**Лабораторная работа №9**

**ИНКАПСУЛЯЦИЯ**

**Цель работы**

Научиться работать с механизмом сокрытия компонентов программы.

**Задание:**

1. Решите задания из приложенного файла ООП. Задание 5. Инкапсуляция. Исключения.

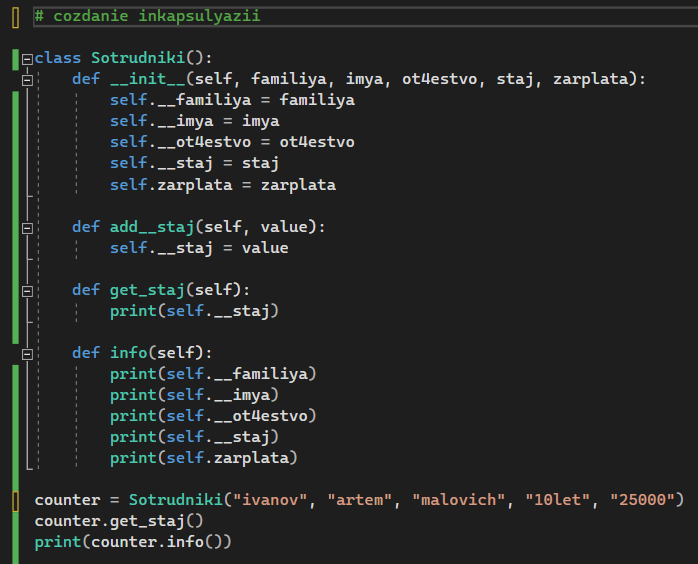


Рисунок 1 – преобразование атрибутов и методов в приватные (инкапсуляция)

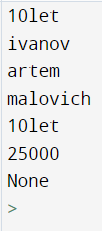


Рисунок 2 – результат кода

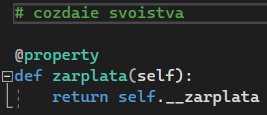


Рисунок 3 – создание свойства

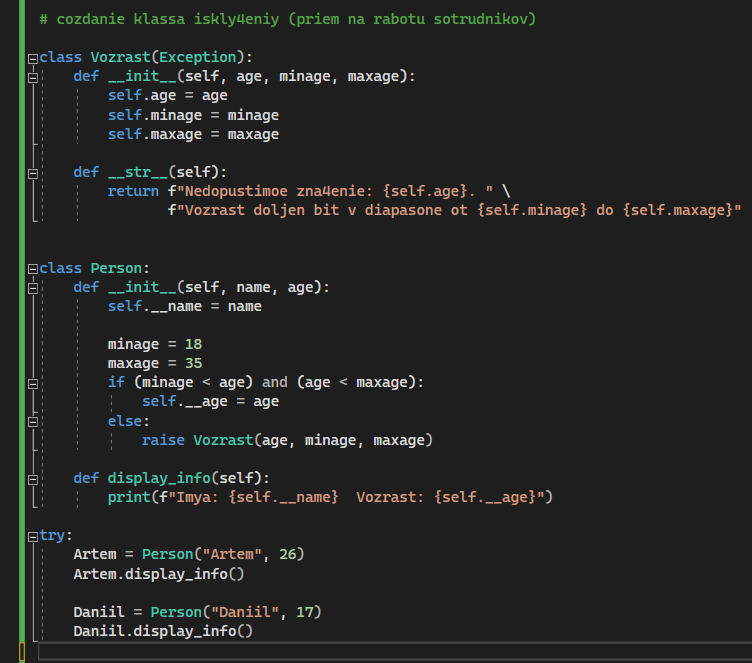


Рисунок 4 – создание класса исключений



Рисунок 5 – результат программы

**Контрольные вопросы:**

**1. Что представляет собой инкапсуляция?**

Это объединение функций и данных в рамках одной структуры, внутреннее состояние которой (данные) скрыто от внешнего мира.

**2. Какие уровни сокрытия можно выделить в языках программирования; в чем их различия?**

* Private – только внутри методов данного класса.
* Protected - внутри методов данного класса и методов всех его потомков.
* Public - любое место программы, в том числе во всех классах и их методах.
* Internal - во всех файлах, необходимых для компиляции программы.

**3. Каким образом инкапсуляция реализована в языке Python?**

Инкапсуляция в Python работает лишь на уровне соглашения между программистами о том, какие атрибуты являются общедоступными, а какие — внутренними.

**4. Каким образом инкапсуляция реализована в языке C#?**

Инкапсуляция в C# осуществляется при помощи модификаторов доступа (private, public…). Конечным пользователем объекта здесь выступает либо объект наследник, либо программист.

**Вывод:**

В ходе лабораторной работы я научился работать с механизмом сокрытия компонентов программы.