

Урок № 34

Словники



До підручника
«Інформатика. 9 клас»
авторів Олени Бондаренко, Василя Ластовецького,
Олександра Пилипчука, Євгена Шестопалова

СЛОВНИКИ

На уроці ми дізнаємося:

- Що таке словник у Python
- Як створювати словники
- Як додавати, змінювати та видаляти елементи
- Як використовувати методи словників
- Як перебирати елементи словника
- Як працює віджет **Combobox**



Створення словників

Поняття словника

Словник (об'єкт типу **dict**) — це неупорядкована сукупність пар **ключ:значення**.

Приклад:

Словник може описувати об'єкт «Людина»:

СПИСОК



Семещук	Антоніна	2024	велоспорт
---------	----------	------	-----------

СЛОВНИК



прізвище	ім'я	рік	захоплення
Семещук	Антоніна	2024	велоспорт



Кожне значення має власний ключ.



Чому для збереження інформації про людину або товар інколи незручно використовувати лише список?

Ключі та значення

Будова словника



Кожне значення в словнику має унікальний ключ.

Ключ — величина одного з незмінюваних типів; може бути рядком, логічним значенням, цілим або дійсним числом.

Приклад:

ключ → 'ім'я'

значення → 'Антоніна'



Які ключі можна створити для словника «шкільний щоденник»?



Створення словника

Фігурні дужки { }

Щоб створити словник, потрібно помістити у фігурні дужки { } пари **ключ:значення**, розділені комами.

Загальний запис:

```
dictionary = {ключ1:значення1, ключ2:значення2}
```

Приклад:

```
zakupy = {}  
magazyn = {'яблука':15, 'груші':25, 'огірки':9,  
           'помідори':15, 'морква':9}
```



Чому порожній словник створюють як {}, а не як []?

Доступ до значення

Ключ відкриває дані

Словник визначає взаємозв'язок пар **ключ:значення**.
Оскільки ключ не повторюється, за ним відбувається доступ до значення.

Приклад:

magazyn['яблука']

Результат: 15. За ключем 'яблука'
отримуємо відповідне значення.



Як отримати зі словника ціну товару 'груші'?



Помилка KeyError

Коли ключа немає

У програмуванні помилка `KeyError` означає, що ви намагаєтеся отримати доступ до елемента в словнику за допомогою ключа, якого не існує.

Приклад:

```
magazyn['цибуля']
```

Результат: `KeyError: 'цибуля'`

Пояснення: помилка виникає, якщо такого ключа в словнику немає.



Чому програмі важливо повідомляти про відсутність ключа?



Перевірка наявності ключа

Використання оператора in

Щоб попередити таку помилку, потрібно перевірити наявність ключа в словнику за допомогою умови **<ключ> in <словник>**.

Приклад:

```
if 'цибуля' in magazin:  
    print(magazin['цибуля'])  
else:  
    print('Товар відсутній')
```

Результат: Товар відсутній.



Чому перевірка **in** робить програму надійнішою?

Додавання та зміна елементів

Оновлення словника

Для додавання елемента в словник слід звернутися до елемента за його ключем і присвоїти йому значення.

Приклад:

magazyn['морква'] = 5

Пояснення: якщо ключ існує — значення змінюється.



Чому один і той самий запис може використовуватися й для додавання, й для зміни даних?



Видалення елементів

Команда **del**

Для видалення елемента за ключем застосовується команда **del**.

Приклад:

`del magazyn['яблука']`

Результат: елемент із ключем 'яблука' видаляється.



Чому видалення даних потребує точного зазначення ключа?



Методи словників

Методи `keys()` і `values()`

Методи допомагають працювати з ключами та значеннями.

Приклад:

```
books = {'І. Котляревський': 'Енеїда',  
         'М. Гоголь': 'Тарас Бульба',  
         'Т. Шевченко': 'Гайдамаки',  
         'О. Гончар': 'Собор'}
```

```
І. Котляревський – Енеїда  
М. Гоголь – Тарас Бульба  
Т. Шевченко – Гайдамаки  
О. Гончар – Собор  
>>>
```



Чому інколи потрібно отримати лише ключі або лише значення?

Метод `keys()`

Робота з ключами

Метод `keys()` повертає `dict_keys()` — придатне для перебору (ітеративне) подання ключів словника.

Приклад:

```
books.keys()
```

Результат:

```
dict_keys(['І. Котляревський', 'М. Гоголь',  
          'Т. Шевченко', 'О. Гончар'])
```

Список ключів: `list(books.keys())`



Для чого може знадобитися
список усіх авторів у словнику?

Метод values()

Робота зі значеннями

Метод **values()** повертає **dict_values()** — додатне для перебору (ітеративне) подання значень словника.

Приклад:

```
list(books.values())
```

Результат:

```
['Енеїда', 'Тарас Бульба', 'Гайдамаки', 'Собор']
```

Пояснення: отримуємо список усіх значень словника.

```
Енеїда  
Тарас Бульба  
Гайдамаки  
Собор  
>>>
```



Іноді корисно працювати не використовуючи ключі, лише зі значеннями. Чому?

Перебір значень словника

Простий цикл for та використання values() у циклі

Послідовно перебрати всі пари **ключ:значення** словника можна за допомогою циклу.

Приклад 1:

```
for key in books:  
    print(key, ' – ', books[key])
```

```
I. Котляревський – Енеїда  
М. Гоголь – Тарас Бульба  
Т. Шевченко – Гайдамаки  
О. Гончар – Собор  
>>>
```

Приклад 2:

```
for value in books.values():  
    print(value)
```

```
Енеїда  
Тарас Бульба  
Гайдамаки  
Собор  
>>>
```

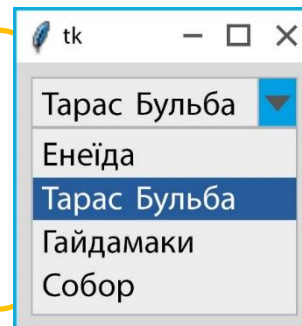
Об'єкти класу Combobox

Поле зі списком

Віджет **tkinter Combobox** (поле зі списком) — це розкривний список для вибору одного зі значень. **Combobox** описано в модулі **tkinter ttk**.

Приклад:

```
from tkinter import ttk
combo1 = ttk.Combobox(root, values = ['Енеїда',
                                     'Тарас Бульба', 'Гайдамаки', 'Собор'])
combo1.pack()
```



Де у повсякденних програмах ви вже бачили поля зі списком?

Методи `current()` і `get()`

Робота з вибором

- `current()` — встановлює або повертає індекс;
- `get()` — повертає значення.

Вибір елемента: `combo1.current(3)`

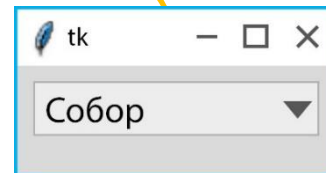
Отримання даних:

```
index = combo1.current()
```

```
item = combo1.get()
```

```
print('індекс вибраного елемента – ', index, ', значення – ', item)
```

Індекс вибраного елемента – 3. Значення – Собор.



Чому програмі важливо знати і номер вибраного елемента, і його значення?

Перевірте себе

Виконайте завдання

У випадяючому списку `combobox` (елемент **`ttk.Combobox`**) встановлено список значень **['Легковий', 'Вантажний']**. Програмно встановити значення **"Вантажний"** як вибране за промовчанням допоможе метод:

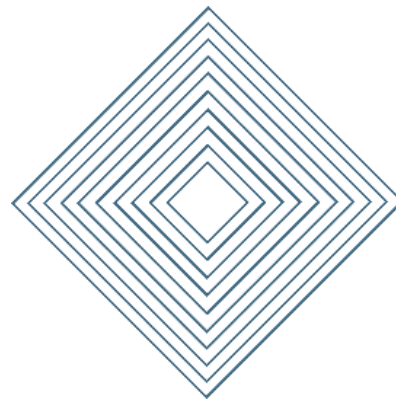
А `combobox.current(1)`

В `combobox.insert("Вантажний")`

Б `combobox.select(1)`

Г `combobox.config(text = "Вантажний")`

Вправа 34



Виконайте вправу 34
за підручником
або в робочому зошиті

Підсумки теми

Ключові поняття

1. Словник
2. Ключ
3. Значення
4. Методи **keys()** і **values()**
5. **Combobox**



Ми навчилися

1. Створювати словники
2. Додавати, змінювати та видаляти елементи
3. Перебирати словники
4. Використовувати **Combobox**

Домашнє завдання



1. Опрацювати § 34.
2. Дати відповіді на питання для самоперевірки 1–6.
3. Пройти тест за QR-кодом або посиланням rnk.com.ua/112954.

