

Звіт-ретроспектива:

1)Які конкретні задачі планували вирішувати за допомогою цієї бібліотеки?

За допомогою бібліотеки PYGAME я планувала вирішити такі задачі:

- Графічний вивід: оскільки **pygame** надає можливість легко створювати вікна, обробляти події від користувача та виводити графіку.
- малювання графічних об'єктів: в моїй грі я використовую функції `draw_grid` та `draw_tic_tac_toe`, які використовують `pygame.draw.line` та `pygame.draw.circle` для малювання ліній та кола.

2)Чому було обрано саме цю бібліотеку, а не аналоги?

`pygame` є однією з популярних бібліотек для створення ігор на мові програмування Python, Головними причинами чому я обрала саме її для своєї лабораторної роботи є простота використання оскільки `pygame` має досить простий та зрозумілий API, також дозволяє легко реалізувати роботу з графікою, аудіо та іншими елементами ігрового середовища.

3)Наскільки просто та зрозуміло було отримати, встановити, налаштувати та почати використовувати цю бібліотеку?

Встановлення та використання `pygame` було досить простим та зрозумілим. Багата документація, приклади коду та активна спільнота роблять `pygame` доволі доступним

4)Наскільки зрозумілою та корисною була документація бібліотеки?

Документація була досить зрозумілою та корисною в використанні. Документація надає базові приклади, які легко слідувати. Також документація містила інструкції з встановлення, що значно полегшило подальшу роботу. Також багато концепцій, таких як обробка подій, малювання графіки та інші, добре пояснені.

5)Наскільки було зрозуміло, як саме використовувати бібліотеку, які класи/методи/функції використовувати для вирішення поставлених задач?

`pygame` пропонує простий та логічний інтерфейс для створення ігор та графічних додатків. Нижче наведені деякі приклади класів та методів, які були використані,:

- **`pygame.init()`**: Ініціалізує **`pygame`**. Це слід викликати на початку програми.
- **`pygame.display.set_mode(size)`**: Створює вікно гри з заданим розміром.
- **`pygame.draw`**: Модуль для малювання елементів. Методи, такі як **`pygame.draw.rect`**, **`pygame.draw.circle`**, **`pygame.draw.line`** використовуються для малювання графічних об'єктів.

6)Наскільки зручно було використовувати бібліотеку, чи не треба було писати багато надлишкового коду?

Використання **`pygame`** було зручне для написання невеликої гри але для більших та більш складних проектів може знадобитися розгляд інших бібліотек чи фреймворків, залежно від конкретних потреб та вимог.

7)Наскільки зрозумілою була поведінка класів/методів/функцій з бібліотеки?

Загалом, використані класи та методи у **`pygame`** є логічними та зрозумілими, особливо для простих проектів.

8)Наскільки зрозумілою була взаємодія між різними класами/методами/функціями цієї бібліотеки, а також взаємодія між бібліотекою та власним кодом?

Взаємодія між різними класами, методами та функціями в **pygame** зазвичай вважається зрозумілою та логічною, особливо для простих проектів. До того ж, **pygame** спрощує процес створення ігор на Python.

9) Чи виникали якісь проблеми з використанням бібліотеки? Чи вдалось їх вирішити, як саме?

Проблем під час використання даної бібліотеки не виникало, оскільки pygame є досить простою в використанні, а доступна документація ще більше полегшила цей процес.

10) Що хорошого можна сказати про цю бібліотеку, які були позитивні аспекти використання бібліотеки?

Для мене головними перевагами були простота використання, доступна документація та широкі можливості.

11) Що поганого можна сказати про цю бібліотеку, які були негативні аспекти використання бібліотеки?

Також були наявні певні негативні аспекти, які могли б забруднити роботу, наприклад: обмеження на 2D графіку, потенційні проблеми з продуктивністю і т.д.