

Задаци методе листе**047 Употреба методе append**

У празну листу корисник уноси имена све док не унесе неку посебну реч. Користити методу append

```
lista = []
ponovo = 'da'
while ponovo == 'da':
    ime = input('Unesi ime: ')
    lista.append(ime)
    print('Da li treba dodati jos jedno ime?')
    ponovo = input('da ili bilo sta drugo za ne: ')
    print()
print('Ovo su unesena imena:')
for ime in lista:
    print(ime)
```

048 Сабирање вредности у листи

Корисник не зна колико ће целобројних вредности да унесе у листу. Написати програм који сабира све унешене вредности у листу и приказује добијени збир

```
A = []
jos = "d"
i = 0
zbir = 0
while jos == "d":
    x = int(input("A(" + str(i) + ") = "))
    A.append(x)
    i += 1
    zbir += x
    jos = input("Da li da treba uneti jos brojeva (d/n)? ")

print("Lista", A, "ima zbir unetih brojeva u listu", zbir)
```

049 Удруживање елемената две листе

Дате су две листе: A = [0, 1, 2] и B = [-1, 0, 2]. У трећу листу поставити вредност која се налазе у обе листе

```
A = [0, 1, 2]
B = [-1, 0, 2]
C = []
for i in range(3):
    for j in range(3):
        if A[i] == B[j]:
            C.append(A[i])
            break

print(C)
```

Питања и задаци за самосталан рад

Задаци

013 Коришћењем метода листе написати програм који омогућава кориснику да унесе целе бројеве као елементе листе, избацује из листе све негативне бројеве а затим приказује листу.

014 Елементе торке (4, 0, 3) унети у листу [-5, 100, 200, -40, 77] на позиције које су исте као њихове вредности (елемент 0 на позицију 0, елемент 3 на позицију 3...). Елементи листе на тим позицијама се одстрањују из листе.

015 Из листе [5, 5, 6, 0, 1, 2, 5, 2, 1] избацити све поновљене вредности.

List Comprehension

Омогућава краћи код када је потребно креирати нову листу на основу вредности из већ постојеће листе.

Синтакса: `nova_lista = [izraz for brojac in sekvenca if uslov == True]`

Као излаз се добија нова листа док секвенца остаје непромењена.

У синтакси, `uslov` је филтер којим се задржавају само вредности којима је услов испуњен.

Исказ `list comprehension` каже, за сваку вредност у секвенци која испуњава дати услов сместити у нову листу резултат израза као елемент те листе.

Израз може бити резултат математичке операције, логичке операције или податак било којег типа.

050 Употреба просте list comprehension

У нову листу сместити вредности из дате листе које су дељиве са 5

```
brojevi = [5, 6, 10, 15, 16]
nova_lista = [x for x in brojevi if x % 5 == 0]
print(nova_lista)
```

051 Употреба сложеније list comprehension

У нову листу сместити квадрате вредности из дате листе које су дељиве са 5

```
brojevi = [5, 6, 10, 15, 16]
nova_lista = [x ** 2 for x in brojevi if x % 5 == 0]
print(nova_lista)
```

052 Селектовање унетих вредности

Корисник уноси целе бројеве. У две различите листе сместити посебно позитивне и негативне бројеве. После завршетка уноса бројева приказати садржај листа и број унешених нула

```
pozitivni = []
negativni = []
broj_nula = 0
jos = "d"
while jos == "d":
    x = int(input("Uneti nov ceo broj: "))
    if x > 0:
```

```

        pozitivni.append(x)
elif x < 0:
    negativni.append(x)
else:
    broj_nula += 1
jos = input("Da li da treba uneti jos brojeva (d/n)? ")

```

```

print("Uneti pozitivni brojevi su", pozitivni)
print("Uneti negativni brojevi su", negativni)
print("Uneto je", broj_nula, "nula")

```

053 Селектовање и измена вредности

Из листе [5, -3, 2, -10, 0, 2, -2] прво избацити вредности дељиве са 5 а преостале негативне вредности променити у 0

```

lista = [5, -3, 2, -10, 0, 2, -2]
nova_lista = []
for i in range(len(lista)):
    print(lista[i])
    if lista[i] < 0 and lista[i] % 5 != 0:
        nova_lista.append(0)
    elif lista[i] % 5 != 0 or lista[i] == 0:
        nova_lista.append(lista[i])
    else:
        pass

```

```

print(nova_lista)

```

054 Селектовање вредности према типу

У нову листу сместити само целе бројеве из листе ["ja", 0, -4, "-4", 3.05]

```

lista = ["ja", 0, -4, "-4", 3.05]
nova_lista = []
for i in range(len(lista)):
    if type(lista[i]) == type(int()):
        nova_lista.append(lista[i])

```

```

print(nova_lista)

```

055 Селектовање стрингова из листе

Корисник уноси стрингове у листу. Прихватити само оне стрингове који су дугачки између 5 и 10 знакова

```

lista = []
jos = "d"
while jos == "d":
    nov_string = input("Uneti nov string: ")
    if len(nov_string) >= 5 and len(nov_string) <= 10:

```

```

    lista.append(nov_string)
    jos = input("Da li da treba uneti jos stringova (d/n)? ")

print(lista)

```

056 Селектовање вредности преко методе index

Коришћењем методе index открити све индексе појављивања вредности 2 у листи [0, 2, 1, 2, 0, 2, 2].

```

lista = [0, 2, 1, 2, 1, 2, 2]
indeksi = []
i = 0
while True:
    if lista[i] == 2:
        indeksi.append(lista.index(2))
        lista[i] = 0
    i += 1
    if i == len(lista):
        break

```

```
print(indeksi)
```

Питања и задаци за самосталан рад

Задаци

- 016 Корисник пуни листу са 4 броја. Коришћењем list comprehension сместити позитивне целе бројеве (као и нуле) у једну а негативне целе бројеве у другу листу.
- 017 Ученици у одељењу се обележавају бројевима од 1 до 10. Ученици који иду на веронауку су 3, 5, 8. Ученици који су девојчице су 1, 3, 7, 8, 9, 10. Који ученици су дечаки и иду на грађанско?
- 018 Корисник уноси целе бројеве у листу. Креирати нову листу са вредностима „да“ или „не“ у зависности од тога да ли је унети цео број или квадрат или куб неког од једноцифрених целих бројева.
- 019 Највећу вредност у корисничкој листи целих бројева сместити на почетак листе, затим парне бројеве по растућем редоследу (као и 0), затим најмању вредност у листи а затим непарне вредности по опадајућем редоследу.
- 020 Дата је листа са презименом фудбалера, средњом оценом игре са две децимале и његов број голова у сезони ['fudbaler1', 4.34, 2, 'fudbaler2', 5.11, 3, 'fudbaler3', 3.76, 2]. Раздвојити податке по листама са презименима, средњим оценама и головима. Приказати податке о појединачном фудбалеру по редовима.