Лабораторна робота

(для учнів 7 класу)

Тема: Вступний курс з об’єктно-орієнтованого програмування. Інкапсуляція.

Мета: Ознайомитися з основами ООП, зокрема з поняттям інкапсуляції. Навчитися створювати класи та застосовувати інкапсуляцію в Python.

Завдання:

Ознайомитися з теоретичною частиною щодо інкапсуляції та роботи з класами в Python. Створити клас, який має приватні атрибути. Написати методи для доступу до кожного атрибуту (getters) та для їх зміни (setters).

Хід роботи

Об'єктно-орієнтоване програмування (ООП) – це підхід до програмування, де програма розглядається як набір об'єктів, що взаємодіють між собою.

Основні поняття ООП:

Клас – шаблон для створення об'єктів.

Об'єкт – конкретний екземпляр класу.

Інкапсуляція – принцип ООП, що дозволяє приховати внутрішні деталі об'єкта і надавати до нього доступ лише через спеціальні методи (getters та setters).

Додаткові ресурси з теоретичним матеріалом:

1. Принципи ООП
2. Поняття інкапсуляції, структура класу

Приклад інкапсуляції:

```
class Student:
    def __init__(self, name, age):
        self.__name = name  # Приватна змінна
        self.__age = age    # Приватна змінна

    # Метод для отримання значення змінної __name
    def get_name(self):
        return self.__name

    # Метод для зміни значення змінної __name
    def set_name(self, name):
        self.__name = name

    # Метод для отримання значення змінної __age
    def get_age(self):
        return self.__age

    # Метод для зміни значення змінної __age
    def set_age(self, age):
        if age > 0:
            self.__age = age
        else:
            print("Вік має бути більше 0")

# Створюємо об'єкт класу Student
student = Student("Оля", 12)
# Використовуємо методи для доступу до приватних змінних
print(student.get_name())  # Виведе: Оля
student.set_age(13)
print(student.get_age())   # Виведе: 13
```

Завдання:

1. Створіть клас Car, який має приватні атрибути марка, модель та рік випуску.
2. Напишіть методи для доступу до кожного атрибуту (getters) та для їх зміни (setters).
3. Створіть об'єкт класу Car та задайте значення його атрибутів. Виведіть їх на екран.

Алгоритм виконання:

1. Створіть новий файл з розширенням .py у середовищі розробки Python.
2. Реалізуйте клас Car з використанням приватних атрибутів та методів для їх доступу і зміни.
3. Створіть об'єкт цього класу і протестуйте методи.

Додаткове завдання:

Додайте в клас Car метод для обчислення віку автомобіля на основі поточного року.

Висновок: учні ознайомилися з основами ООП, зокрема з поняттям інкапсуляції. Навчилися створювати класи та застосовувати інкапсуляцію в Python.