

Лабораторна робота №10

Словники як засіб для опрацювання структурованих даних.

Використання можливостей модуля csv для зберігання структурованих даних та пошук інформації у файлах.

***Мета роботи:** оволодіння навичками обробки та зберігання структурованих даних за допомогою словників та csv-файлів.*

Короткі теоретичні відомості

В *Python* немає структури даних схожої за функціоналом до записів *Pascal* чи структур *C++*. Ця технологія підтримує об'єктно-орієнтовану парадигму, відповідно, для реалізації комбінованого типу в контексті цієї задачі можна було б скористатися *класом*. З іншого боку, серед базових типів *Python* є *списки*, *кортежі* та *словники*, які цілком можна було б «приладнати» до зберігання набору даних різного типу.

Кортежем, зокрема, можна скористатися, якщо пронумерувати поля запису і звертатися до їхніх даних за номером поля. Такий підхід не завжди зручний, деколи краще використовувати *асоціативні масиви*, – в них кожен елемент доступний не за його номером, а за його назвою. Функціонал асоціативного масиву у *Python* надає тип даних *dict* (*словник*).

Словник утворюється зі списку елементів, кожен з яких утворений з пари елементів: *ключ*, тобто назва даних, та *значення*, тобто самі дані. Щоправда з використанням словників в також треба бути обережним. Словник, на відміну від кортежу, дозволяє додавати нові елементи в уже заповнений асоціативний масив. Тобто в коді програми потрібно слідкувати, щоб до даних про одну книжку в екземплярі словника не дописувалися ще якісь властивості, наприклад, дані наступної книжки зі списку. Але в нашому прикладі ми зупинимося саме на використанні словників.

Оголошення та створення словника в *Python* доволі просте:

```
<змінна> = {'<ключ 1>': <значення 1>, '<ключ 2>':  
            <значення 2>, ..., '<ключ n>': <значення 2>}
```

Для кращого контролю за полями запису винесемо код вводу даних про книжку в окрему функцію і вводитимемо дані лише через цю функцію.

Можливість змінювати словники є потенційним місцем проблем для запису-читання даних в двійковому форматі. Насправді, дані комбінованих типів не завжди зберігають в такому вигляді. До прикладу, в програмах на *C#* є можливість використовувати різні серіалізатори, наприклад, замість бінарного серіалізатора скористатися класом який серіалізує дані структур в текстовий формат xml (*eXtensible Markup Language*), що часто використовується для передачі та зберігання структурованих даних в мережі *Internet*.

Тут ми скористаємося з можливостей модуля csv, який дозволяє Python-програмам зберігати структуровані дані в текстових файлах у спеціальному форматі csv (*Comma-Separated Values*). Такий формат зручний своєю універсальністю, до прикладу, один з діалектів csv читається процесорами електронних таблиць MS Excel та OOo Calc і його можна перетворити на електронну таблицю.

Завдання 1

Скласти програму для опрацювання структурованих даних згідно варіанта. Програма повинна читати дані відповідної структури з консолі та записувати у вигляді словника з даними у csv-файл. Передбачити дозапис даних у файл так, щоб внаслідок виконання програми файл містив 10-15 екземплярів словника. Після завершення вводу даних вивести увесь вміст файлу на консоль у вигляді таблиці.

1. Створити програму для опрацювання файлу з даними про авіарейси. Структура даних: номер рейсу, пункт призначення, час відправлення, ціна квитка, кількість вільних місць. В програмі передбачити ввід даних користувачем та збереження їх у файл, а також вивід усіх даних з записаного раніше файлу.

2. Написати програму для опрацювання файлу з інформацією про складання екзаменаційної сесії студентами. Структура даних: шифр групи, прізвище студента, назва предмета, оцінка, назва предмета... (всього 5 предметів). В програмі передбачити ввід даних користувачем та збереження їх у файл, а також вивід на екран усіх даних з записаного раніше файлу.

3. У файл записують дані про продаж комп'ютерної техніки наступної структури: код покупця, назва товару, ціна за одиницю товару, продана кількість. Написати програму яка дозволить користувачеві створювати файл з даними, вносити в нього дані, а також виводити раніше збережені у файлі дані на екран (у вигляді таблиці).

4. Створити програму для роботи з файлом, що містить дані про рух пасажирських потягів від деякої станції. Запис містить поля: назва кінцевої станції, номер потяга, час відправлення, кількість наявних місць по категоріях вагонів – СВ, купейний, плацкартний, загальний. В програмі передбачити ввід даних користувачем та збереження їх у файл, а також вивід на екран усіх даних з записаного раніше файлу.

5. У файл записують дані про продаж турів клієнтам туристичної агенції наступної структури: прізвище та ім'я клієнта, назва туру та його тривалість, вартість туру та відмітка про оплату. Написати програму яка дозволить користувачеві створювати файл з даними, вносити в нього дані, а також виводити раніше збережені у файлі дані на екран (у вигляді таблиці).

6. Створити програму для опрацювання файлу з даними про оплату комунальних послуг через платіжну систему банку. Структура даних:

прізвище платника, адреса платника, вид послуги, дата оплати, сума. В програмі передбачити ввід даних користувачем та збереження їх у файл, а також вивід усіх даних з записаного раніше файлу.

7. Створити програму для опрацювання файлу з інформацією про запис на прийом протягом поточного тижня відвідувачів поліклініки. Структура даних: прізвище відвідувача, домашня адреса, рік народження, прізвище сімейного лікаря, день тижня, потрібна спеціалізація лікаря. В програмі передбачити ввід даних користувачем та збереження їх у файл, а також вивід у формі таблиці усіх даних з записаного раніше файлу.

8. У файл записують інформацію про вакансії, подані ІТ-компаніями, яка містить наступні дані: назва компанії, назва позиції, мова/технологія програмування, вимоги до претендента, орієнтовна заробітна плата. Розробити програму для створення файлу з даними, внесення в нього даних, а також виводу усіх даних з файла на екран (у вигляді таблиці).

9. Ріелтор зберігає у файлі інформацію з оголошень про продаж нерухомості у вигляді записів такої структури: адреса помешкання, кількість кімнат, житлова площа, загальна площа, поверх, ціна, телефон власника. Реалізувати програму, яка дозволить ріелторові створювати новий файл з даними, вносити в нього дані, а також виводити раніше збережені у файлі дані на екран.

10. Створити програму для роботи з файлом, що містить дані про продаж вживаних автомобілів. Запис містить поля: марка авто, модель авто, тип кузова, об'єм двигуна, рік випуску, ціна. В програмі передбачити ввід даних користувачем та збереження їх у файл, а також вивід на екран усіх даних з записаного раніше файлу.

11. Рекрутер записує у файлі інформацію щодо резюме, розміщених ІТ-фахівцями у сервісах пошуку роботи. Записи містять інформацію такої структури: прізвище та ім'я, електронна адреса, місто проживання, перелік технологій (текст, в якому перераховані технології відділяються комами), досвід роботи, бажання працювати віддалено, можливість зміни міста проживання. Реалізувати програму для створення файлу з даними, внесення в нього даних, а також виводу усієї інформації з файла на екран.

12. Менеджер паркінгу реєструє у файлі інформацію про парковку авто та оплату послуг їх власниками. Записи містять поля: марка та модель авто, державний номерний знак, телефон власника, година парковки (ціле число) та кількість годин, за яку вноситься передоплата. Скласти програму для вводу даних та збереження їх у файл, а також для виводу на екран усіх даних із записаного раніше файлу.

13. Створити програму для опрацювання файлу з даними про міжміські автобусні рейси. Запис містить поля: назва міста призначення, час відправлення, вартість квитка, загальна кількість місць в автобусі, кількість проданих квитків. В програмі передбачити ввід даних користувачем та збереження їх у файл, а також вивід усіх даних з записаного раніше файлу.

14. Менеджер онлайн-магазину зберігає у файлі дані про замовлення, що надходять. Структура даних: повна назва товару, ціна, кількість, прізвище та ім'я замовника, населений пункт для доставки, спосіб доставки, здійснення оплати. Написати програму для автоматизації роботи менеджера. В програмі передбачити ввід даних менеджером та збереження їх у файлі, а також вивід усіх даних з записаного раніше файлу.

15. Дані про доступні для перегляду на онлайн-сервісі фільми зберігаються у файлі у вигляді записів такої структури: назва фільму, жанр, рік виходу на екрани, назва студії, прізвище режисера, загальна тривалість та кількість серій. Розробити програму для створення файлів з даними про фільми, запису даних у файл та для виводу збережених у файлі даних на екран.

16. Створити програму для опрацювання файлу з даними про наявність книг у книжковому магазині. Запис містить поля: назва видавництва, назва книги, прізвище автора, рік видання, роздрібна ціна, кількість примірників на складі. В програмі передбачити ввід даних користувачем та збереження їх у файл, а також вивід усіх даних з записаного раніше файлу.

Приклад виконання завдання 1.

Варіант № 16. Умова задачі: Скласти програму для опрацювання структурованих даних згідно варіанта. Програма повинна читати дані відповідної структури з консолі та записувати у вигляді словника з даними у csv-файл. Передбачити дозапис даних у файл так, щоб внаслідок виконання програми файл містив 10-15 екземплярів словника. Після завершення вводу даних вивести увесь вміст файлу на консоль у вигляді таблиці.

Створити програму для опрацювання файлу з даними про наявність книг у книжковому магазині. Запис містить поля: назва видавництва, назва книги, прізвище автора, рік видання, роздрібна ціна, кількість примірників на складі. В програмі передбачити ввід даних користувачем та збереження їх у файл, а також вивід усіх даних з записаного раніше файлу.

Програма повинна імітувати роботу простої інформаційної системи та надавати користувачеві такий функціонал:

- *зберігати інформацію про перелік книг у магазині у формі даних вказаної структури;*
- *створювати новий файл (базу даних) та заповнювати його даними про книги;*
- *записувати інформацію про окрему книгу в існуючий, створений раніше файл, з даними;*
- *виводити на екран перелік книг, що зберігаються у файлі (базі даних) та дані про них.*

Для коректної роботи програми слід описати відповідний комбінований тип даних (**запис** про окрему книгу), що поєднуватиме в собі різні базові типи

даних під спільною назвою та дозволить оперувати цим набором даних як одним цілим.

Не будемо обтяжувати програмний код не надто важливим функціоналом для вибору файлу з даними, а реалізуємо роботу програми з одним і тим же файлом, ім'я якого запишемо безпосередньо в коді програми. Для зручного структурування коду організуємо програму у вигляді набору підпрограм, кожна з яких вирішує окреме завдання: створення файлу і внесення даних, вивід усього переліку книг, чи додавання даних про нову книгу в існуючий перелік. В головній програмі, при такому підході, можна реалізувати «вічний» цикл в ході якого користувач обиратиме потрібну йому дію та переглядатиме її результат.

Код програми:

```
import csv

db_name = 'books_shop.csv'

# функція для вводу даних про окрему книжку
# формує і повертає словник з даними
def input_book():
    book_title = input('Назва книги: ')
    authors = input('Автори: ')
    publisher = input('Видавництво: ')
    publish_year = int(input('Рік видання: '))
    price = float(input('Ціна: '))
    count = int(input('Кількість: '))
    book = {'book_title' : book_title, 'authors': authors,
            'publisher' : publisher, 'publish_year':
            publish_year, 'price' : price, 'count' : count}
    return book

# функція для створення\перестворення бази
# ввід даних про книжки і запис їх у файл
def write_new_file(filename):
    books = [] # список книг для запису у файл
    again = 1
    with open(filename, 'w', newline='',
```

Лабораторний практикум з програмування на Python

```
        encoding='cp1251') as file:
    columns = ['book_title', 'authors', 'publisher',
               'publish_year', 'price', 'count']
    writer = csv.DictWriter(file, fieldnames=columns)
    writer.writeheader()
    while again > 0:
        books.append(input_book())
        print('Додати ще одну книгу?')
        print('1 - так, додати ще;')
        print('0 - зберегти дані і вийти')
        again = int(input('>>>'))
    writer.writerows(books)
    print('нову базу книг створено')

# зчитування даних про книги з csv-файла
# та вивід їх на консоль
def read_all(filename):
    with open(filename, 'r', newline='',
              encoding='cp1251') as file:
        reader = csv.DictReader(file)
        for row in reader:
            print(row['authors'], ' ', row['book_title'],
                  ', - ', row['publisher'], ': ',
                  row['publish_year'], '. ', row['price'],
                  ' грн., ', row['count'], ' в наявності.')

# дозапис даних про нову книгу
def write_book(filename, data):
    with open(filename, 'a', newline='',
              encoding='cp1251') as file:
```

```

columns = ['book_title', 'authors', 'publisher',
           'publish_year', 'price', 'count']
writer = csv.DictWriter(file, fieldnames=columns)
writer.writerow(data)

# Основна програма
choise = 4 # ненульове значення для входу в цикл
# за умовою в меню 0 - вихід з циклу, і цикл не
# виконається жодного разу, якщо ініціалізувати цю
# змінну значенням 0, а оголошення та
# ініціалізація змінної choise потрібні до циклу
while choise != 0:
    # вивід меню програми
    print('Оберіть потрібну дію:')
    print('1 - створити/перестворити базу книг.')
    print('2 - дописати нову книгу в базу')
    print('3 - переглянути всі книги в базі')
    print('0 - завершити роботу програми')
    # зчитування пункту меню обраного користувачем
    # та виконання відповідної дій з даними
    choise = int(input('>>>'))
    if choise == 1:
        write_new_file(db_name)
    elif choise == 2:
        write_book(db_name, input_book())
    elif choise == 3:
        read_all(db_name)

```

Завдання 2.

Скласти програму, яка опрацьовує csv-файл, утворений програмою з завдання 1. Програма повинна виводити у вигляді таблиці увесь вміст файла, далі дати користувачеві змогу вибрати критерій пошуку даних відповідно до завдань варіанта.

1. Розробити програму для пошуку даних у файлі, що містить інформацію про авіарейси. Структура даних: номер рейсу, пункт призначення, час відправлення, ціна квитка, кількість вільних місць. В програмі передбачити вивід у вигляді таблиці інформації про всі рейси до заданого користувачем пункту призначення та усі рейси, що відправляються у заданий користувачем проміжок часу.

2. Створити програму для пошуку даних у файлі, що містить дані про складання екзаменаційної сесії студентами. Структура даних: шифр групи, прізвище студента, назва предмета, оцінка, назва предмета... (всього 5 предметів). В програмі передбачити вивід списку студентів, які отримали “5” з усіх предметів, а також рейтингу (середньої оцінки з усіх предметів) в упорядкованому за спаданням вигляді.

3. Файл містить дані про продаж комп’ютерної техніки наступної структури: код покупця, назва товару, ціна за одиницю товару, продана кількість. Розробити програму, яка дозволить користувачеві за введенням з клавіатури кодом покупця виводити список куплених ним товарів та загальну вартість покупок, а за введеною назвою товару, – коди покупців, що його придбали та кількість проданих одиниць.

4. Створити програму для пошуку даних у файлі з інформацією про рух пасажирських потягів від деякої станції. Структура даних: назва кінцевої станції, номер потяга, час відправлення, кількість наявних місць по категоріях вагонів – СВ, купейний, плацкартний, загальний. Програма повинна виводити список потягів за вказаною назвою кінцевої станції та проміжком часу відправлення, а також відбирати лише ті з них, які мають вільні місця вказаної категорії.

5. Файл містить інформацію про продаж турів клієнтам туристичної агенції. Структура даних: прізвище та ім’я клієнта, назва туру та його тривалість, вартість туру та відмітка про оплату. Розробити програму яка дозволить користувачеві за введеною з клавіатури назвою туру знайти усіх клієнтів, що придбали цей тур та порахувати їх кількість та загальну суму виручки від продажу турів, а також вивести список усіх клієнтів, що замовили вказаний тур, але не внесли кошти на його оплату.

6. Створити програму для пошуку даних у файлі, що містить інформацію про оплату комунальних послуг у банку наступної структури: прізвище платника, адреса платника, вид послуги, дата оплати, сума. В програмі

передбачити вивід даних про всі оплати послуг за вказаною адресою а також вивід підсумкової таблиці з сумарною оплатою кожного виду послуг.

7. Створити програму для пошуку даних у файлі, що містить дані про запис на прийом, протягом поточного тижня, відвідувачів поліклініки. Структура даних: прізвище відвідувача, домашня адреса, рік народження, прізвище сімейного лікаря, день тижня, потрібна спеціалізація лікаря. В програмі передбачити вивід списку відвідувачів та підрахунок їх кількості для кожної спеціалізації на вказаний день тижня, а також вивід списку відвідувачів та їх кількості на кожен день тижня для заданої спеціалізації лікаря.

8. Файл містить інформацію про вакансії, подані ІТ-компаніями, яка містить наступні дані: назва компанії, назва позиції, мова/технологія програмування, вимоги до претендента, орієнтовна заробітна плата. Розробити програму, яка дозволить користувачеві відбирати за файла вакансії за введеною з клавіатури назвою мови програмування та діапазоном для розміру орієнтовної зарплатні, а також переглядати список усіх вакансій, поданих вказаною компанією.

9. Файл містить дані про продаж нерухомості у вигляді записів такої структури: адреса помешкання, кількість кімнат, житлова площа, загальна площа, поверх, ціна, телефон власника. Скласти програму яка дозволить користувачеві шукати у файлі та виводити на екран інформацію за такими параметрами: за введеною з клавіатури кількістю кімнат та діапазоном цін, або за кількістю кімнат та орієнтовною загальною чи житловою площею помешкання.

10. Скласти програму для пошуку даних у файлі з інформацією про продаж вживаних автомобілів. Запис містить поля: марка авто, модель авто, тип кузова, об'єм двигуна, рік випуску, ціна. Програма повинна виводити список авто за такими вказаними користувачем параметрами: марка, модель та діапазон цін або марка, модель та період випуску (у роках, від... до...).

11. Рекрутер зберігає у файлі інформацію щодо резюме, розміщених ІТ-фахівцями у сервісах пошуку роботи. Записи містять інформацію такої структури: прізвище та ім'я, електронна адреса, місто проживання, перелік технологій (текст, в якому перераховані технології відділені комами), досвід роботи, бажання працювати віддалено, можливість зміни міста проживання. Розробити програму яка дозволить йому відбирати для офісу із заданого міста резюме, що відповідають вказаній технології, розділивши їх на три частини: претенденти, що проживають у заданому місті, претенденти, готові до віддаленої роботи та претенденти, що погодяться переїхати у вказане місто.

12. Менеджер паркінгу реєструє у файлі інформацію про парковку авто та оплату послуг їх власниками. Записи містять поля: марка та модель авто, державний номерний знак, телефон власника, година парковки (ціле число) та кількість годин, за яку вноситься передоплата. Скласти програму для пошуку даних у файлі, яка дозволить менеджеру, вказавши поточний час (годину) дізнатися список авто, власникам яких доведеться доплатити за парковку та за

скільки годин, а також дізнатися, чи авто може без доплати покинути паркінг, вказавши його номер.

13. Створити програму для пошуку даних у файлі, що містить інформацію про міжміські автобусні рейси. Запис містить поля: назва міста призначення, час відправлення, вартість квитка, загальна кількість місць в автобусі, кількість проданих квитків. Запрограмувати визначення заповненості (у %) рейсів до вказаного міста чи обчислення сумарної виручки для кожного рейсу (за вибором користувача програми) та вивід результатів у вигляді таблиці.

14. Менеджер онлайн-магазину опрацьовує замовлення, дані про які зберігаються у файлі. Структура даних: повна назва товару, ціна, кількість, прізвище та ім'я замовника, населений пункт для доставки, спосіб доставки, здійснення оплати. Запрограмувати вивід з файлу усіх оплачених замовлень, які слід доставити у зазначене місто, та вивід усіх неоплачених замовлень за заданим способом доставки товару.

15. Дані про доступні для перегляду на онлайн-сервісі фільми зберігаються у файлі у вигляді записів такої структури: назва фільму, жанр, рік виходу на екрани, назва студії (кінокомпанії), прізвище режисера, загальна тривалість та кількість серій. Скласти програму для пошуку фільмів у файлі. Програма повинна за вказаним жанром виводити на екран два списки окремо одно-, а окремо багатосерійні фільми цього жанру, а також за вказаною назвою кінокомпанії та діапазоном років виводити інформацію про фільми, зняті компанією у цей період.

16. Створити програму для пошуку книг у книжковій крамниці, дані якої зберігаються у файлі, що містить записи такої структури: назва видавництва, назва книги, прізвище автора, рік видання, роздрібна ціна, кількість примірників на складі. Запрограмувати пошук книг за прізвищем автора чи назвою (або її частиною), а також вивід списку усіх книг, які вийшли у вказаному видавництві.

Приклад виконання завдання 2.

Варіант № 16. Умова задачі: Скласти програму, яка опрацьовує csv-файл, утворений програмою з завдання 1. Програма повинна виводити у вигляді таблиці увесь вміст файлу, далі дати користувачеві змогу вибрати критерій пошуку даних відповідно до завдань варіанта.

Створити програму для пошуку книг у книжковій крамниці, дані якої зберігаються у файлі, що містить записи такої структури: назва видавництва, назва книги, прізвище автора, рік видання, роздрібна ціна, кількість примірників на складі. Запрограмувати пошук книг за прізвищем автора чи назвою (або її частиною), а також вивід списку усіх книг, які вийшли у вказаному видавництві.

Як і у попередньому завданні, реалізуємо програму у вигляді окремих функцій для кожного пункту завдання. А їх виконання в основній програмі будемо запускати за вибором користувача з допомогою текстового меню.

Код програми для розв'язування задачі на мові *Python*:

```
import csv

db_name = 'books_shop.csv'

# функція для пошуку записів про книги
# за вказаним прізвищем одного з авторів
def find_by_author(filename):
    author = input('Введіть прізвище автора\n>> ')
    with open(filename, 'r', newline='',
              encoding='cp1251') as file:
        reader = csv.DictReader(file)
        for row in reader:
            if author in row['authors']:
                print(row['authors'], ' ', row['book_title'], ', - ',
                      row['publisher'], ': ', row['publish_year'], '. ',
                      row['price'], ' грн., ' , row['count'],
                      ' в наявності.')

# функція для пошуку записів про книги
# за назвою, або її частиною
def find_by_title(filename):
    book = input('Введіть назву книги\n>> ')
    with open(filename, 'r', newline='',
              encoding='cp1251') as file:
        reader = csv.DictReader(file)
        for row in reader:
            if book in row['book_title']:
                print(row['authors'], ' ', row['book_title'], ', - ',
```

Лабораторний практикум з програмування на Python

```
        row['publisher'], ': ', row['publish_year'], '. ',
        row['price'], ' грн., ' , row['count'],
        ' в наявності.')
```

функція для пошуку записів про книги

за вказаним видавництвом

```
def find_by_publisher(filename):
    publisher = input('Введіть видавництво\n>> ')
    with open(filename, 'r', newline='',
              encoding='cp1251') as file:
        reader = csv.DictReader(file)
        for row in reader:
            if publisher == row['publisher']:
                print(row['authors'], ' ', row['book_title'], ', - ',
                      row['publisher'], ': ', row['publish_year'], '. ',
                      row['price'], ' грн., ' , row['count'],
                      ' в наявності.')
```

Основна програма

ненульове значення для входу в цикл

```
choise = 4
```

за умовою в меню 0 - вихід з циклу, якщо

ініціалізувати choise нулем, цикл не виконається

жодного разу, а оголошення та ініціалізація

змінної choise потрібні до початку циклу

```
while choise != 0:
    print('Оберіть потрібну дію:')
    print('1 - пошук книг за автором')
    print('2 - пошук книг за назвою')
    print('3 - пошук книг за видавництвом')
    print('0 - завершити роботу програми')
```

```
choise = int(input('>>> '))  
if choise == 1:  
    find_by_author(db_name)  
elif choise == 2:  
    find_by_title(db_name)  
elif choise == 3:  
    find_by_publisher(db_name)
```

Запитання для самоконтролю

1. Які засоби передбачені в ***Python*** для зберігання у файлах структурованих даних?
2. Які колекції можна використовувати для зберігання даних у csv-файлах?
3. Як у csv-файлах зберігаються дані зі словників?
4. Як структуровані дані подаються у форматі csv (***Comma-Separated Values***)?
5. Які функції ***Python*** використовуються для запису та читання файлів?