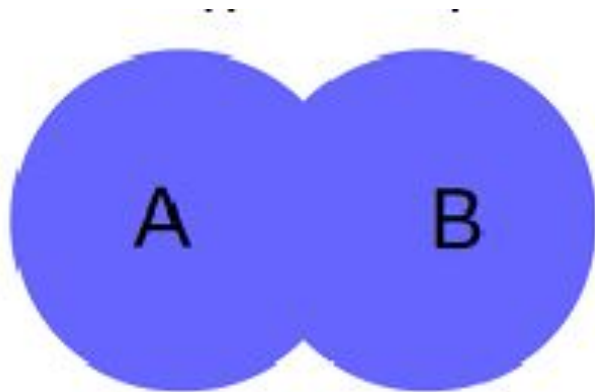
- `add(x)` -> Илова кардани `x` дар `set`
- `clear()` -> Хориҷ кардан аз `set`
- `copy()` -> Нускаи `set` бармегардонад
- `x.difference(y)` -> Маҷмӯъи элементҳоро бар мегардонад, ки дар `x` бошанд, аммо дар `y`
- `x.difference_update(y)` -> Элементҳое, ки дар `x` ва `y` мебошанд, хориҷ мекунад
- `discard(x)` -> Элементи муайяншударо нест мекунад
- `x.intersection(y)` -> Маҷмӯъро, ки буриши `x` ва `y` мебошад, бармегардонад
- `intersection_update(y)` -> Унсурҳои маҷмӯъро, ки дар `y` дигар дода нашудаанд, нест мекунад
- `x.isdisjoint(y)` -> `True` агар `x` ва `y` ягон унсури умумӣ надошта бошанд

Методҳои кор бо маҷмӯъ



- `x.issubset(y)` -> True агар ҳамаи унсурҳо аз `x` дар `y` бошанд
- `issuperset()` -> True агар ҳамаи элементҳо аз `y` дар `x` бошанд
- `pop()` -> Унсури охиринро тоза мекунад ва бармегардонад
- `remove()` -> Элементи муайяншударо нест мекунад
- `x.symmetric_difference(y)` -> Маҷмӯъи унсурҳоро, ки ҳангоми `x` ва `y` буриш намекунанд, бармегардонад
- `symmetric_difference_update()` -> Элементҳоро илова мекунад, ки дар маҷмӯъи дигар нестанд
- `union()` -> Маҷмӯъҳои сершуморро дар бар мегирад
- `x.update(y, z)` -> Маҷмӯъаҳои сершуморро бо `x` навиштааст

Мисоли маҷмӯъҳо дар Python 3



Маҷмӯъ

-> `A = {"a", "b", "c"}`

`B = {"f", "d", "a"}`

Python 3

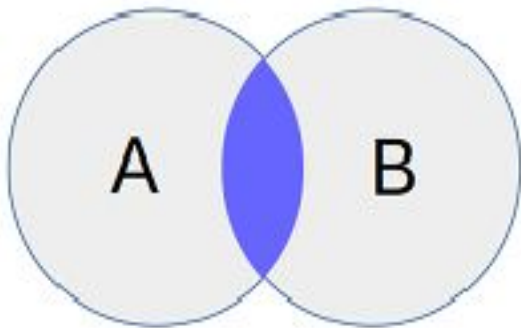
-> `javob = A.union(B)`
`javob = A | B`

`print(javob)`

$A = \{a, b, c\}$ $B = \{f, d, a\}$

$A \cup B = \{a, b, c, d, f\}$

Мисоли маҷмӯъҳо дар Python 3



$A = \{a, b, c\}$ $B = \{f, d, a\}$

$A \cap B = \{a\}$

Маҷмӯъ

->

```
A = {"a", "b", "c"}
```

```
B = {"f", "d", "a"}
```

Python 3

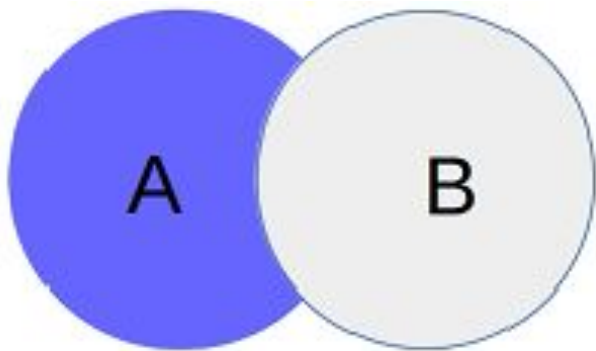
->

```
javob = A.intersection(B)
```

```
javob = A & B
```

```
print(javob)
```

Мисоли маҷмӯъҳо дар Python 3



Маҷмӯъ

Python 3

$A = \{a, b, c\}$ $B = \{f, d, a\}$

$A / B = \{b, c\}$

-> `A = {"a", "b", "c"}`

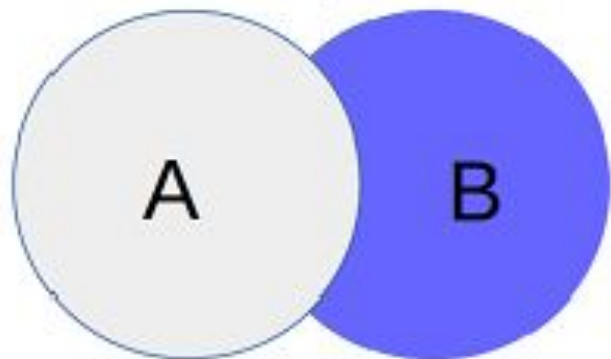
`B = {"f", "d", "a"}`

`javob = A.difference(B)`

-> `javob = A - B`

`print(javob)`

Мисоли маҷмӯъҳо дар Python 3



Маҷмӯъ

Python 3

$A = \{a, b, c\}$ $B = \{f, d, a\}$

$B / A = \{f, d\}$

-> `A = {"a", "b", "c"}`

`B = {"f", "d", "a"}`

`javob = B.difference(A)`

-> `javob = B - A`

`print(javob)`

Саволҳои санҷишӣ



1. Дар Python маҷмӯъ чист ва тарзи навишташ чӣ гуна аст?
2. Оё дар Python маҷмӯъро тағйир дода метавонем?

Вазифаи хонагии № 7

Array5(Abramyan). Адади бутуни n (> 2) дода шудааст. Массиви ададҳои бутуни андозааш n , ки аз n аъзои аввали пайдарпаии ададҳои Фибоначчи $f_k: f_1 = 1, f_2 = 1, f_k = f_{k-2} + f_{k-1}, k = 3, 4, \dots$ иборат аст, ташкил карда шуда, аз чоп бароварда шавад.

Array9(Abramyan). Массиви ададҳои бутуни андозааш n дода шудааст. Ҳамаи ададҳои ҷуфти массивро бо тартиби камшавии индексҳояшон ва миқдори онҳо k аз чоп бароварда шаванд.



Мавзӯи 8

1. Мафҳуми функция
2. Истифодаи функцияҳои стандартии математикӣ
3. Истифодаи функцияҳои стандартии сатрӣ ва сана

Мафҳуми функция



Функцияҳо барои ҷудо намудани барнома ба қисмҳои нисбатан мустақил истифода мешаванд. Функцияҳо татбиқи бисёр доранд. Барномаи мураккаб ба якчанд қисмҳои нисбатан содда тақсим карда мешавад ва барои ҳар як ин қисми функцияи алоҳида тартиб додан мумкин аст.

Масалан, дар барнома метавонад ягон қисми кода мавҷуд бошад, ки он дар дахҳо ҷои ин барнома такрор меёбад. Агар ҳамин қисми такрорёбандаи кодро дар намуди функцияи алоҳида нависем ва ин функцияро дар ҷойҳои даркорӣ истифода барем, ҳаҷми коди барнома ихтисор мешавад, намуди он соддатар мегардад.

Истифодаи функсияҳои стандартии математикӣ



```
import math as m
def my_function(x):
    x = m.fabs(x)
    y = m.sqrt(x)
    y = m.exp(m.sin(y) + 1)
    return y

print(my_function(2))
```

```
import math as m

def my_func ( a, b, c ):
    p = (a + b + c) / 2
    s = m.sqrt(p * (p - a) * (p - b) * (p - c))
    return s

a = int (input ("a = "))
b = int (input ("b = "))
c = int (input ("c = "))

print(my_func(a,b,c))
```

Гузариш аз радиан ба дараҷа ва баръакс



<code>degrees(x)</code>	Арзиши кунчи x -ро аз радианҳо ба градус табдил медиҳад.
<code>radians(x)</code>	Арзиши кунчи x -ро аз дараҷа ба радиан табдил медиҳад.

Функсияҳои тригонометрӣ



<code>acos(x)</code>	<code>atan(x)</code>	<code>cos(x)</code>	<code>sin(x)</code>
<code>asin(x)</code>	<code>atan2(y, x)</code>	<code>hyp(x, y)</code>	<code>tan(x)</code>

Истифодаи функцияҳои стандартӣ



Коди аломатро бо истифодаи функцияи `ord` муайян кардан мумкин аст. Ин функция сатро ҳамчун вуруд қабул мекунад, ки он бояд маҳз аз як аломат иборат бошад. Функция коди ин аломатро бар мегардонад.

Масалан, `ord` ('A') = 65 бармегардад.

Барои ба коди ададии додашуда бозгардонидани рамзи ба он мувофиқ функцияи `chr` истифода мешавад.

Яъне функцияи `chr()` ба функцияи `ord()` баръаск аст.

```
print(ord('A'))
```

Ҷавоб: 65

```
print(chr(65))
```

Ҷавоб: A

Истифодаи функцияҳои стандартӣ

Барои муайян кардани шумораи элементҳои ягон тағйирёбанда Python методҳои дохилии ба монанди `my_list.length()` ё `my_string.length` надорад. Ба ҷои ин функция `len()` истифода мешавад: `len(my_list)` ва `len(my_string)`.

```
a = ["Welcome", "to", "Python"]
```

```
print(len(a))
```

Ҷавоб: 3

Истифодаи функцияҳои стандартӣ



```
s = str(input())
```

```
print(s.capitalize()) #Аввалин ҳарфи матнро калон мекунад;
```

```
s = "SAloM PyTHON"
```

```
print(s.casefold()) #Ҳама ҳарфҳои калонро ба ҳарфҳои хурд табдил медиҳад;
```

```
Ҷавоб:salom python
```

Tuple

Tuple - Ин функсия ба вазифаи рӯйхат хеле монанд аст, ба истиснои он, ки ба ҷои рӯйхатҳо кортежҳо сохта мешавад.

```
>>> numbers = [2, 1, 3, 4, 7]
```

```
>>> tuple(numbers)
```

Ҷавоб : (2, 1, 3, 4, 7)

Истифодаи функцияҳои стандартии сана



<code>date</code>	санаро нишон медиҳад
<code>time</code>	вақтро дар бар мегирад
<code>datetime</code>	вақт ва сана дар бар мегирад
<code>timedelta</code>	як давраи мушаххаси вақтро тавсиф мекунад, ки дар байни ду лаҳзаи гуногун ҷойгир аст
<code>tzinfo</code>	маълумоти мухталифи минтақаи вақтро нишон медиҳад
<code>timezone</code>	вақтро бо истифодаи стандарти UTC тасвир мекунад

Истифодаи функсияҳои стандартии сана



Дастӣ дохил кардан

```
import datetime
```

```
a = datetime.date(2001, 10, 28)
```

```
print(a)
```

Ҷавоб: 2001-10-28

Соати компютерро ҳудаш бо
тарзи автоматӣ мегирад

```
import datetime
```

```
a = datetime.date.today()
```

```
print(a)
```

Ҷавоб: 2018-11-08

Истифодаи функсияҳои стандартии сана



```
import datetime
```

```
a = datetime.time(12, 18, 35, 5867)
```

```
print(a)
```

Ҷавоб: 12:18:35.005867

```
import datetime
```

```
c = datetime.datetime(2017, 7, 18, 4, 52, 33)
```

```
print(c)
```

```
print(type(c))
```

Ҷавоб: 2017-07-18 04:52:33

```
<class 'datetime.datetime'>
```

Истифодаи функсияҳои стандартии сана



```
import datetime
```

```
a = datetime.datetime(2007, 2, 13)
```

```
b = datetime.datetime(2013, 10, 25, 12, 8, 47)
```

```
print(a)
```

```
print(b)
```

```
2007-02-13 00:00:00
```

```
2013-10-25 12:08:47
```

```
import datetime
```

```
a = datetime.datetime.today()
```

```
b = datetime.datetime.now()
```

```
print(a)
```

```
print(b)
```

```
2018-11-08 11:12:53.256261
```

```
2018-11-08 11:12:53.256270
```

Истифодаи функсияҳои стандартии сана



```
import datetime
```

```
a = datetime.datetime.today().strftime("%d.%m.%Y")
```

```
b = datetime.datetime.today().strftime("%H:%M:%S")
```

```
print(a)
```

```
print(b)
```

08.11.2018

11:41:04

Саволҳои санҷишӣ



1. Функсия чист?
2. Кадом функсияҳои стандартии Python - ро медонед.

Вазифаи хонагии № 8

Proc4. Функцияи навъаш void-и *TrianglePS(a, p, s)* муайян карда шавад, ки аз рӯи тарафи a -и секунҷаи баробартараф периметри он $p = 3 \cdot a$ ва масоҳати он $s = a^2 \cdot \sqrt{3} / 4$ -ро ҳисоб мекунад (ҳамаи параметрҳо ададҳои ҳақиқӣ). Бо ёрии ин функция периметр ва масоҳати се секунҷаҳои баробартараф аз рӯи тарафи додашуда ҳисоб карда шаванд.

Proc8. Функцияи навъаш void-и *AddRightDigit(d, k)* муайян карда шавад, ки ба адади бутуни мусбати k рақами d -ро аз тарафи рост илова мекунад (k – параметри ададии бутун, d – рақам). Бо ёрии ин функция пайдарпай ба адади додашудаи k рақамҳои додашудаи d_1 ва d_2 аз тарафи рост илова карда шуда, натиҷаи ҳар як иловакунӣ аз ҷоп бароварда шавад.

Дар ин ҷо void ба он маъно истифода мешавад, ки худӣ функция бевосита қимат бознамегардонад.



Мавзӯи 9

1. Сохтани функцияҳо
2. Функцияҳои `def` ва `return`
3. Функцияи `lambda`

Мафҳуми функция



Функцияҳо барои ҷудо намудани барнома ба қисмҳои нисбатан мустақил истифода мешаванд. Функцияҳо татбиқи бисёр доранд. Барномаи мураккаб ба якчанд қисмҳои нисбатан содда тақсим карда мешавад ва барои ҳар як ин қисми функцияи алоҳида тартиб додан мумкин аст.

Масалан, дар барнома метавонад ягон қисми кода мавҷуд бошад, ки он дар дахҳо ҷои ин барнома такрор меёбад. Агар ҳамин қисми такрорёбандаи кодро дар намуди функцияи алоҳида нависем ва ин функцияро дар ҷойҳои даркорӣ истифода барем, ҳаҷми коди барнома ихтисор мешавад, намуди он соддатар мегардад.

Функсияҳо, дастури def

Функсия дар python объектест, ки далелҳоро қабул мекунад ва қиматро бар мегардонад. Одатан, функсия бо истифодаи дастури def муайян карда мешавад.

Биёед функсияи соддатаринро муайян кунем:

```
def add(x, y):  
    return x + y
```

Дар дастури return гуфта мешавад, ки арзиши функсия баргардонида мешавад. Дар ҳолати мо, функсия суммаи x ва y-ро бармегардонад.

Дар ин функсия `add` номи функсия аст.

Истифодаи функция



Ҳоло мо метавонем функцияи `add`-ро даъват кунем:

```
>>> add(1, 10)
```

```
11
```

```
>>> add('abc', 'def')
```

```
'abcdef'
```

Функсияҳои беном, дастури lambda

Агар ба функсия иҷрои миқдори зиёди рамз лозим набошад, шумо метавонед функсияи беномро истифода баред. Дар ин ҳолат ба мо функсияи lambda лозим мешавад.

Функсияҳои беном метавонанд танҳо як ибораро дар бар гиранд, аммо онҳо зудтар мебошанд. Функсияҳои беном бо истифода аз дастури lambda сохта мешаванд. Илова бар ин, онҳо набояд ба тағирёбанда таъин карда шаванд, чунон ки мо бо дастури `def func ()` кардем:

Функсияҳои беном, дастури lambda

Мисол барои сохтани функсияи lambda:

```
multiple = lambda a, b: a * b
```

```
multiple(7, 12)
```

Чунин функсия ном надорад, аммо онро ба тағирёбанда таъин кардан мумкин аст, ки онро дар оянда бояд ҳамчун функсияи оддӣ номид.

Саволҳои санҷишӣ



1. Фарқи lambda-функсия аз функсияҳои бо def муайяншаванда дар чист?
2. lambda-функсия дар кадом маврид истифода бурда мешаванд?

Вазифаи хонагии № 9

Проц6. Функцияи навъаш void-и *DigitCountSum(k, c, s)* муайян карда шавад, ки миқдори рақамҳои c -и адади бутуни мусбати k ва суммаи онҳо s -ро ҳисоб мекунад (ҳамаи параметрҳо ададҳои бутун). Бо ёрии ин функция миқдор ва суммаи рақамҳо барои ҳар як панҷ ададҳои бутуни додашуда ҳисоб карда шаванд.

Проц5. Функцияи навъаш void-и *RectPS(x1, y1, x2, y2, p, s)* муайян карда шавад, ки периметр ва масоҳати росткунҷаи тарафҳояш ба тирҳои координата параллел аз рӯи координатаҳои қуллаҳои муқобилхобидаи $(x1, y1)$ ва $(x2, y2)$ -ро ҳисоб мекунад (ҳамаи параметрҳо ададҳои ҳақиқӣ). Бо ёрии ин функция периметр ва масоҳати се росткунҷаҳои бо қуллаҳои муқобилхобида додашуда ҳисоб карда шаванд.

Дар ин ҷо void ба он маъно истифода мешавад, ки худӣ функция бевосита қимат бознамегардонад.



Мавзӯи 10

1. Модулҳо ва библиотекаҳо

Сохтани бозӣ бо модули tkinter



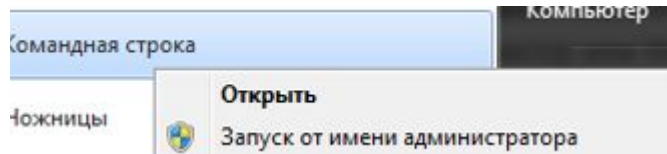
tkinter модули графикӣ мебошад, ки ба шумо имкон медиҳад барномаҳоро бо оинаи интерфейсӣ эҷод кунед. Ин модул интерфейси забони машҳури барномасозии ва асбоби TKL / TK TK-ро ташкил медиҳад.

tkinter ба монанди tcl/tk як модули кросс аст ва метавонад дар аксар системаҳои маъмулӣ истифода шавад (Windows, Linux, Mac OS X ва ғайра). Аз python-3.0, модул мувофиқи PEP 8 ба tkinter (бо ҳарфи хурд) номгузорӣ шудааст.

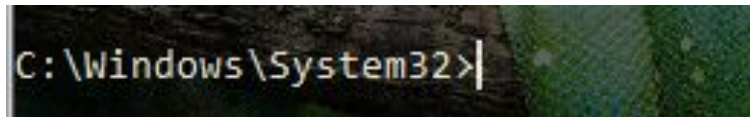
Насби модули tkinter

Пеш аз ҳама мо бояд модули tkinter - ро насб кунем.

Барои ин :

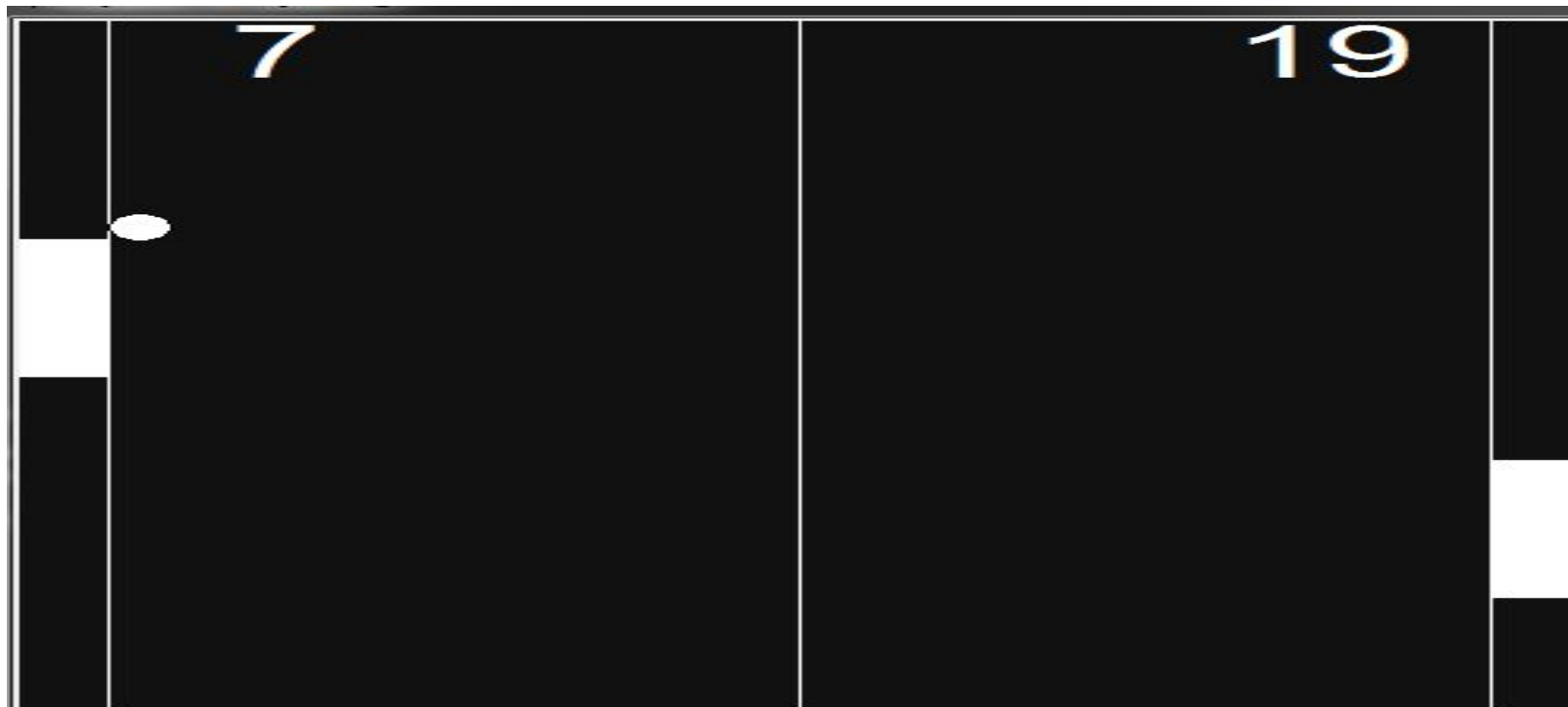


“Запуск от имени администратора”-ро интихоб менамоем. Бояд дар сатри фармон(командная строка) чунин пайдо мешавад:



Баъди ин “pip install tkinter”-ро ба ин ворид мекунем:

Сохтани бози “Пинг-понг”



Саволҳои санҷишӣ



1. Якчанд пакетҳо номабар кунед ва таъиноти онҳоро гӯед.
2. Барои эълони истифодаи пакет кадом калимаи махсус истифода бурда мешавад?

Вазифаи хонагии № 10

Matrix5. Ададҳои бутуни мусбати m ва n , инчунин адади d ва маҷмуи иборат аз m ададҳо дода шудаанд. Матритсаи андозааш $m \times n$, ки сутуни якуми он ба ададҳои маҷмуи додашуда (бо тартиби додашавایی ададҳо дар маҷмуъ) баробар буда, элементҳои сутунҳои оянда ба суммаи элементҳои мувофиқи сутуни пешина ва адади d баробар аст, ташкил карда шавад (дар натиҷа, элементҳои ҳар як сатри матритса прогрессияи арифметикиро ташкил медиҳанд).

Matrix7. Матритсаи андозааш $m \times n$ ва адади бутуни k ($1 < k < m$) дода шудаанд. Элементҳои сатри k -уми матритсаи додашуда аз ҷоп бароварда шавад.



Мавзӯи 11

1. Хатогиҳо ва истисноҳо
2. Коркарди истисно
3. Оператори try-except

Хатогиҳо

Дар ҳама гуна барномаҳо, махсусан калон, хатогиҳо рух дода метавонанд, ки боиси корношоямии ин барномаҳо мегардад. Яъне барнома он чизеро, ки бояд иҷро кунад, иҷро намекунад. Интерпретатори Python ҳангоми дучор шудан бо ибораи хато намедонад, ки онро чӣ гуна тафсир кунад. Аз ин рӯ, он иҷрои барномаро қатъ мекунад ва ҷои хатогиро нишон медиҳад:

```
>>> 1a = 10
```

File "<stdin>", line 1

1a = 10


^SyntaxError: invalid syntax

Хатогиҳо



Дар истилоҳоти Python, дар ин ҷо истисно ба синфи `SyntaxError` мансуб доништа шуд. Мувофиқи ҳуҷҷатҳои Python, хатогиҳои синтаксисро одатан ҳамчун хато меноманд ва ҳамаи дигарон - ба истисноҳо. Дар баъзе забонҳои барномасозӣ калимаи "истисно" истифода намешавад ва хатогиҳо ба синтаксисӣ ва семантикӣ тақсим карда мешаванд. Вайрон кардани семантика одатан маънои онро дорад, ки гарчанде ки ибораҳо аз нигоҳи синтаксиси забон дуруст навишта шуда бошанд ҳам, барнома мувофиқи чашмдошт кор намекунад.

Истисноҳо (exceptions)



Вақте ки дар барнома дар раванди навишти код ё санҷиши он хатогиҳо пайдо мешаванд, он код аз тарафи барномасоз ислоҳ карда мешавад, то хатогиҳо рух надиҳанд. Аммо, амали корбар аксар вақт боиси он мегардад, ки дар барнома истисно ба миён меояд.

Масалан, барнома воридкунии рақамро интизор аст, аммо шахс ҳарф ворид мекунад. Кӯшиши табдил додани он ба рақам ба истисноии `ValueError` оварда мерасонад ва барнома (садамавӣ) анҷом дода мешавад.

Дар ин ҳолат, дар забонҳои барномасозӣ, аз ҷумла да Python як оператори махсус вучуд дорад, ки ба шумо имкон медиҳад, ки истисноҳоро, ки ба миён меоянд, онро чунин коркард мекунад, ки барнома кори худро давом диҳад ё кори худро дуруст иҷро кунад.

Оператори try-except

```
n = input("Адади бутун ворид кунед: ")
```

try:

```
n = int(n)
```

```
print("Бо муваффақият")
```

except:

```
print("Чизе хато кард")
```

2 натиҷаҳоро дида мебароем

Ин натиҷа ҳангоми ворид кардан ба "b"

```
Адади бутун ворид кунед: b
Чизе хато кард
```

Ин натиҷа ҳангоми ворид кардан ба "2"

```
Адади бутун ворид кунед: 2
Бо муваффақият
```

Саволҳои санчишӣ



1. Якчанд пакетҳо номабар кунед ва таъиноти онҳоро гӯед.
2. Барои эълони истифодаи пакет кадом калимаи махсус истифода бурда мешавад?

Вазифаи хонагии № 11



1. Адад дода шудааст. Try - ехсерт - ро истифода бурда, барномае нависед, ки ҳангоми тақсими адад ба 0(сифр) хатогии “division by zero” барорад.



Мавзӯи 12

1. Кор бо файлҳои матнӣ
2. Хондан (read)
3. Сабт (write)
4. Илова намудан (append)

Кор бо файлҳои матнӣ



Пеш аз хондан аз файл ва сабт намудан ба файл, он бояд кушода шавад. Барои кушодани файл дар Python, функсияи `open()` истифода мешавад. Ҳангоми даъват ин функсия як объекти нави файло эҷод мекунад, ки он дар оянда кушода шавад.

Тарзҳои кор бо файл

read, r	Кушодани файл танҳо барои хондан.
write, w	Кушодани файл танҳо барои сабт, агар ин файл набошад онро месозад.
append, a	Илова намудани матн (муҳтаво) ба охири файл.

Хосиятҳои объектҳои файл дар python.

file.closed	Агар файл баста шуда бошад, онгоҳ True бармегардонад.
file.mode	Тарзи дастрасӣ ба файли кушодашуда.
file.name	Номи файло бармегардонад.

Хондани файл (read)

Шумо метавонед файло дар Python хонед.

Бо даъват кардани файли бо типи (.txt) дар "ҳолати хондан" (r).

Мисоли истифодабарӣ:

#файли тайёро бо ин тарз кушода мешавад;

```
with open('foo.txt', 'r') as fp:
```

```
    kushodan = fp.read()
```

```
    print(kushodan)
```

Хондани файл (read)

Ин методи файли `file.read()` хондани методи файл аз `size`(дарози) байт, вобаста нест. Агар ба охир расидани файли пеш аз гирифтани андозаи муайяншудаи байтҳо расидааст, пас метод танҳо байтҳоро дастрас менамояд.

- **file** - объекти файл
- **size** - **int**, миқдор байтҳо

Барои хондани маълумоти файл `f.read` -ро истифода мебаранд ва онро дар мундариҷаи тағирёбанда барои хондани файлҳо дар Python нигоҳ доранд.

```
tak = f.read()
```

Барои чоп кардан дар экран аз `print()` истифода мебарем.

Хондани файл (read)

```
with open('dars1.txt', 'r') as fp:
```

```
# Хондани тамоми мӯҳтавои файл
```

```
    tagi = fp.read()
```

```
for azsar in tagi.splitlines():
```

```
    print(azsar)
```

Ҷавоб:

Salom olam

Salom dunyo

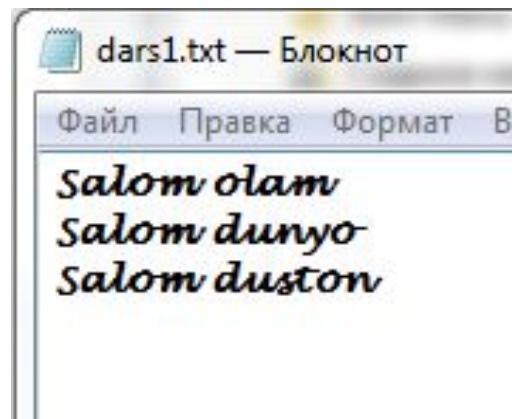
Salom duston

Мисол: Ин даруни файл

Salom olam

Salom dunyo

Salom duston



Сабт ба файл (write)

Синтаксис:

`file.write(str)`

Имконот:

- `file` - объекти файл
- `str` - сатр (string), маълумоти ба файл сабтшаванда
- `int` - миқдори байтҳо

Тавсиф:

Методи `file.write ()` сатри `str` -ро ба файл сабт мекунад.

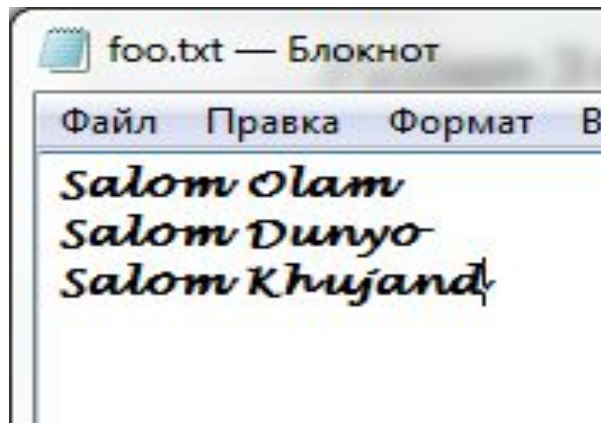
Аз сабаби вучуд доштани буфер, то он даме, ки метод `file.flush()` ё `file.close()` даъват карда нашавад, сабти сатр дар файл ба итмом расида ҳисобида намешавад. Яъне барои пурра ба итмом расидани раванди сабт ба файл бояд метод `file.flush ()` ва ё `file.close ()` даъват карда шавад.

Сабт ба файл (write)

```
with open('foo.txt', 'w') as fp:
```

```
    fp.write('Salom Olam\nSalom\nDunyo\nSalom Khujand')
```

```
fp.close() #Пушидани файл
```



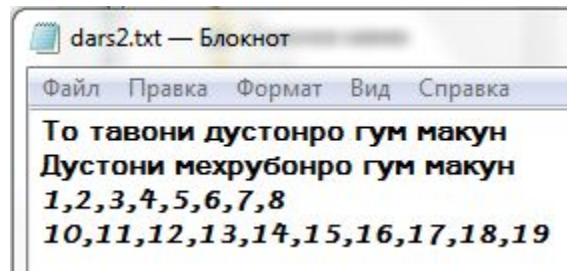
```
tagi = 'То тавони дустонро гум  
макун\nДустони меҳрубонро гум  
макун\n1,2,3,4,5,6,7,8\n10,11,12,13,14,15,1  
6,17,18,19'
```

```
# Navistan
```

```
with open('dars2.txt', 'w') as tp:  
    tp.write(tagi)
```

```
# khondan fayl
```

```
with open('dars2.txt', 'r') as tp:  
    print(tp.read())
```



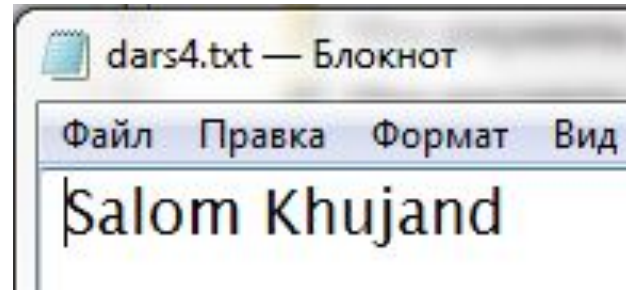
Илова ба файл (append)

Агар аломати плюс (+) баъди рамзи дастрасии файл гузошта шавад, он гоҳ дар сурати мавҷуд набудани файл автоматӣ файли нав сохта мешавад.

```
my_file = open("dars4.txt", 'a+')
```

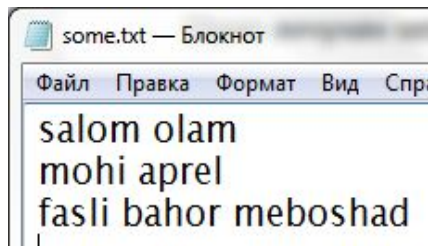
```
my_file.write("Salom Khujand")
```

```
my_file.close()
```



Илова ба файл (append)

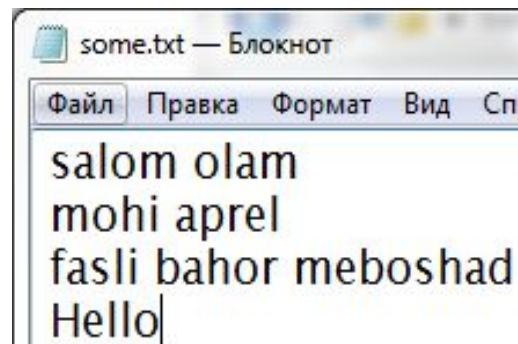
Шумо инчунин метавонед матни навро ба файли аллакай мавҷудбуда илова кунед.



```
my_file = open("some.txt", 'a')
```

```
my_file.write("Hello")
```

```
my_file.close()
```



Тоза кардани файл (seek)

Агар мо файли сабт шударо кушоем ва намедонем дар кучо ҷойгир аст. Шояд мо аллакай баъзе маълумотҳоро сабт кардем. Мо метавонем танҳо нишонро ба ибтидо ҳаракат кунем ва онро пӯшем. Дар ин ҳолат, ҳуҷҷат холӣ хоҳад буд.

Дар ин мисол, кашфиёт махсусан дар речаи мураккаб анҷом дода шуд. Пас аз пӯшида, ҳатто агар дар файл маълумот мавҷуд бошад, онҳоро нест мекунамд.

```
f = open('dars5.txt', 'w+')
```

```
f.seek(0) # Холи кардан файли дар боло омада.
```

```
f.close()
```

Саволҳои санҷишӣ



1. Тарзҳои кор бо файло номбар кунед.
2. Речай “а+” аз речай “а” чӣ фарқият дорад?

Вазифаи хонагии № 12



File1. Сатри s дода шудааст. Агар s-ро номи файл гузоштан мумкин бошад, пас файли холӣ бо ҳамин ном сохта шавад ва TRUE чоп карда шавад. Агар бо номи s ном гузоштан мукин набошад, пас FALSE чоп карда шавад.

File5. Дар файл ададҳои бутун сабт шудаанд. Миқдори элементҳое, ки дар ин файл мавҷуд аст, ёфта шавад. Агар файл бо ин ном мавҷуд набошад, пас -1 чоп карда шавад.



Мавзӯи 13

1. Маълумоти ибтидои оиди синфҳо (ООР)
2. Инкапсулятсия, полиморфизм, ворисшавӣ
3. Мисолҳои инкапсулятсия ва ворисшавӣ

OOP (Object-Oriented Programming)



Забони барномасозии Python имконият медиҳад, ки бо методологияҳои гуногун барномарезӣ карда шавад. Аммо асоси ҳуди Python усули объективӣ аст. Аз ин рӯ дар Python дар услуби OOP кор кардан осон аст.

Class чист?



Class - дар барномасозии ба объект нигаронидашуда, як қолаби сохтани объектҳое мебошад, ки қиматҳои ибтидоии ҳолатҳоро таъмин мекунанд: бахшидани қиматҳои ибтидоии аъзо-тағирёбандаҳо ва амалисозии рафтори аъзо-функсияҳо ё аъзо-усулҳо.

Намунаи синф (class)

```
class person:
```

```
    name = ""
```

```
    age = 0
```

```
    def set(self,name,age):
```

```
        self.name=name
```

```
        self.age=age
```

```
person_1 = person()
```

```
person_2 = person()
```

```
person_1.name = "Ali"
```

```
person_1.age = 20
```

```
person_2.set("Vali",22)
```

```
print("Name person ",person_2.name)
```

```
print("age person ",person_2.age)
```

```
print("Name person ",person_1.name)
```

```
print("age person ",person_1.age)
```

Натиҷа:

```
===== RESTART: Python38-32/class.py =====
```

```
Name person  Vali
```

```
age person  22
```

```
Name person  Ali
```

```
age person  20
```

```
>>>
```

Инкапсулятсия ва ворисшавӣ:



Намуна:

```
class person:
    name = ""
    age = 0
    def _setname(self,name):
        self.name=name
    def __setage(self,age):
        self.age=age
```

```
class student(person):
    kurs=0

student_1 = student()
student_1._person__setage(20)
student_1._setname("Ali")
student_1.kurs=1;

print("Name student ",student_1.name)
print("age student ",student_1.age)
print("kurs ",student_1.kurs)
```

Натиҷа:

```
===== RESTART: Python38-32/q.py =====
Name person Ali
age person 20
kurs student 1
>>>
```

Саволҳои санҷишӣ



Вазифаи хонагии № 13





Мавзӯи 14

1. Аз скрипти Python 3 пайваст шудан бо базаи маълумоти MySQL
2. Мисолҳои истифодаи SELECT, INSERT, UPDATE, DELETE

Кор бо базаи маълумоти MySQL дар Python 3



Барои дастрасӣ ба пойгоҳи додаҳои MySQL аз Python, ба шумо драйвери пойгоҳи додаҳо лозим аст. MySQL MySQL Connector / Python як роҳи стандартии пойгоҳи додаҳо мебошад, ки аз ҷониби MySQL пешниҳод шудааст.

Барои ҳамроҳ шудани Python бо MySQL модули `mysql.connector` насб мекунем:

```
pip install mysql.connector
```

Пайваст кардани Python бо MySQL

Пеш аз ҳама бояд пойгоҳи дода (database, база данных) созем дар системаи идоракунии додаҳо MySQL:

```
CREATE DATABASE python
```

Баъд тавассути коди зерин аз скрипти Python бо базаи маълумот пайваст мешавем:


```
conn = mysql.connector.connect(host='localhost', database='python',  
user='root',password='')
```

Сохтани ҷадвал тавассути python



```
import mysql.connector as mysql
db = mysql.connect(
    host="localhost",
    user="root",
    password="",
    database="python")
cursor = db.cursor()
cursor.execute("DROP TABLE IF EXISTS EMPLOYEE")
sql = "CREATE TABLE EMPLOYEE ( FNAME CHAR(20) NOT NULL, LNAME CHAR(20),
AGE INT )"
cursor.execute(sql)
db.close()
```

Истифодаи дархости SELECT



```
import mysql.connector
mydb = mysql.connector.connect(
    host="localhost",
    user="root",
    password="",
    database="python"
)
mycursor = mydb.cursor()
mycursor.execute("SELECT * FROM t_python")
myresult = mycursor.fetchall()
for x in myresult:
    print(x)
```

Истифодаи дархости INSERT



```
import mysql.connector
```

```
mydb = mysql.connector.connect(  
    host="localhost",  
    user="root",  
    password="",  
    database="python")
```

```
mycursor = mydb.cursor()  
sql = "INSERT INTO t_python (nom, shahr) VALUES (%s, %s)"  
val = ("John", "Конибодом")  
mycursor.execute(sql, val)  
mydb.commit()  
print(mycursor.rowcount, "record inserted.")
```

Истифодаи дархости UPDATE



```
import mysql.connector
```

```
mydb = mysql.connector.connect(  
    host="localhost",  
    user="root",  
    password="",  
    database="python")
```

```
mycursor = mydb.cursor()  
sql = "UPDATE t_python SET shahr = 'конибодом' WHERE shahr = 'New York'"  
mycursor.execute(sql)  
mydb.commit()  
print(mycursor.rowcount, "record(s) affected")
```

Истифодаи дархости DELETE



```
import mysql.connector
```

```
mydb = mysql.connector.connect(  
    host="localhost",  
    user="root",  
    password="",  
    database="python")
```

```
mycursor = mydb.cursor()  
sql = "DELETE FROM t_python WHERE shahr = 'New York'"  
mycursor.execute(sql)  
mydb.commit()  
print(mycursor.rowcount, "record(s) affected")
```


Саволҳои санҷишӣ



Вазифаи хонагии № 14





Мавзӯи 15

1. `pycurl`
2. `requests`

pycurl




Насби pycurl

Барои Windows, пеш аз истифодаи PyCURL дар барномаҳои шумо, бояд якчанд вобастагӣ насб карда шавад.

Агар шумо версияи расмии Python-ро истифода баред (яъне шумо версияи Python -ро аз сайти расмии <https://www.python.org> зеркашӣ кардед), инчунин pip, ба шумо лозим аст, ки фармони зеринро дар сатри фармон иҷро кунед ва насб хоҳад шуд иҷро карда шавад:

```
pip install pycurl
```

Python 3 барои навкорон



```
import pycurl
from io import BytesIO
b_obj = BytesIO()
crl = pycurl.Curl()
# Арзиши URL-ро таъин кунед
crl.setopt(crl.URL,
'https://wiki.python.org/moin/BeginnersGuide')
# Байте нависед, ки utf-8 рамзгузорӣ шудааст
crl.setopt(crl.WRITEDATA, b_obj)
# Интиқоли файло иҷро мекунад ва малумотро аз сервер мегирад
crl.perform()
# Сеанс curl ба охир расад
crl.close()
# Мазмуни дар объекти BytesIO Ҳифзшударо гиред (бо аломатҳои байтӣ)
get_body = b_obj.getvalue()
# Байтҳои дар get_body Ҳифзшударо ба HTML рамзкушоӣ кунед ва
чоп кунед
result
print('Output of GET request:\n%s' % get_body.decode('utf8'))
```

Саволҳои санҷишӣ



Вазифаи хонагии № 15





Мавзӯи 16

1. Маълумоти ибтидоӣ оиди барномасозии асинхронӣ
2. Намунаҳо

Барномаҳои асинхронӣ



Барномасозии асинхронӣ дар Python торафт маъмултар мегардад. Барои истифодаи асинхрон пакетҳои гуногун ҳастанд, ки яке аз онҳо **Asyncio** мебошад, ки библиотекаи стандартии Python 3.4 мебошад.

Барои фаҳмидани кори барномаи асинхронӣ ба мо лозим аст, ки чӣ тавр коркардани ҷараён (thread, поток) дар протсесорро дида бароем.

Функсияи оддӣ (Синхронӣ)



```
from time import sleep
```

```
def my_func(add):
```

```
    for a in range (1, 5):
```

```
        sleep(1)
```

```
        print(add, a)
```

```
my_func("hello")
```

```
my_func("world")
```

```
hello 1
```

```
hello 2
```

```
hello 3
```

```
hello 4
```

```
world 1
```

```
world 2
```

```
world 3
```

```
world 4
```

Функсияи асинхронӣ

```
from time import sleep
```

```
import asyncio
```

```
async def spider(stt):
```

```
    for a in range(1, 4):
```

```
        await asyncio.sleep(1)
```

```
    print(stt, a)
```

```
spiders = [
```

```
    asyncio.ensure_future(spider("Blog")),
```

```
    asyncio.ensure_future(spider("News")),
```

```
    asyncio.ensure_future(spider("Forum"))
```

```
]
```

```
event_loop = asyncio.get_event_loop()
```

```
event_loop.run_until_complete(asyncio.gather(*spiders))
```

```
event_loop.close()
```

Функсияи асинхронӣ

Натиҷа:

Blog 1
News 1
Forum 1
Blog 2
News 2
Forum 2
Blog 3
News 3
Forum 3

Саволҳои санҷишӣ



Вазифаи хонагии № 16



Маводи истифодабурдашуда

1. <https://docs.python.org/3/tutorial/> | Санаи дастрасӣ: 2021-04-12
2. <https://oftob.com/tj> | Санаи дастрасӣ: 2021-04-11
3. <https://coderoad.ru/31501806> | Санаи дастрасӣ: 2021-04-12
4. <https://docs-python.ru/tutorial/metody-fajlovogo-obekta-potoka-python/> |
Санаи дастрасӣ: 2021-04-12
- 5.