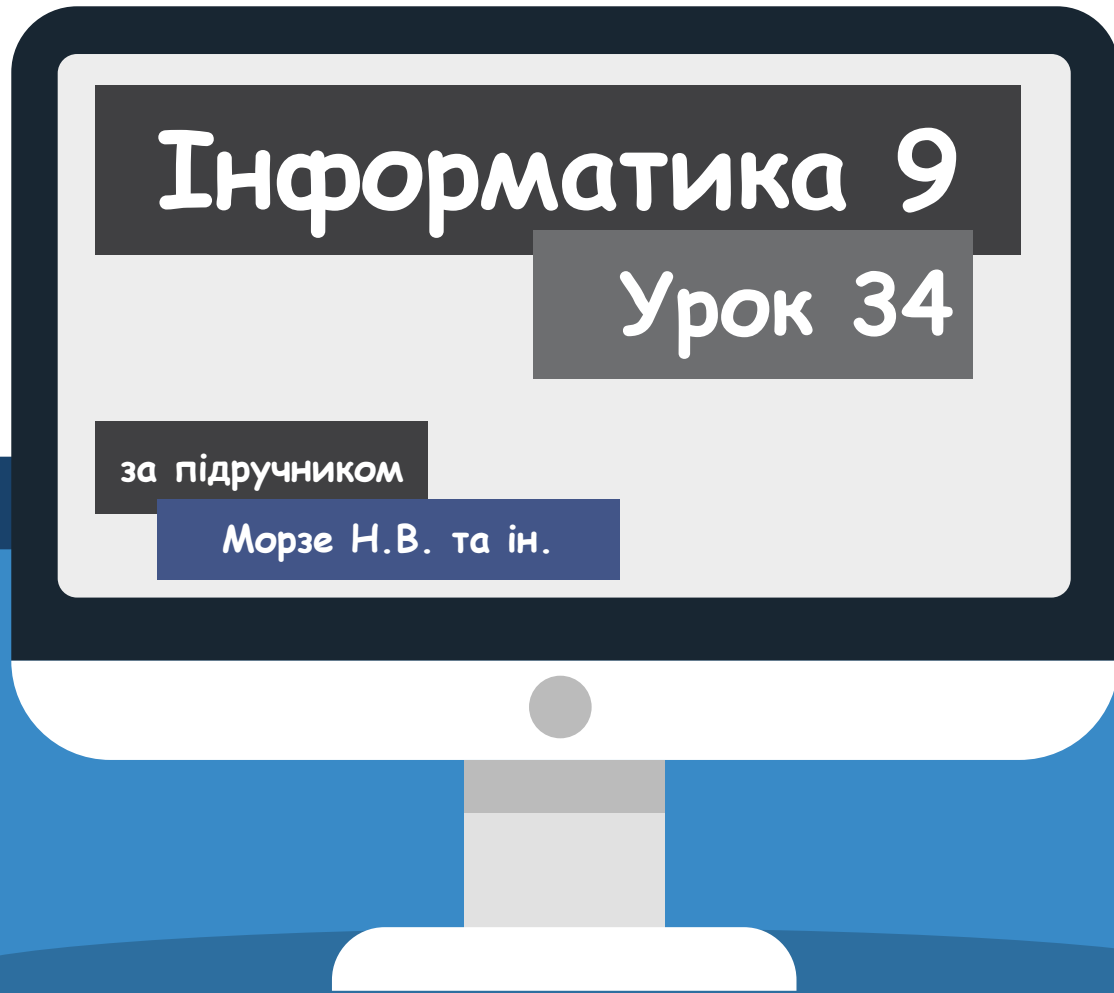


# Дії над списками у мові програмування Python



За навчальною програмою 2017 року



# Які дії можна виконувати над списками в мові Python?

**Над списками можна виконувати прості дії:**

- **об'єднання,**
- **дублювання списків,**
- **отримання фрагменту списку,**
- **визначення кількості елементів списку,**
- **додавання до списку нового елемента.**



# Які дії можна виконувати над списками в мові Python?

Подібні операції ми виконували над **рядками**. Але на відміну від рядків, **списки** — це змінювані послідовності.

Якщо уявити **рядок** як об'єкт у пам'яті, то коли над ним виконуються операції об'єднання і повторення, цей рядок не змінюється, а в результаті операції створюється інший рядок в іншому місці пам'яті.



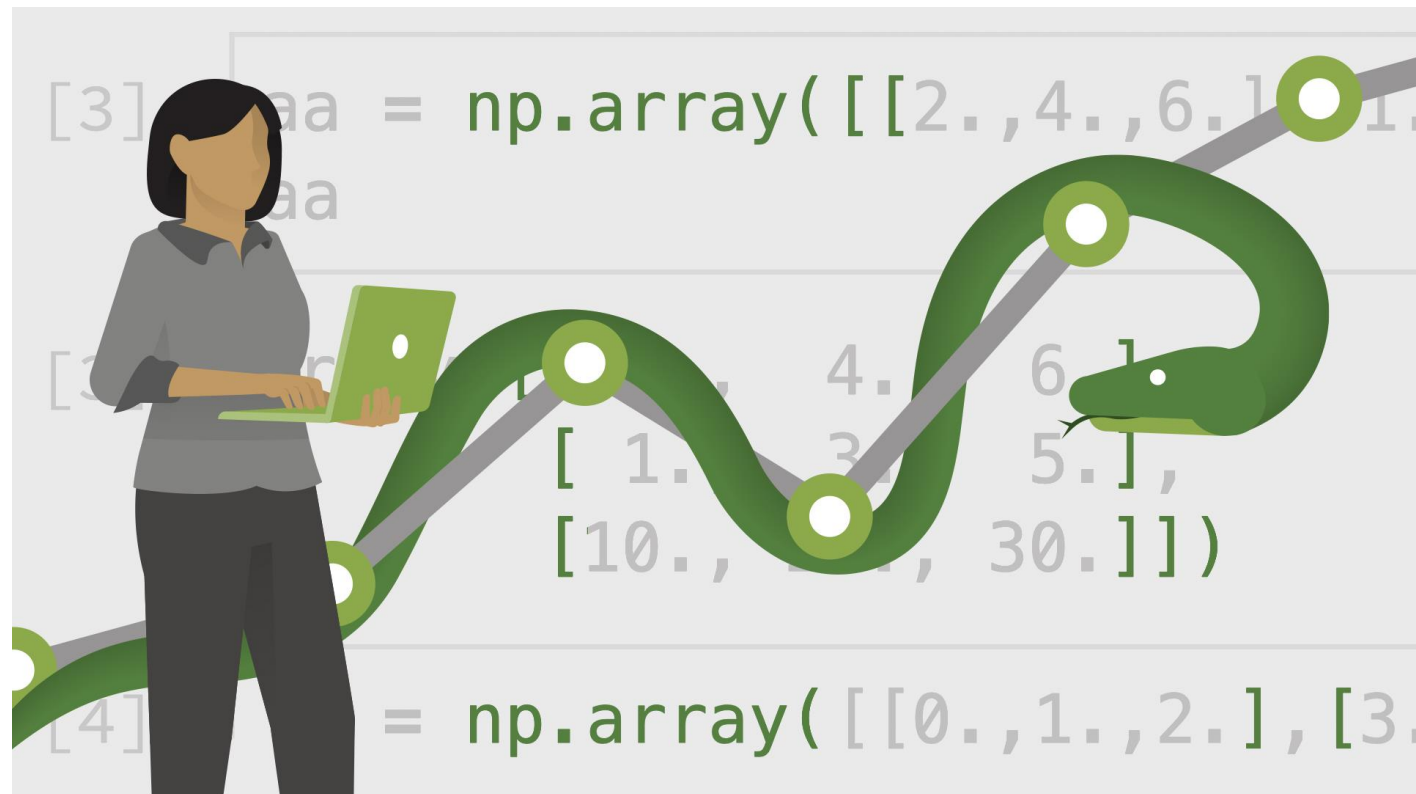
**У рядок не можна додати новий символ або видалити існуючий, не створивши при цьому нового рядка.**



# Які дії можна виконувати над списками в мові Python?

При виконанні операцій зі **списками** інші списки можуть не створюватися, а змінюється безпосередньо оригінал.

**Зі списків можна видаляти елементи, додавати нові. Списки можуть копіюватися, коли результат операції присвоюється іншій змінній.**



# Які дії можна виконувати над списками в мові Python?

**Об'єднання (конкатенація) списків задається операндом «+».** Особливості застосування цієї операції подано на прикладах у таблиці:

## Фрагмент програми

```
1 a = [1, 2, 3]
2 b = [4, 5, 6]
3 print (a+b)
```

```
1 a = [1, 2, 3]
2 b = [4, 5, 6]
3 print (b+a)
```

## Результат

```
[1, 2, 3, 4, 5, 6]
```

```
[4, 5, 6, 1, 2, 3]
```

# Які дії можна виконувати над списками в мові Python?

(Продовження...) **Об'єднання (конкатенація) списків:**

**Фрагмент програми**

```
1 a = [1, 2, 3]
2 b = [4]
3 print (a+b)
```

```
1 a = [1, 2, 3]
2 b = 4
3 print (a+b)
```

**Результат**

```
[1, 2, 3, 4]
```

```
print (a+b)
TypeError: can only concatenate list
(not "int") to list
```

# Які дії можна виконувати над списками в мові Python?

**Дублювання** списку задають операндом `<<*>>` за правилами, які описано на прикладах таблиці:

## Фрагмент програми

```
1 a = [1, 2, 3]
2 b = 3
3 print (a*b)
```

```
1 a = [1, 2, 3]
2 b = -3
3 print (a*b)
```

## Результат

```
[1, 2, 3, 1, 2, 3, 1, 2, 3]
```

```
[]
```

# Які дії можна виконувати над списками в мові Python?

(Продовження...) **Дублювання** списку:

**Фрагмент програми**

```
1 a = [1, 2, 3]
2 b = -1
3 print (a*b)
```

```
1 a = [1, 2, 3]
2 b = 2.5
3 print (a*b)
```

**Результат**

```
[]
```

```
print (a*b)
TypeError: can't multiply sequence by non-int of type 'float'
```

# Які дії можна виконувати над списками в мові Python?

Фрагмент списку має назву **зріз**. Щоб отримати потрібний фрагмент, вказують початковий і кінцевий індекс елемента за правилами, подібними до утворення зрізів у рядках:

## Фрагмент програми

```
1 a = [1, 2, 3, 4, 5, 6]
2 b = a[1:3]
3 print (b)
```

```
1 a = [1, 2, 3, 4, 5, 6]
2 b = a[:3]
3 print (b)
```

## Результат

[2, 3]

[1, 2, 3]

# Які дії можна виконувати над списками в мові Python?

(Продовження...) **Зріз** списку:

## Фрагмент програми

```
1 a = [1, 2, 3, 4, 5, 6]
2 b = a[3:]
3 print (b)
```

```
1 a = [1, 2, 3, 4, 5, 6]
2 b = a[3:10]
3 print (b)
```

## Результат

[4, 5, 6]

[4, 5, 6]

# Які дії можна виконувати над списками в мові Python?

Для визначення кількості елементів списку використовують функцію **len()**.

Наприклад, при виконанні фрагмента коду:





```
a=[2, 4, 6, 8, 10]  
print (len(a))
```

Отримаємо значення

5

# Які дії можна виконувати над списками в мові Python?

Елементи списку можна **змінювати**. Для цього слід звернутись до індексу списку та за адресою цього індекса надати нове значення:

List = [  <sup>0</sup>  <sup>1</sup>  <sup>2</sup>  <sup>3</sup> ]

```
1 a = ['a', 'b', 'c', 'd']
2 a[2] = 'f'
3 print (a)
```

**Результат:**

**['a', 'b', 'f', 'd']**

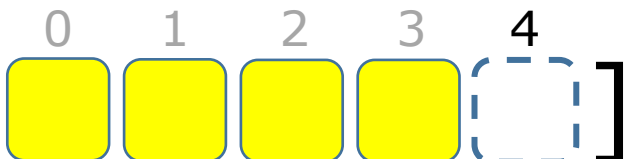
# Які дії можна виконувати над списками в мові Python?

Щоб **додати новий** елемент у список, використовують метод **append()**:

Список

`list.append(x)`

потрібне значення

List = [  ]

```
1 a = ['a', 'b', 'c', 'd']
2 a.append('e')
3 print (a)
```

**Результат:**

`['a', 'b', 'c', 'd', 'e']`

# Які дії можна виконувати над списками в мові Python?

Щоб додати в список елемент на вказану позицію, використовують метод **insert**:

Список

`list.insert(i, x)`

потрібне значення

List = [ 0123  
0 1 2 3  
[  12 3  
[  1 2 3  
[  3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[  1 2 3  
[





# Які дії можна виконувати над списками в мові Python?

Для **видалення** елемента зі списку використовують оператор **del**:

`del list[i]`

*індекс (позиція) елемента у списку*

*Список*

List = [  <sup>0</sup>  <sup>1</sup>  <sup>2</sup>  <sup>3</sup> ]

```
1 a = ['a', 'b', 'c', 'd']
2 del a[2]
3 print (a)
```

*Результат:*

`['a', 'b', 'd']`

# Які дії можна виконувати над списками в мові Python?



*Видаляти можна не тільки окремі елементи, а й цілі діапазони, вказуючи зрізи списку.*

Для **видалення конкретного значення** зі списку використовують метод:

Список

`list.remove(x)`

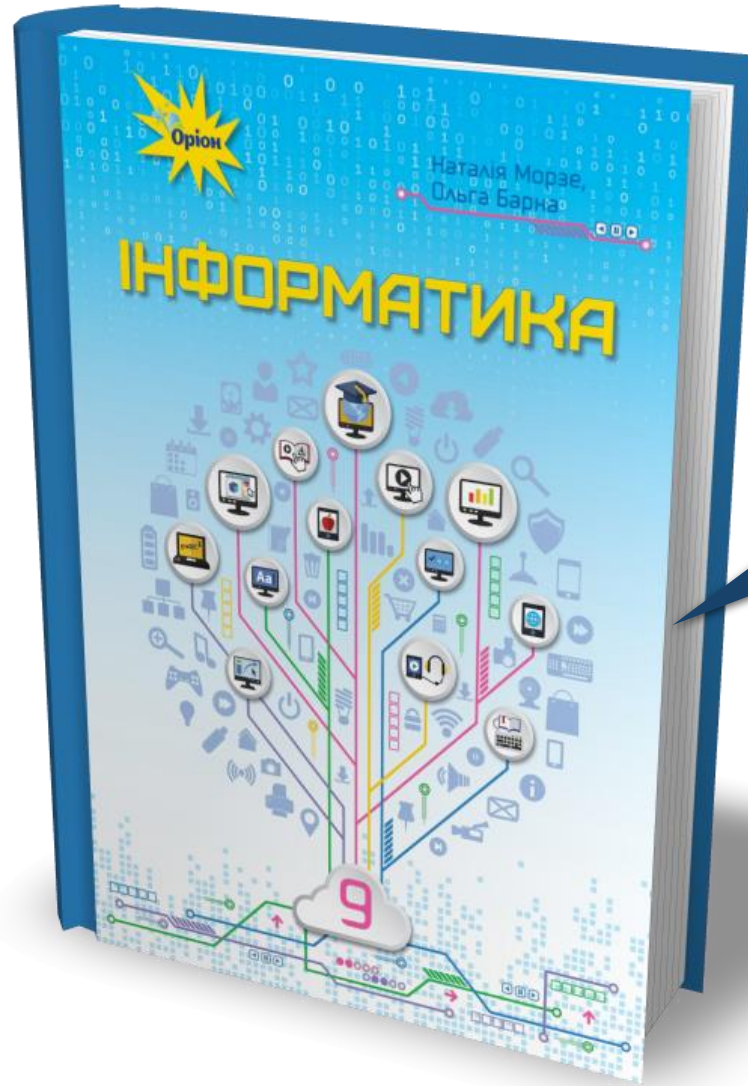
значення, яке видаляють

```
1 a = ['a', 'b', 'c', 'd']
2 a.remove('b')
3 print (a)
```

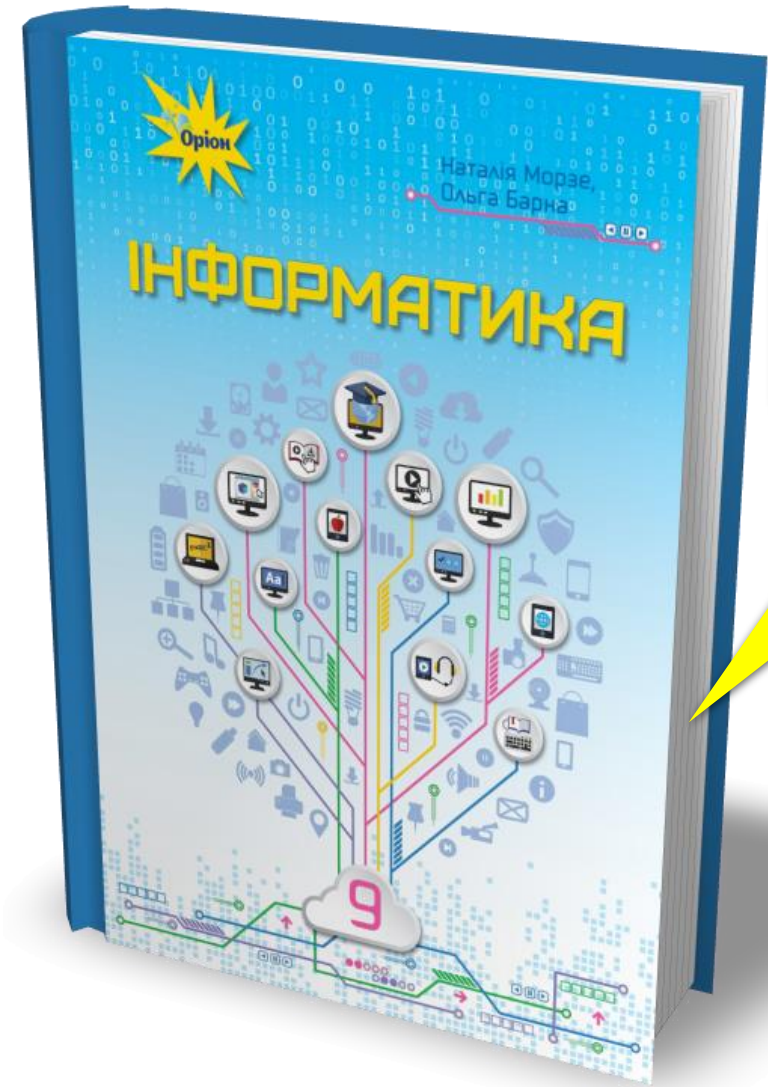
**Результат:**

`['a', 'c', 'd']`

# Домашнє завдання



**Проаналізувати**  
**§ 11, с. 128-132**



**Сторінка  
137**



Інформатика 9

Урок 34

за підручником

Морзе Н.В. та ін.

**Дякую за увагу!**

За навчальною програмою 2017 року

