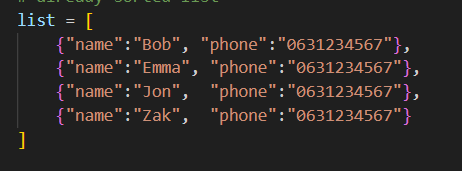
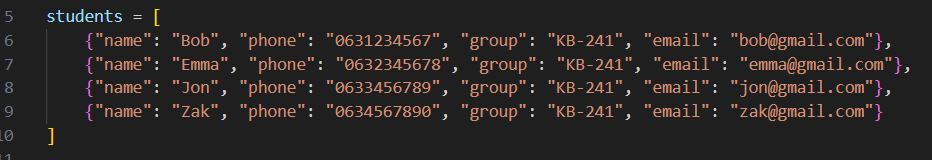
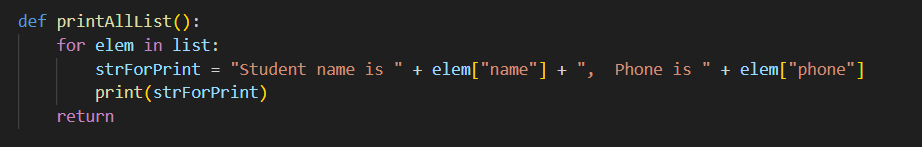
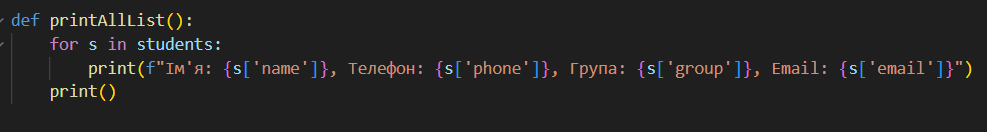
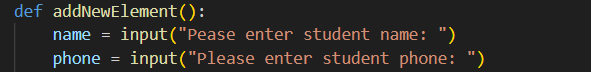
**Лабораторна робота №1  
 Списки. Словники. Кортежі**

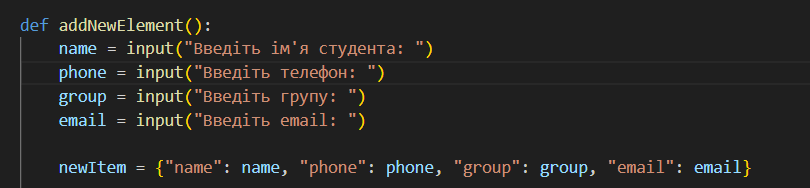
**Виконав: Шатун Іван, КБ-241**  
**Мета роботи**: Використовуючи теоретичне підґрунтя про складні структури даних Списки, Словники Кортежі, та використовуючи існуючий код, доробити програму додавши функціонал, що буде вказано в завданні до лабораторної роботи.  
  
 **Теоретичні відомості**

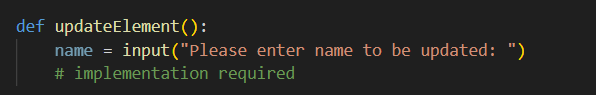
**Списки** Масив – набір фіксованої кількості елементів, що розміщені в пам’яті комп’ютера безпосередньо один за одним, а доступ до них здійснюється за індексом (номер даного елементу в масиві).  
 В Python для реалізації масиву використовуються списки. Список – тип даних, що представляє собою послідовність певних значень, що можуть повторюватись. Але на відміну від масиву – кількість елементів у списку може бути довільною.  
 Списки – структура даних, що може містити елементи різних типів, що перераховані через кому та заключені в квадратні дужки.  
 Списки служать для того, щоб зберігати об’єкти в певному порядку, особливо якщо порядок або вміст можуть змінюватися. Можна змінювати список, додати в нього нові елементи, а також видалити або перезаписати існуючі. Можна змінити кількість елементів у списку, а також самі елементи. Одне і те ж значення може зустрічатися в списку кілька разів.

**Словники** Словник дуже схожий на список, але порядок елементів в ньому не має значення, і вони вибираються не за допомогою зміщення. Замість цього для кожного значення вказується пов’язаний з ним унікальний ключ. Таким ключем може бути об’єкт одного з незмінних типів: рядок, булева змінна, ціле число, число з плаваючою точкою, кортеж і іншими об’єктами. Елементи словника можуть містити об’єкти довільного типу даних і мати необмежений рівень вкладеності. Елементи в словниках розташовуються в довільному порядку.  
 Словники можна змінювати – це означає, що можна додати, видалити і змінити їх елементи, які мають вигляд "ключ – значення"  
 Щоб створити словник, необхідно заключити в фігурні дужки ({}) розділені комами пари ключ: значення.

**Кортежі** Кортежі, як і списки, є послідовностями довільних елементів. На відміну від списків кортежі незмінні.  
 Всі операції над списками, що не змінюють список (додавання, множення на число, функції index() і count() і деякі інші операції) можна застосовувати до кортежів. Можна також по-різному змінювати елементи місцями і так далі. Щоб створити порожній кортеж використовується оператор ().  
 **Хід роботи**  
**1.** Ознайомився з теоретичним матеріалом лабораторної.  
**2.** У файлі lab\_01.py був частково реалізований телефонний довідник студентів. Він містив: нескінченний цикл запитів користувача (create, update, delete, print, exit), готові функції: addNewElement() — додавання нового студента (поля name і phone), deleteElement() — видалення існуючого студента, printAllList() — виведення списку, порожню функцію updateElement(), яку потрібно було реалізувати самостійно.  
Кожен запис про студента мав лише два поля:  
{"name": "Jon", "phone": "0631234567"}  
Список студентів був відсортований за полем name.  
**3.** В файлі я змінив дані про студента з двох до чотирьох полів, додав номер групи та електронну пошту, а також перейменовано список, для того щоб не перекривати вбудований тип Python list.  
Було:  
   
Стало:  
  
**4.** Оновлено функцію printAllList(), для того щоб виводились всі чотири поля через f-рядок:  
Було:  


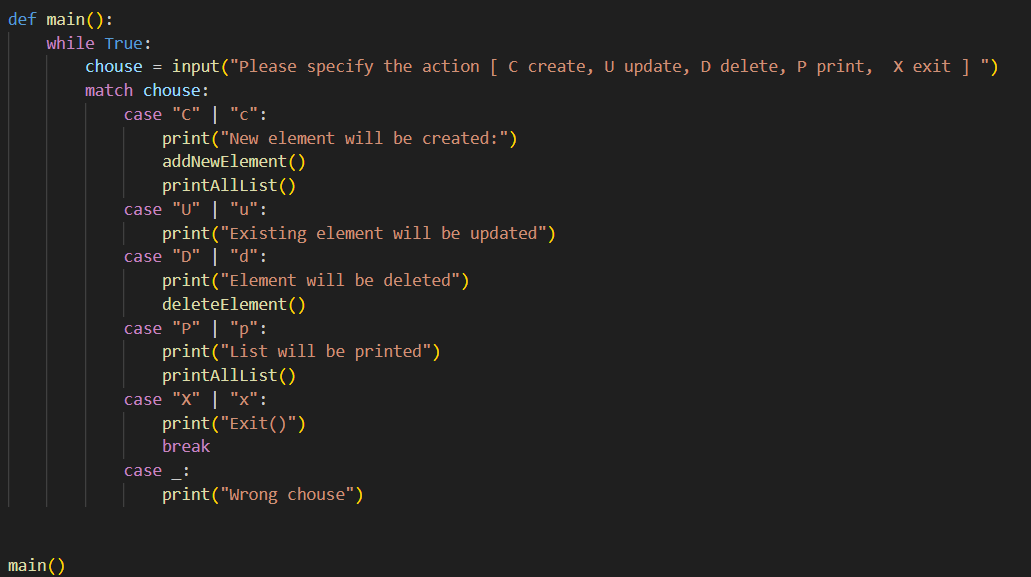
Стало:  
  
**5.** Оновлено функцію addNewElement(), додав введення нових полів. Алгоритм вставки для збереження сортування залишив без змін.  
Було:  


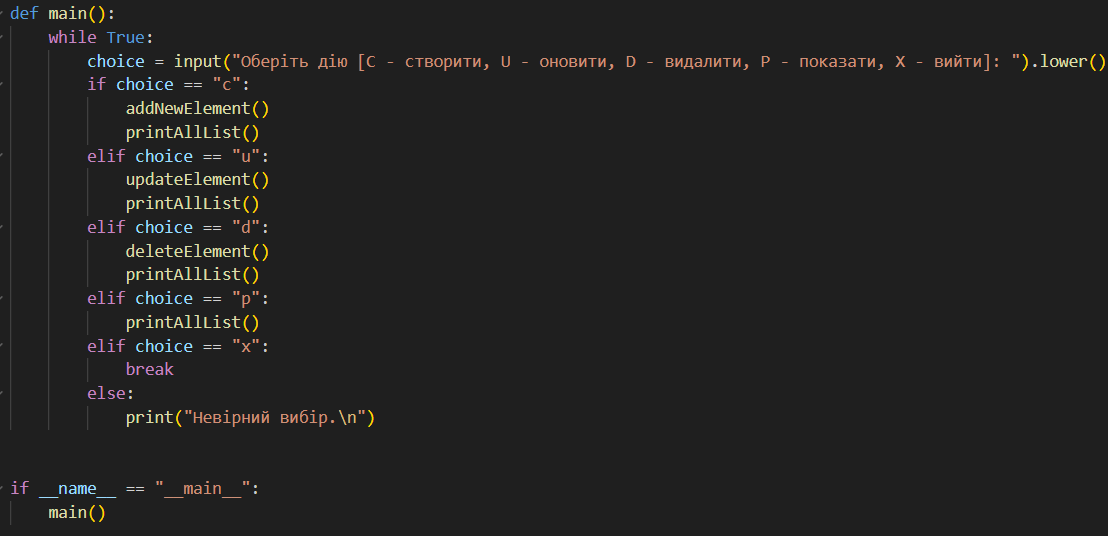
Стало:  
  
**6.** Реалізовано нову функцію updateElement() та реалізовані такі функції як пошук студента за іменем, введення нових даних (або залишення старих), видалення старого запису, вставка оновленого у відсортоване місце.

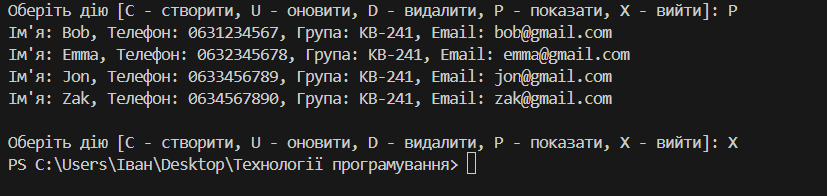
Було:  


Стало:  


7. Оновлено main(), замість конструкції match-case використано if/elif, це трохи спрощує код.

Було:  


Стало:  


Результат роботи програми:   
  
  
 **Висновок**

Початковий код містив лише два поля — ім’я та номер телефону, а також функції для додавання й видалення записів. Програму було доповнено двома новими полями — група та електронна пошта, оновлено функції виведення і додавання студентів, а також повністю реалізовано функцію зміни даних із збереженням відсортованого списку. У результаті отримано повноцінний відсортований довідник студентів, який дозволяє додавати, редагувати, видаляти та переглядати записи, використовуючи списки та словники в Python.  
  
Посилання на GitHub:  
<https://github.com/Ivan3nc/KB-241-Tech-Program-Shatun>