

УДК 004.415.3

ПОРІВНЯЛЬНИЙ АНАЛІЗ КОД-СТАЙЛ ГАЙДІВ РІЗНИХ МОВ ПРОГРАМУВАННЯ

Соколова А.М.

e-mail: alina.sokolova@nure.ua

Науковий керівник – к.т.н., доц. Руденко Д.О.

Харківський національний університет радіоелектроніки, каф. ІНФ
м. Харків, Україна

Code style guides are essential in software development as they ensure consistency, improve code readability, and facilitate collaboration within teams. This article presents a comparative analysis of coding standards in C#, JavaScript, and Java. The study also highlights the role of clean code principles in maintaining software quality and reducing technical debt. Additionally, automated tools such as StyleCop, ESLint, and Checkstyle help enforce coding standards, making code more maintainable and easier to understand.

У сучасній розробці програмного забезпечення велика увага приділяється чистоті та організованості коду. Код-стайл гайд (code style guide) є набором правил та рекомендацій щодо оформлення коду, що покращує читабельність, полегшує підтримку та допомагає зменшити кількість помилок. Дослідження принципів написання чистого коду в різних мовах програмування дозволяє виявити як загальні тенденції, так і специфічні особливості кожної з них. Пропонується порівняльний аналіз керівництв зі стилю коду для трьох популярних мов програмування: C#, JavaScript і Java.

Чистий код – це код, який легко читати, розуміти та підтримувати. Основні принципи чистого коду включають читабельність, простоту, логічну структурованість і консистентність. Читабельність означає використання зрозумілих імен змінних та функцій, що одразу передають їх призначення. Простота досягається за рахунок мінімізації вкладеності та дублювання коду, що полегшує його розуміння. Логічна структурованість базується на дотриманні принципів SOLID, DRY, KISS і YAGNI, які допомагають писати більш ефективний та гнучкий код. Консистентність означає використання єдиного стилю кодування у всьому проекті, що дозволяє розробникам легко розбиратися у чужому коді.

Кожна мова програмування має свої стандарти оформлення коду. У C# компанія Microsoft рекомендує дотримуватися певних правил. Наприклад, наймінг-конвенції передбачають використання PascalCase для назв класів і методів, а camelCase – для змінних та параметрів. Організація коду базується на принципі «один клас – один файл», що робить код більш структурованим. Для форматування коду використовують 4 пробіли для відступів і уникають зайвих порожніх рядків. Обробка винятків у C# передбачає обов'язкове використання конструкцій try-catch для безпечного виконання програми.

JavaScript має менш суворі правила код-стайлу, проте є загальноприйняті рекомендації. Код організується за принципом модульності, а великі функції розбиваються на менші. Форматування коду може варіюватися, але зазвичай застосовують відступи у 2 або 4 пробіли, залежно від налаштувань проєкту. Обробка помилок у JavaScript часто виконується за допомогою `try-catch` або `promise-based error handling`, що дозволяє ефективно працювати з асинхронними операціями.

Java має більш суворі стандарти, рекомендовані компанією Oracle. Для назв змінних та методів застосовується `camelCase`, а для класів – `PascalCase`. Код у Java суворо дотримується об'єктно-орієнтованої парадигми, що впливає на організацію проєктів. Обробка винятків у Java є обов'язковою для критичних помилок і реалізується через `try-catch` або `throws`.

Дотримання код-стайлу особливо важливе в командній розробці. Узгоджений стиль кодування значно спрощує спільну роботу. Уніфіковані правила оформлення допомагають швидко адаптуватися до нового проєкту та зменшують ризик появи неузгоджених змін у коді. Дотримання стандартів код-стайлу позитивно впливає на продуктивність і підтримку програмного забезпечення. Чистий код спрощує тестування, рефакторинг та виправлення помилок, що зменшує витрати часу та ресурсів на подальше обслуговування проєкту. Для автоматизації перевірки відповідності коду стандартам використовуються спеціальні інструменти, такі як `StyleCop` для `C#`, `ESLint` та `Prettier` для JavaScript, а також `Checkstyle` для Java. Вони допомагають автоматично знаходити порушення стилю, що дозволяє розробникам зосередитися на логіці програми, а не на форматуванні.

Проведений аналіз показує, що принципи чистого коду є універсальними для більшості мов програмування. Основні відмінності між `C#`, JavaScript та Java стосуються форматування та організації коду. `C#` і Java дотримуються строгих стандартів, тоді як JavaScript дозволяє гнучкість у стилі написання. Однак, незалежно від обраної мови, дотримання принципів чистого коду допомагає підвищити якість програмного забезпечення, покращити його читабельність та спростити подальше обслуговування.

Список використаних джерел:

1. .NET Coding Conventions – `C#`. Microsoft Learn: Build skills that open doors in your career. URL: <https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/coding-style/coding-conventions> (date of access: 03.03.2025).

2. JavaScript Style Guide (Airbnb). URL: <https://github.com/airbnb/javascript> (date of access: 03.03.2025).

3. Code Conventions for the Java Programming Language: Contents. URL: <https://www.oracle.com/java/technologies/javase/codeconventions-contents.html> (date of access: 03.03.2025).