**Звіт про виконання Лабораторної роботи №1 з курсу Технології програмування на мові Python**

**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.

**Завдання до лабораторної роботи**

Реалізувати **відсортований** телефонний довідник студентів групи.

Для виконання задання надано частину готового функціоналу, яка розміщена в одній директорії з завданням до лабораторної роботи та має назву **lab\_01.py.**

Частина готового функціоналу реалізує безкінечний цикл запитів до користувача. Типи запитів: додати нового студента, змінити данні про існуючого студента, видалити запис, роздрукувати всю таблицю та вихід із програми. Реалізован функціонал додавання нового запису та видалення існуючого. Всі дії відбуваються з **відсортованим** списком студентів.

Перед виконанням роботи слід ознайомитись з існуючим функціоналом.

Необхідно розширити відомості про студента до 4х полів. На даний час використовується лише два поля (name та phone).

Необхідно переробити існуючий функціонал враховуючи розширення відомості про студента до 4х полів.

Необхідно реалізувати з нуля функціонал зміни інформації про студента враховуючи той факт, що вже існує реалізація додавання нового запису та видалення існуючого. **При зміні інформації про студента список має залишатись відсортованим**.

**Хід виконання завдання**

Текст програми:

|  |
| --- |
| # lab\_01.py  # Список студентів (вже відсортований за ім'ям)  students = [      {"name": "Bob", "phone": "0631234567", "group": "KB-241", "email": "bob@mail.com"},      {"name": "Emma", "phone": "0631234567", "group": "KB-242", "email": "emma@mail.com"},      {"name": "Jon", "phone": "0631234567", "group": "KB-242", "email": "jon@mail.com"},      {"name": "Zak", "phone": "0631234567", "group": "KB-243", "email": "zak@mail.com"}  ]  # --- Друк списку студентів ---  def printAllList():      print("\n--- Поточний довідник студентів ---")      for s in students:          print(f"Name: {s['name']}, Phone: {s['phone']}, Group: {s['group']}, Email: {s['email']}")      print("------------------------------------")  # --- Додавання нового студента ---  def addNewElement():      print("\n--- Додавання нового студента ---")      name = input("Введіть ім'я: ").strip()      phone = input("Введіть телефон: ").strip()      group = input("Введіть групу: ").strip()      email = input("Введіть email: ").strip()      new\_student = {"name": name, "phone": phone, "group": group, "email": email}      # Знаходимо позицію для вставки, щоб зберегти сортування      insert\_pos = 0      for s in students:          if name.lower() > s["name"].lower():              insert\_pos += 1          else:              break      students.insert(insert\_pos, new\_student)      print(f" Студента {name} додано.")  # --- Видалення студента ---  def deleteElement():      print("\n--- Видалення студента ---")      name = input("Введіть ім'я студента для видалення: ").strip()      for i, s in enumerate(students):          if s["name"].lower() == name.lower():              del students[i]              print(f" Студента {name} видалено.")              return      print(f" Студента {name} не знайдено.")  # --- Оновлення даних студента ---  def updateElement():      print("\n--- Зміна інформації про студента ---")      name = input("Введіть ім'я студента для оновлення: ").strip()      for i, s in enumerate(students):          if s["name"].lower() == name.lower():              print(f"Знайдено студента: {s['name']}, {s['phone']}, {s['group']}, {s['email']}")              print("Введіть нові дані (натисніть Enter, щоб залишити без змін):")              new\_name = input(f"Нове ім'я [{s['name']}]: ").strip()              new\_phone = input(f"Новий телефон [{s['phone']}]: ").strip()              new\_group = input(f"Нова група [{s['group']}]: ").strip()              new\_email = input(f"Новий email [{s['email']}]: ").strip()              # Оновлюємо поля, якщо введено нові значення              if new\_name != "":                  s["name"] = new\_name              if new\_phone != "":                  s["phone"] = new\_phone              if new\_group != "":                  s["group"] = new\_group              if new\_email != "":                  s["email"] = new\_email              # Видаляємо та вставляємо знову, щоб зберегти сортування              updated\_student = s              del students[i]              insert\_pos = 0              for st in students:                  if updated\_student["name"].lower() > st["name"].lower():                      insert\_pos += 1                  else:                      break              students.insert(insert\_pos, updated\_student)              print(" Інформацію про студента оновлено.")              return      print(f" Студента {name} не знайдено.")  # --- Основне меню ---  def main():      while True:          choice = input("\nВиберіть дію [C - створити, U - оновити, D - видалити, P - друк, X - вихід]: ").strip().upper()          match choice:              case "C":                  addNewElement()                  printAllList()              case "U":                  updateElement()                  printAllList()              case "D":                  deleteElement()                  printAllList()              case "P":                  printAllList()              case "X":                  print("\n Вихід з програми...")                  break              case \_:                  print(" Невірний вибір, спробуйте ще раз.")  main() |

Посилання на github: <https://github.com/TP-KB-242-Kobzar-Roman/TP-KB-242-Kobzar-Roman/blob/main/TP-KB-242-Kobzar-Roman-lpr.docx>

Знімок екрану з посилання на github:

