**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ**

**“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО”**

**Факультет прикладної математики**

**Лабораторна робота №1**

**Тема: «Що таке ООП. Створення об'єктів»**

**Виконала:** студентка групи КМ-31

Буряк Софія

**КИЇВ — 2024**

# **Результати виконаної роботи:**

# **Посилання на виконану роботу:** <https://github.com/sofiia-buriak/OOP>

## 

# **Відповіді на запитання:**

## **1. Що таке ООП?**

Об'єктно-орієнтоване програмування (ООП) — це парадигма програмування, яка базується на концепції "об'єктів", що взаємодіють одне з одним.

**Джерела:**

1. [Wikipedia: Об'єктно-орієнтоване програмування](https://uk.wikipedia.org/wiki/%D0%9E%D0%B1%27%D1%94%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D1%96%D1%94%D0%BD%D1%82%D0%BE%D0%B2%D0%B0%D0%BD%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F)
2. [DanIT: ООП - Принципи та переваги](https://dan-it.com.ua/uk/blog/chto-takoe-obektno-orientirovannoe-programmirovanie-principy-preimushhestva-i-nedostatki/)

## **2. Основні принципи ООП**

Фундаментальні принципи ООП:

* **Інкапсуляція (Encapsulation)** — об'єднання даних і методів в одному об'єкті.
* **Наслідування (Inheritance)** — можливість створювати нові класи на основі вже існуючих.
* **Поліморфізм (Polymorphism)** — здатність об'єктів з однаковим інтерфейсом мати різні реалізації.
* **Абстракція (Abstraction)** — виділення суттєвих характеристик об'єкта, ігноруючи несуттєві деталі.

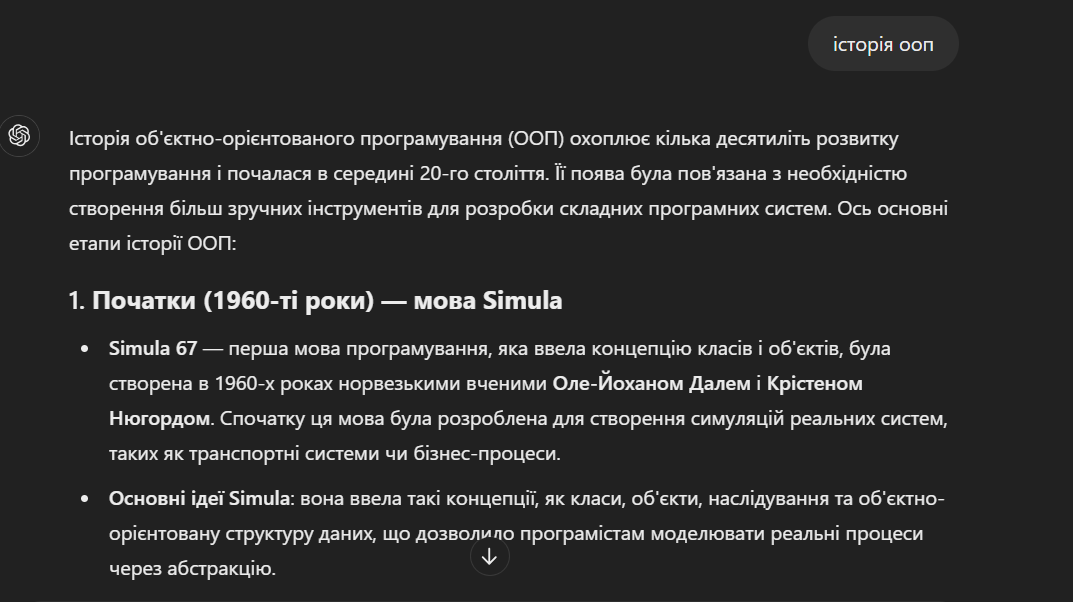
**Джерела:**

1. [Medium: Основні принципи ООП](https://medium.com/madit-story/%D1%89%D0%BE-%D1%82%D0%B0%D0%BA%D0%B5-%D0%BE%D0%BE%D0%BF-oop-%D0%BE%D1%81%D0%BD%D0%BE%D0%B2%D0%BD%D1%96-%D0%BF%D1%80%D0%B8%D0%BD%D1%86%D0%B8%D0%BF%D0%B8-8839c5b793ef)
2. [Wikipedia: Інкапсуляція](https://uk.wikipedia.org/wiki/%D0%86%D0%BD%D0%BA%D0%B0%D0%BF%D1%81%D1%83%D0%BB%D1%8F%D1%86%D1%96%D1%8F_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F))
3. [Wikipedia: Наслідування](https://uk.wikipedia.org/wiki/%D0%9D%D0%B0%D1%81%D0%BB%D1%96%D0%B4%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F))
4. [Wikipedia: Поліморфізм](https://uk.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BB%D1%96%D0%BC%D0%BE%D1%80%D1%84%D1%96%D0%B7%D0%BC_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F))

## **3. Як та коли придумали ООП?**

ООП було створено в 1960-х роках для вирішення проблем, пов'язаних зі зростанням складності програмного забезпечення. Винахідниками ООП вважаються Оле-Йохан Даль та Крістен Нюгорд, які розробили мову програмування Simula.

**Джерела:**



## **4. Альтернативні та додаткові стилі програмування**

* **Процедурне програмування:** програма ділиться на послідовні функції або процедури.
* **Функціональне програмування:** використовуються функції без зміни стану та побічних ефектів.
* **Декларативне програмування:** описується "що" потрібно зробити, а не "як" це зробити.

**Джерела:**

1. [Foxminded: Види програмування](https://foxminded.ua/vydy-prohramuvannia)
2. [Wikipedia: Декларативне програмування](https://uk.wikipedia.org/wiki/%D0%94%D0%B5%D0%BA%D0%BB%D0%B0%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F)
3. [Wikipedia: Функційне програмування](https://uk.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D1%96%D0%B9%D0%BD%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F)
4. [Wikipedia: Процедурне програмування](https://uk.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D0%B4%D1%83%D1%80%D0%BD%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F)

## **5. Переваги ООП**

Об'єктно-орієнтоване програмування дозволяє моделювати складні системи через абстракцію об'єктів і їх взаємодії, що полегшує управління та масштабування великих проектів у порівнянні з процедурним та функціональним підходами.

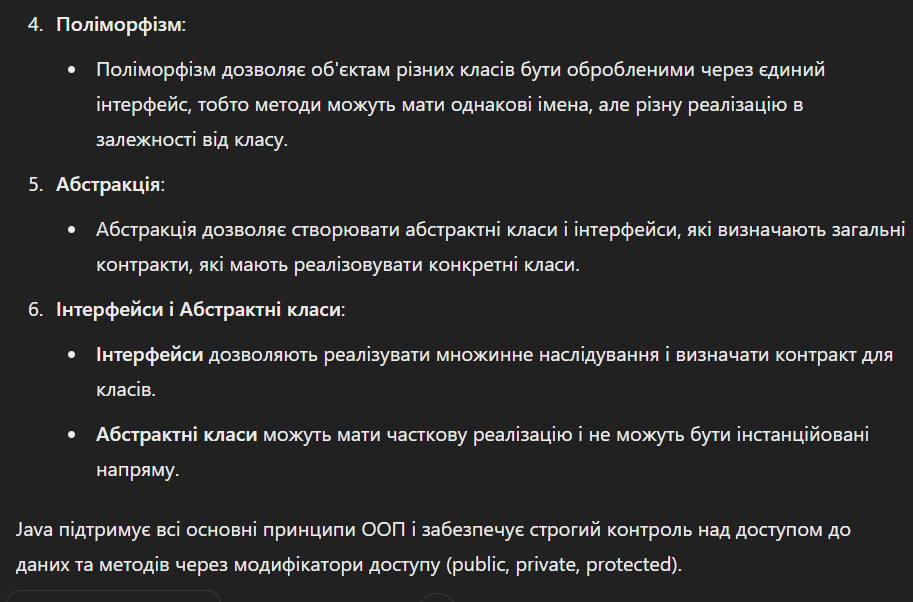
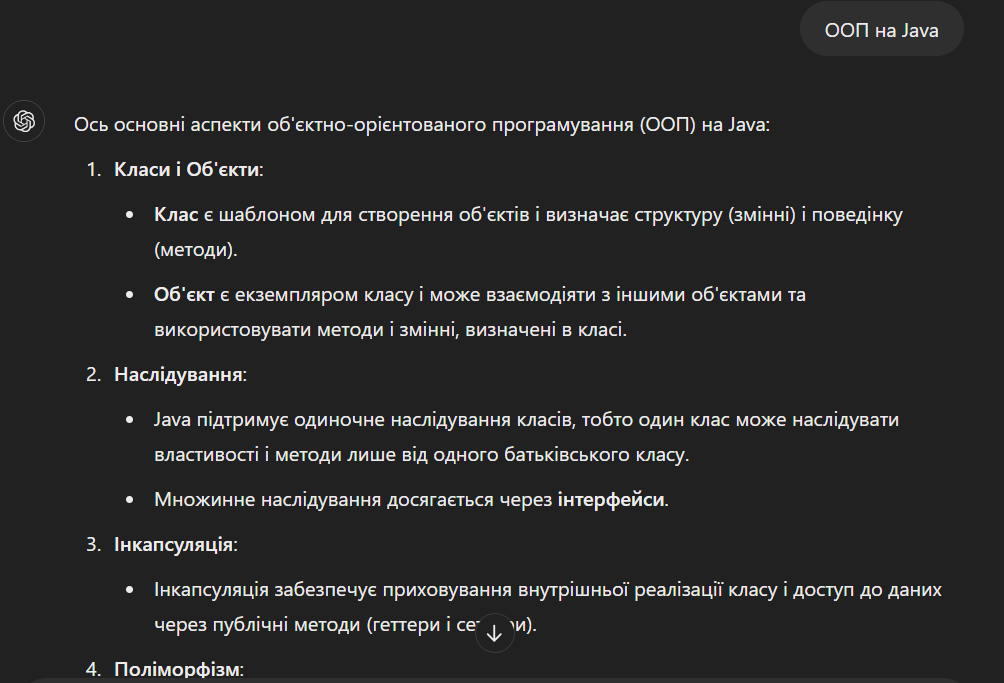
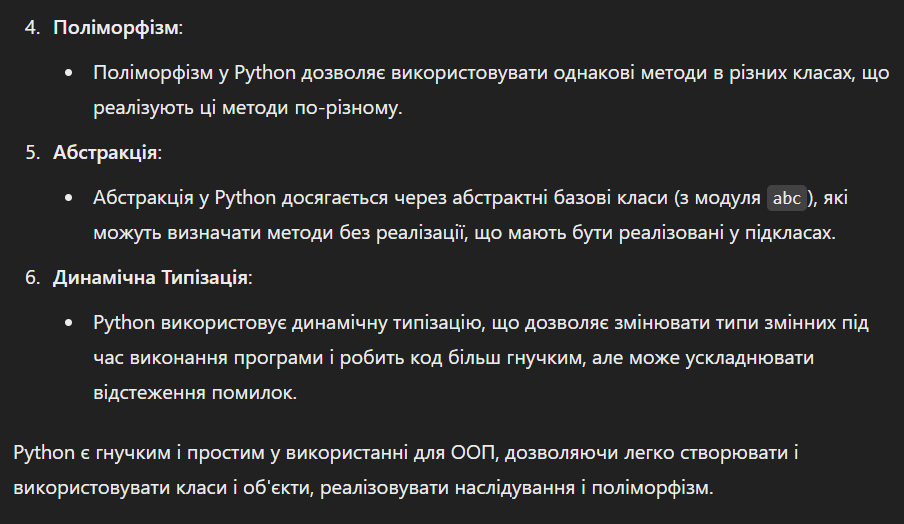
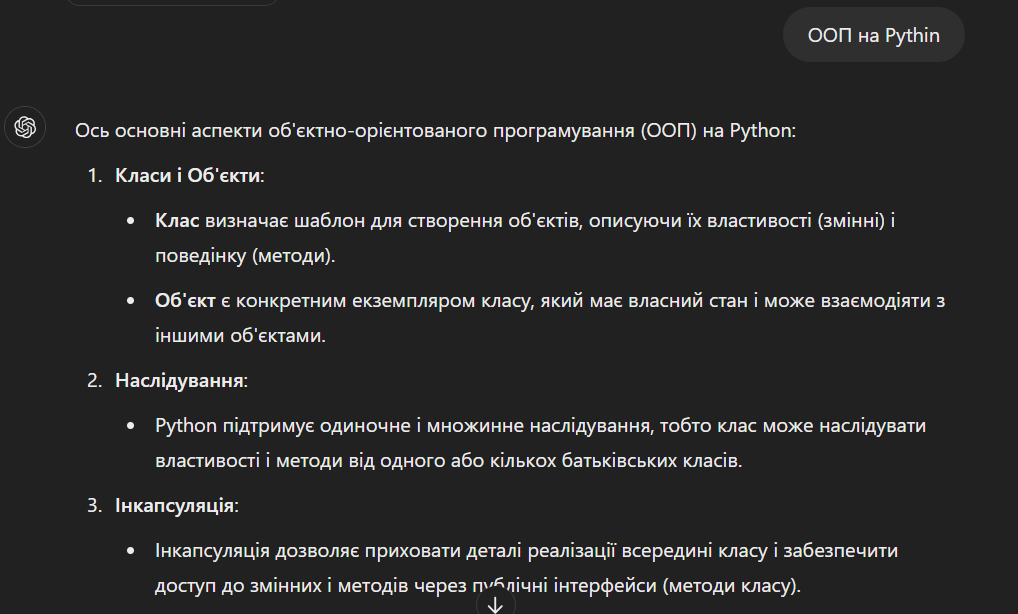
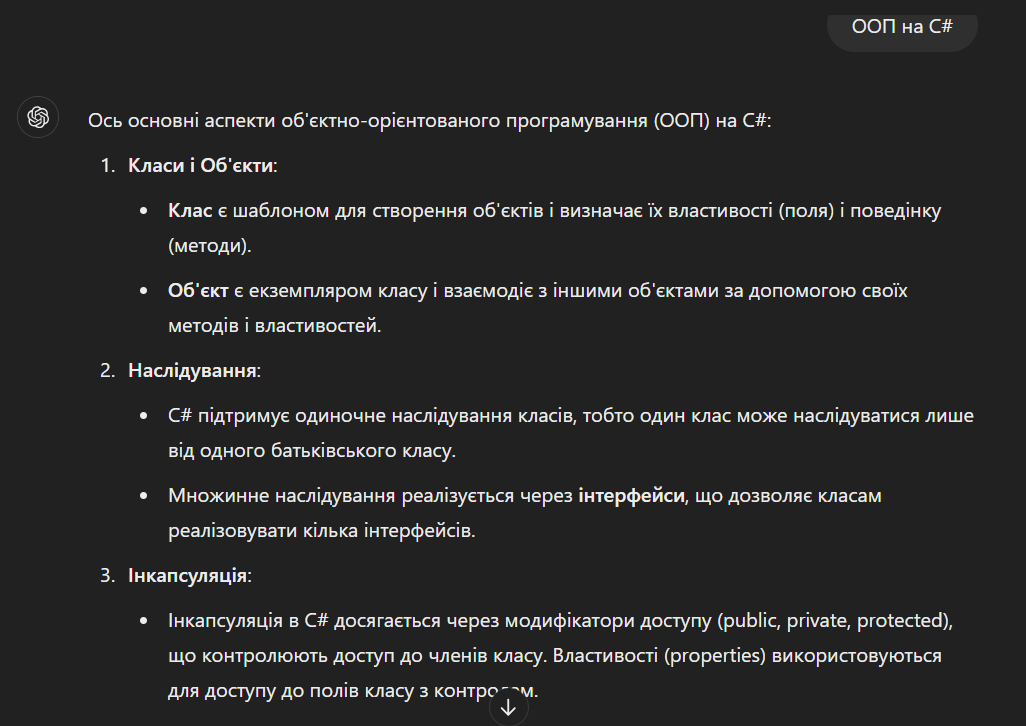
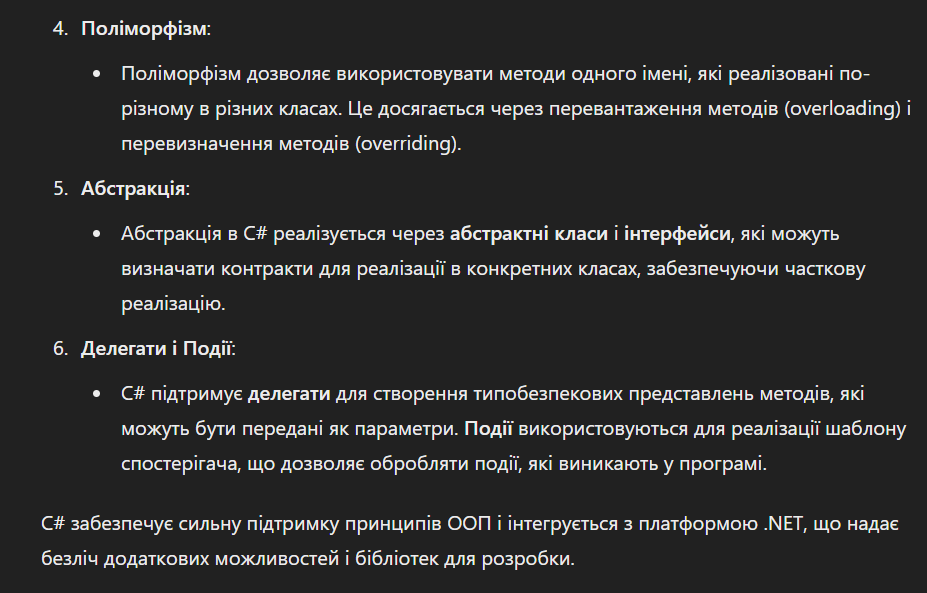
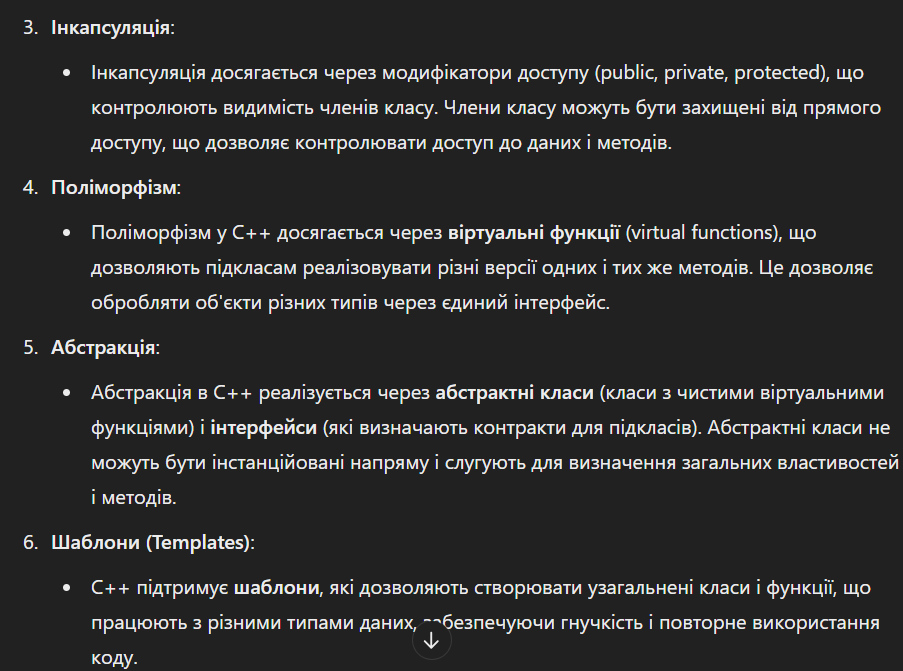
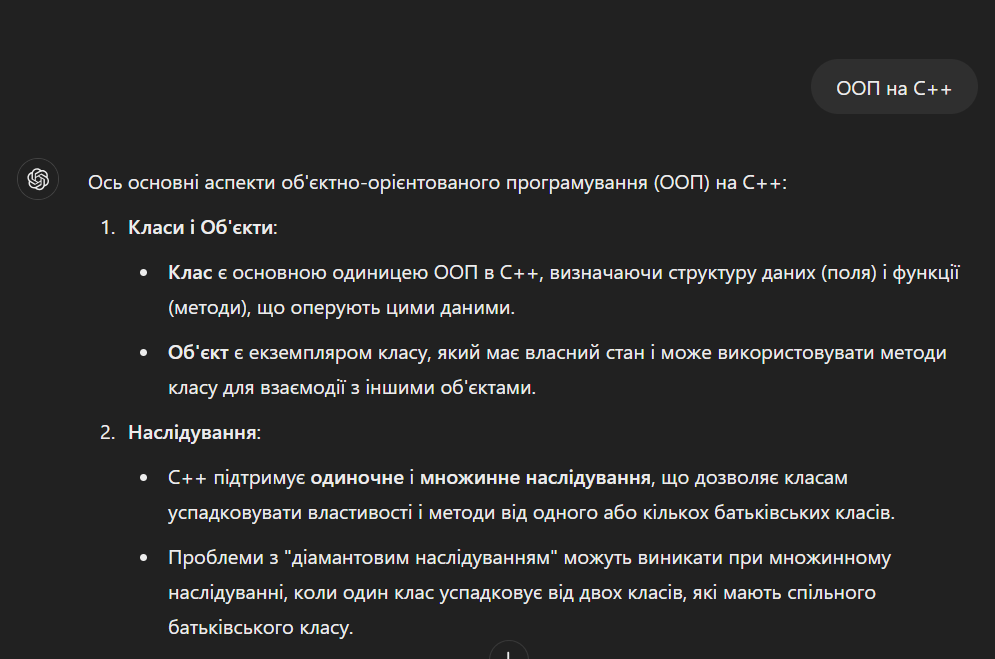
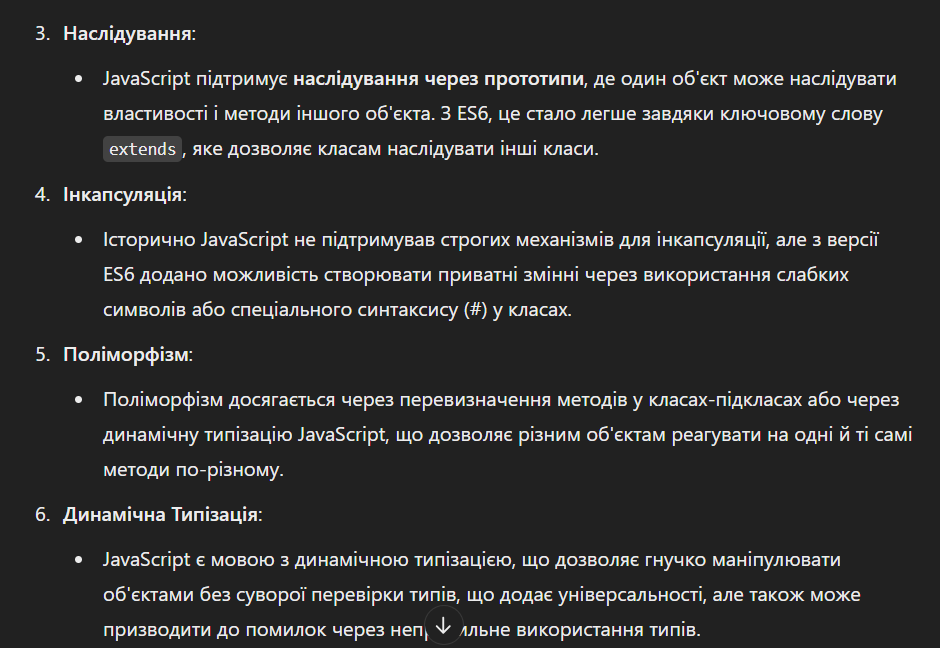
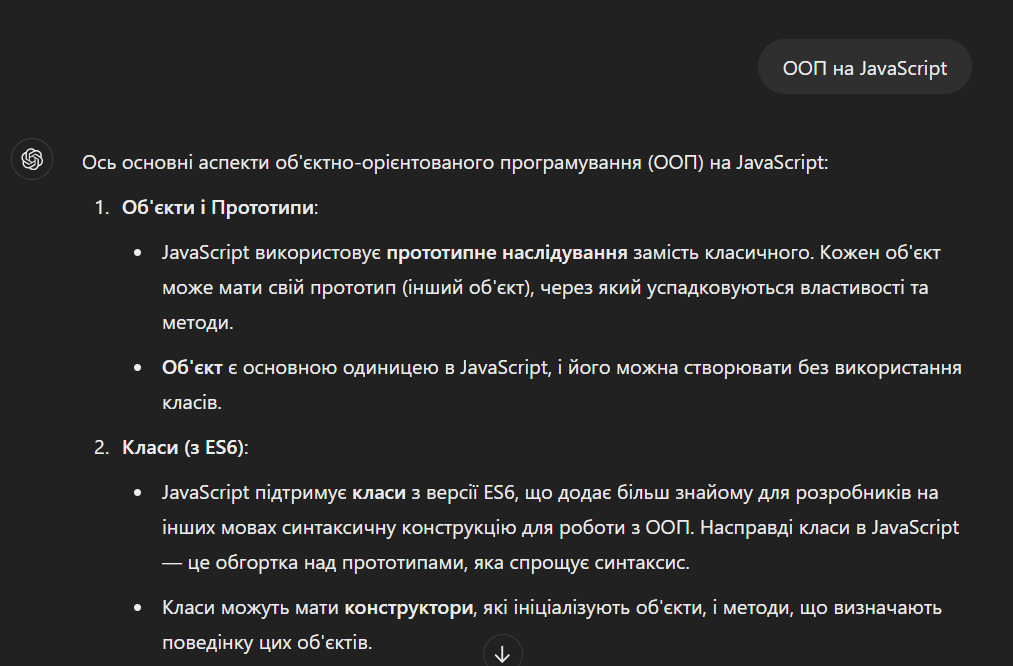
## **6. Які мови програмування підтримують ООП?**

Java, C#, C++, Python, Ruby, Objective-C, ActionScript 3, Swift, Vala.

## **7. Різниця у програмуванні в ООП стилі між мовами Java, C#, C++, Python, JS, PHP**

* **Java:** суворе дотримання ООП з одиночним наслідуванням класів, підтримка інтерфейсів.
* **C#:** одиночне наслідування та підтримка інтерфейсів, властивості та делегати.
* **C++:** підтримка одиночного та множинного наслідування, складнощі з "діамантовим наслідуванням".
* **Python:** гнучка мова з динамічною типізацією та підтримкою множинного наслідування.
* **JavaScript:** використовує прототипне наслідування, що відрізняється від класичної моделі.
* **PHP:** підтримує ООП, але часто комбінується з процедурним програмуванням.

**Джерела:**

* 1. 
  2. 
  3.  
  4. 
  5. 
  6. 