**Лабораторна робота №1**

**Тема**: Списки. Словники. Кортежі.

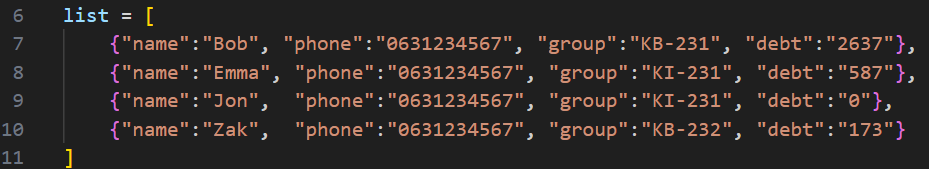
**Мета**: Використовуючи теоретичне підґрунтя про складні структури даних Списки, Словники Кортежі, та використовуючи існуючий код, доробити програму додавши функціонал, що буде вказано в завданні до лабораторної роботи.

**Завдання до лабораторної роботи**

1. Розширити відомості про студента до 4х полів. На даний час використовується лише два поля (name та phone).
2. Переробити існуючий функціонал враховуючи розширення відомості про студента до 4х полів.
3. 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 роботи**

Для початку я додав 2 поля до елементів списку:



Далі переробив існуючі функції додавання та друкування для роботи з 4 полями елементів списку:



Останнім була реалізація функції зміни інформації про студента, в завданні було сказано враховувати той факт, що вже існує реалізація додавання нового запису та видалення існуючого, тому можна просто взяти ті частини коду, з видалення можна взяти пошук позиції за іменем елемента, а з додавання можна взяти сортування перед додаванням елемента. Заміна працює наступним чином: спочатку шукається потрібний елемент з відповідним іменем, якщо його знайдено, то поточна інформація елемента зберігається та виводиться користувачеві, після цього він може ввести змінені дані, або просто пропустити поле вводу, щоб інформація не змінювалась, далі йде перевірка нових даних, якщо всі дані збігаються із початковими, то заміна не відбувається, якщо ж збігається тільки ім’я, то просто змінюється інформація в 3 інших полях, а якщо ім’я було змінене, то формується новий елемент із введеними або залишеними даними, видаляється старий і шукається позиція для вставки нового, після чого цей елемент додається на знайдену позицію:



Текст програми:

|  |
| --- |
| ## List [Item1, Item2, Item3]  ## Item {"name":"Jon", "phone":"0631234567"}  # already sorted list  list = [      {"name":"Bob", "phone":"0631234567", "group":"KB-231", "debt":"2637"},      {"name":"Emma", "phone":"0631234567", "group":"KI-231", "debt":"587"},      {"name":"Jon",  "phone":"0631234567", "group":"KI-231", "debt":"0"},      {"name":"Zak",  "phone":"0631234567", "group":"KB-232", "debt":"173"}  ]  def printAllList():      for elem in list:          strForPrint = "Student name is " + elem["name"] + ",  Phone is " + elem["phone"] + ",  Group is " + elem["group"] + ",  debt is " + elem["debt"] + " UAH"          print(strForPrint)      return  def addNewElement():      name = input("Pease enter student name: ")      phone = input("Please enter student phone: ")      group = input("Pease enter student group: ")      debt = input("Please enter student debt (in UAH): ")      newItem = {"name": name, "phone": phone, "group": group, "debt": debt}      # find insert position      insertPosition = 0      for item in list:          if name > item["name"]:              insertPosition += 1          else:              break      list.insert(insertPosition, newItem)      print("New element has been added")      return  def deleteElement():      name = input("Please enter name to be delated: ")      deletePosition = -1      for item in list:          if name == item["name"]:              deletePosition = list.index(item)              break      if deletePosition == -1:          print("Element was not found")      else:          print("Delete position " + str(deletePosition))          # list.pop(deletePosition)          del list[deletePosition]      return  def updateElement():      name = input("Please enter name to be updated: ")      updatePosition = -1      for item in list:          if name == item["name"]:              updatePosition = list.index(item)              break      if updatePosition == -1:          print("Student not found")      else:          name1 = list[updatePosition]["name"]          phone1 = list[updatePosition]["phone"]          group1 = list[updatePosition]["group"]          debt1 = list[updatePosition]["debt"]          curinf = "Student current iformation: name — " + name1 + " , phone — " + phone1 + ", group — " + group1 + " , debt — " + debt1          print(curinf)          name = input("Enter new name or press Enter to skip:") or name1          phone = input("Enter new phone or press Enter to skip:") or phone1          group = input("Enter new group or press Enter to skip:") or group1          debt = input("Enter new debt or press Enter to skip:") or debt1            if name == name1 and phone == phone1 and group == group1 and debt == debt1:              print("you haven`t updated student information")          elif name == name1:              list[updatePosition]["phone"] = phone              list[updatePosition]["group"] = group              list[updatePosition]["debt"] = debt          else:              updatedItem = {"name": name, "phone": phone, "group": group, "debt": debt}              del list[updatePosition]              insertPosition = 0              for item in list:                  if name > item["name"]:                      insertPosition += 1              list.insert(insertPosition, updatedItem)          print("information has been updated")      return      # implementation required  def main():      while True:          chouse = input("Please specify the action [ C create, U update, D delete, P print,  X exit ] ")          match chouse:              case "C" | "c":                  print("New element will be created:")                  addNewElement()                  printAllList()              case "U" | "u":                  print("Existing element will be updated")                  updateElement()              case "D" | "d":                  print("Element will be deleted")                  deleteElement()              case "P" | "p":                  print("List will be printed")                  printAllList()              case "X" | "x":                  print("Exit()")                  break              case \_:                  print("Wrong chouse")  main() |

**Висновок:** під час виконання роботи я використовуючи теоретичне підґрунтя про складні структури даних Списки, Словники Кортежі, та використовуючи існуючий код, доробив програму додавши вказаний функціонал.