**Звіт про виконання лабораторних завдань до лекцій з курсу Технології програмування на мові Python**

Лабораторна робота №3

**ООП.**

**Мета роботи**: Використовуючи теоретичне підґрунтя про ООП у мові Python переробити програму телефонного довідника студентів використовуючи принципи ООП для формування відомостей про студентів.

**ООП та мова Python**

Python має безліч вбудованих типів, наприклад, int, str і так далі, які ми можемо використовувати у програмі. Але Python також дозволяє визначати власні типи за допомогою класів. Клас є деякою сутністю. Конкретним здійсненням класу є об'єкт.

Можна ще провести таку аналогію. У нас у всіх є деяке уявлення про людину, яка має ім'я, вік, якісь інші характеристики Людина може виконувати деякі дії - ходити, бігати, думати і т.д. Тобто це уявлення, яке включає набір характеристик та дій, можна назвати класом. Конкретне втілення цього шаблону може відрізнятися, наприклад, одні мають одне ім'я, інші - інше ім'я. І реально існуюча людина представлятиме об'єкт цього класу.

**Завдання до лабораторної роботи**

Переробити функціональність телефонного довідника студентів групи, що був розроблений у Лабораторній роботі №2 використовуючи принципи ООП:

1. розробити клас Студент групи з відповідними атрибутами;
2. розробити клас Список групи, має містити не словники, як виконано в лабораторній роботі №2, а об’єкти класу Студент групи; додавання нового запису, видаленні існуючого чи зміна даних має бути виконана через методи класу Список групи.
3. розробити клас для роботи з файлами для зчитування початкової інформації про список групи та збереження інформації по завершенню програми.
4. список студентів має містити не словники, як виконано в лабораторній роботі №2, а об’єкти класу Студент групи;
5. описання всіх класів мають міститися в окремих файлах, що мають відповідні імена(наприклад Studen, StudentList, Utils)
6. основний функціонал програми має бути покритий Юніт тестами

Хід виконання завдання:

1. Відкрила ВСкод
2. Прочитала завдання
3. Виконала завдання
4. Затестила
5. Закинула в звіт

**lab3.py**

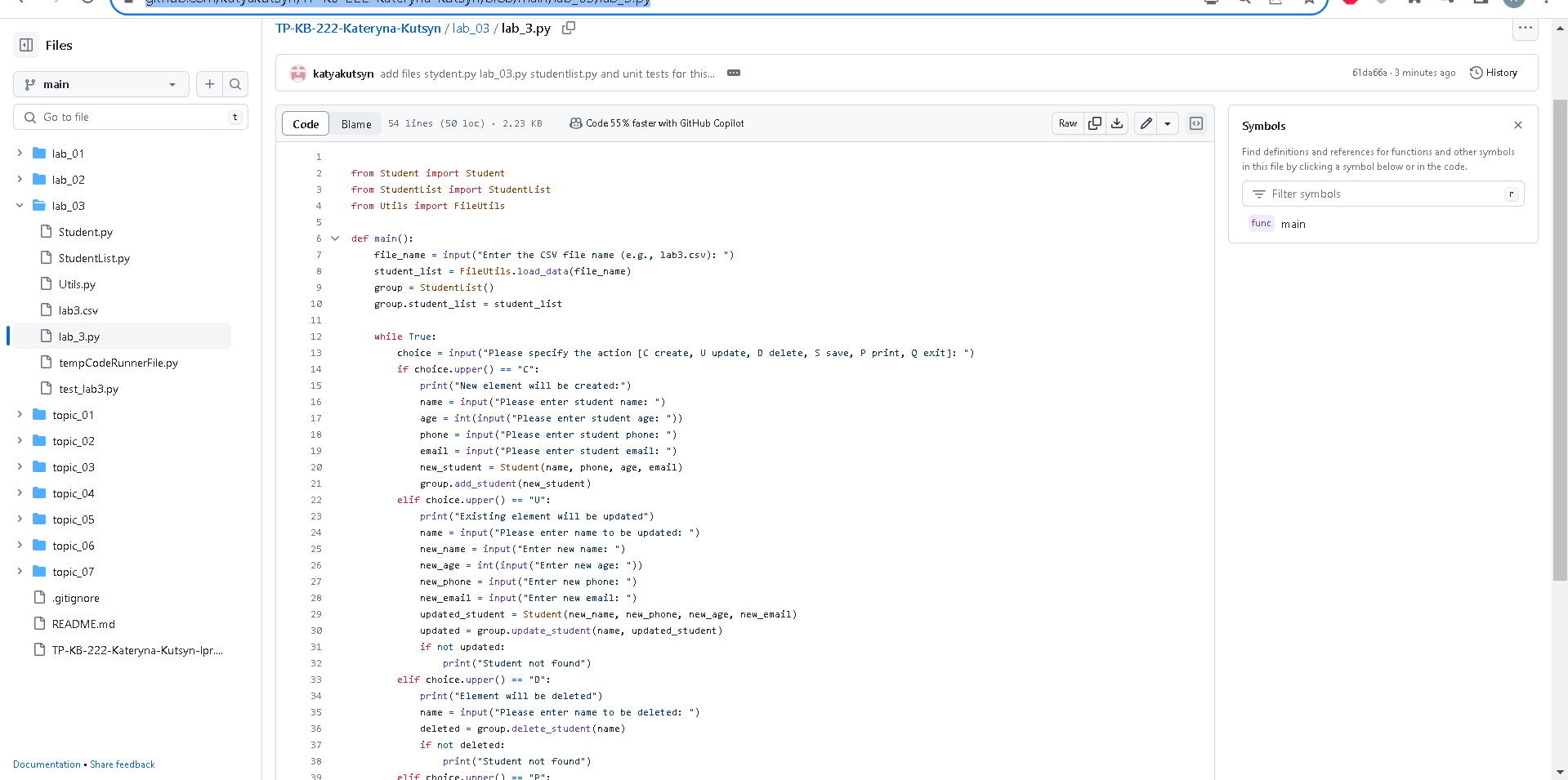
Текст програми:

|  |
| --- |
| from Student import Student  from StudentList import StudentList  from Utils import FileUtils  def main():  file\_name = input("Enter the CSV file name (e.g., lab3.csv): ")  student\_list = FileUtils.load\_data(file\_name)  group = StudentList()  group.student\_list = student\_list  while True:  choice = input("Please specify the action [C create, U update, D delete, S save, P print, Q exit]: ")  if choice.upper() == "C":  print("New element will be created:")  name = input("Please enter student name: ")  age = int(input("Please enter student age: "))  phone = input("Please enter student phone: ")  email = input("Please enter student email: ")  new\_student = Student(name, phone, age, email)  group.add\_student(new\_student)  elif choice.upper() == "U":  print("Existing element will be updated")  name = input("Please enter name to be updated: ")  new\_name = input("Enter new name: ")  new\_age = int(input("Enter new age: "))  new\_phone = input("Enter new phone: ")  new\_email = input("Enter new email: ")  updated\_student = Student(new\_name, new\_phone, new\_age, new\_email)  updated = group.update\_student(name, updated\_student)  if not updated:  print("Student not found")  elif choice.upper() == "D":  print("Element will be deleted")  name = input("Please enter name to be deleted: ")  deleted = group.delete\_student(name)  if not deleted:  print("Student not found")  elif choice.upper() == "P":  print("List will be printed")  all\_students = group.get\_all\_students()  for student in all\_students:  str\_for\_print = f"Student name is {student.name}, Age is {student.age}, Phone is {student.phone}, Email is {student.email}"  print(str\_for\_print)  elif choice.upper() == "S":  FileUtils.save\_data(file\_name, group.student\_list)  elif choice.upper() == "Q":  print("Dosviduli")  break  else:  print("Dosviduli")  if \_\_name\_\_ == "\_\_main\_\_":  main() |

Посилання на github:

[https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/lab\_03/lab\_3.py](https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/lab_03/lab_3.py%20)

Знімок екрану з посилання на github:



**Student.py**

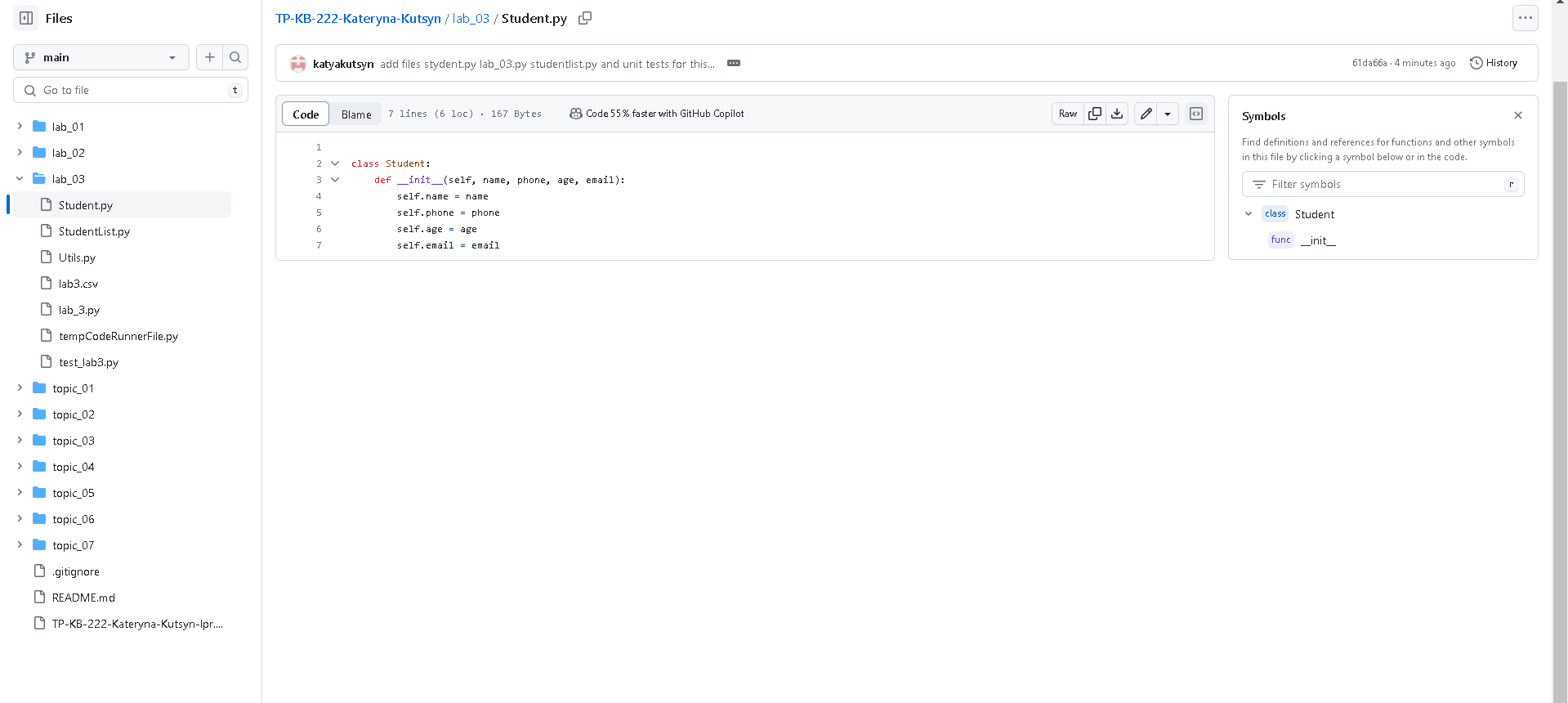
Текст програми:

|  |
| --- |
| class Student:  def \_\_init\_\_(self, name, phone, age, email):  self.name = name  self.phone = phone  self.age = age  self.email = email |

Посилання на github:

<https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/lab_03/Student.py>

Знімок екрану з посилання на github:



**StudentList.py**

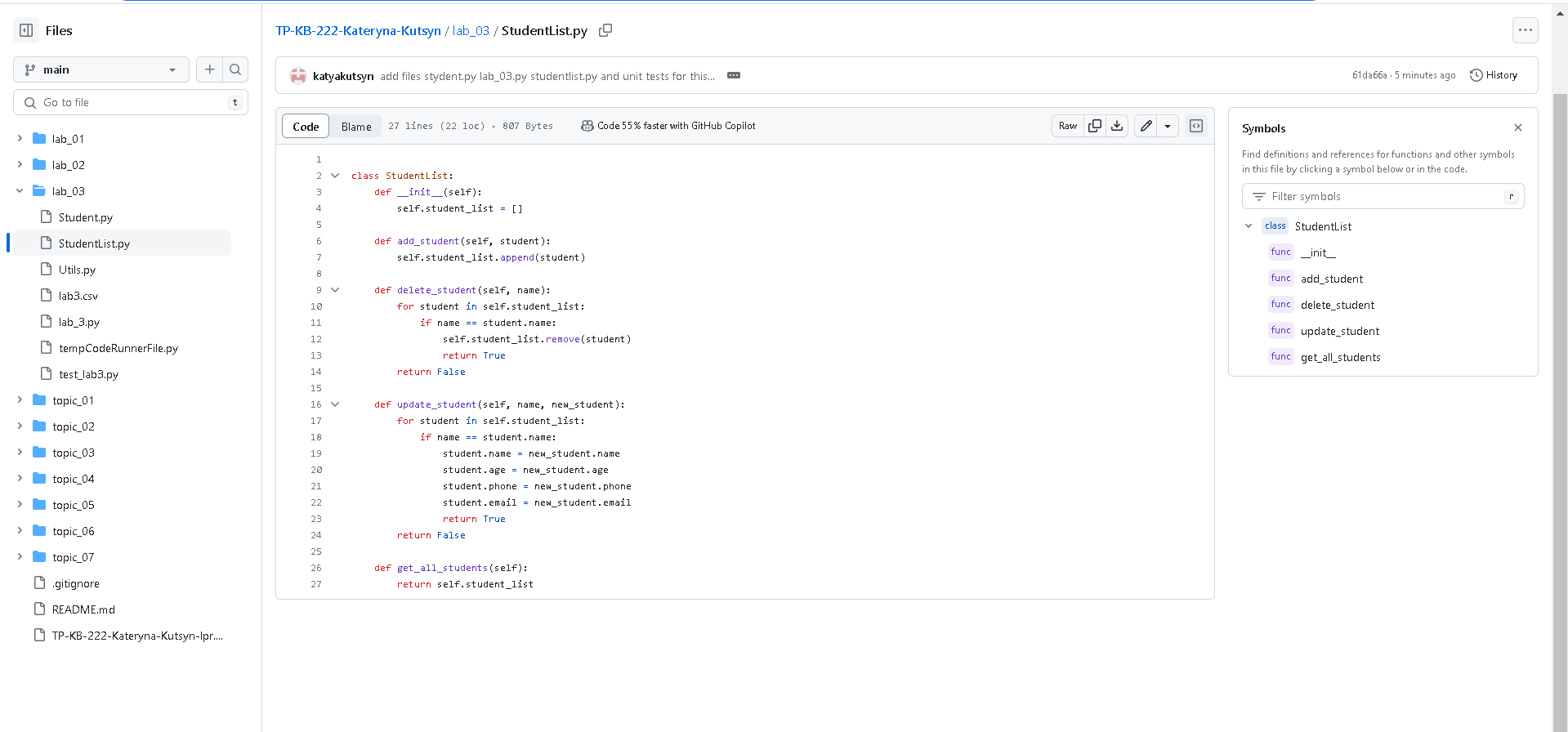
Текст програми:

|  |
| --- |
| class StudentList:  def \_\_init\_\_(self):  self.student\_list = []  def add\_student(self, student):  self.student\_list.append(student)  def delete\_student(self, name):  for student in self.student\_list:  if name == student.name:  self.student\_list.remove(student)  return True  return False  def update\_student(self, name, new\_student):  for student in self.student\_list:  if name == student.name:  student.name = new\_student.name  student.age = new\_student.age  student.phone = new\_student.phone  student.email = new\_student.email  return True  return False  def get\_all\_students(self):  return self.student\_list |

Посилання на github:

<https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/lab_03/StudentList.py>

Знімок екрану з посилання на github:



**Utils.py**

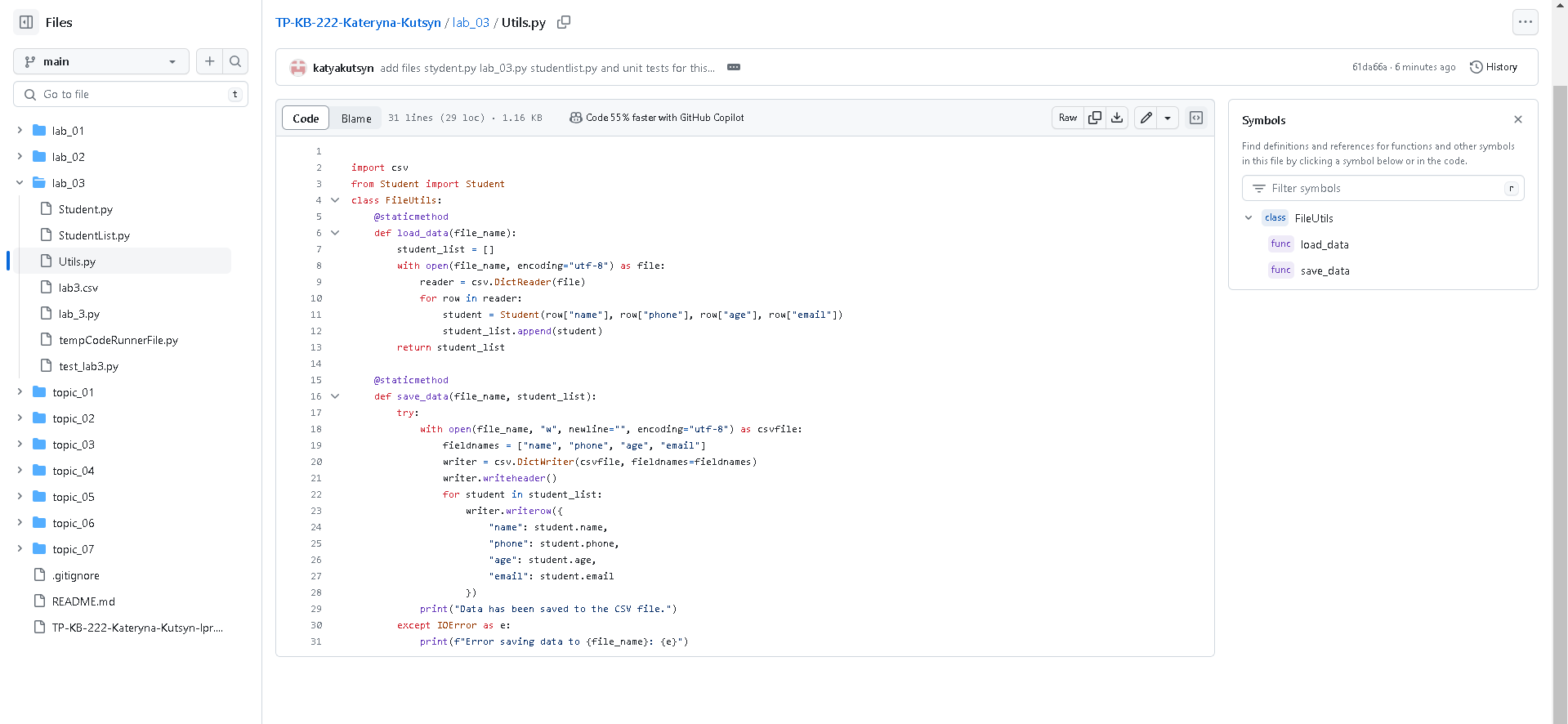
Текст програми:

|  |
| --- |
| import csv  from Student import Student  class FileUtils:  @staticmethod  def load\_data(file\_name):  student\_list = []  with open(file\_name, encoding="utf-8") as file:  reader = csv.DictReader(file)  for row in reader:  student = Student(row["name"], row["phone"], row["age"], row["email"])  student\_list.append(student)  return student\_list  @staticmethod  def save\_data(file\_name, student\_list):  try:  with open(file\_name, "w", newline="", encoding="utf-8") as csvfile:  fieldnames = ["name", "phone", "age", "email"]  writer = csv.DictWriter(csvfile, fieldnames=fieldnames)  writer.writeheader()  for student in student\_list:  writer.writerow({  "name": student.name,  "phone": student.phone,  "age": student.age,  "email": student.email  })  print("Data has been saved to the CSV file.")  except IOError as e:  print(f"Error saving data to {file\_name}: {e}") |

Посилання на github:

<https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/lab_03/Utils.py>

Знімок екрану з посилання на github:



**Test\_lab3.py**

Текст програми:

|  |
| --- |
| import unittest  from Student import Student  from StudentList import StudentList  class TestStudentList(unittest.TestCase):  def setUp(self):  self.student\_list = StudentList()  def test\_student\_list\_initialization(self):  self.assertEqual(len(self.student\_list.get\_all\_students()), 0)  def test\_student\_list\_add\_student(self):  student = Student("John", "1234567890", 20, "john@example.com")  self.student\_list.add\_student(student)  students = self.student\_list.get\_all\_students()  self.assertEqual(len(students), 1)  self.assertIn(student, students)  def test\_student\_list\_delete\_student(self):  student = Student("John", "1234567890", 20, "john@example.com")  self.student\_list.add\_student(student)  self.student\_list.delete\_student("John")  self.assertEqual(len(self.student\_list.get\_all\_students()), 0)  if \_\_name\_\_ == '\_\_main\_\_':  unittest.main() |

Посилання на github:

<https://github.com/katyakutsyn/TP-KB-222-Kateryna-Kutsyn/blob/main/lab_03/test_lab3.py>

Знімок екрану з посилання на github:

