Assigment 1 Nurtai aizat

The goal of this project was to implement the **Builder design pattern** in Java while applying the principles of **Clean Code**. Instead of choosing a common example like a Car Builder, I created a **Report Builder**, which constructs a customizable report with fields such as title, author, content, and footer.

This implementation demonstrates how the Builder pattern can simplify the creation of complex objects with many optional parameters, making the code more readable, flexible, and maintainable.

**Several Clean Code principles were applied in this project:**

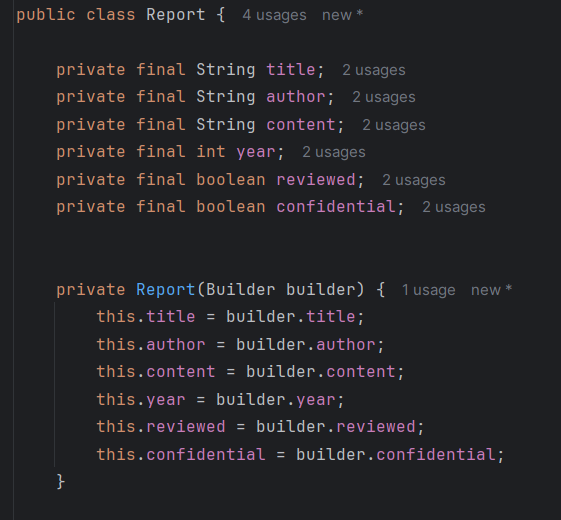
**Meaningful Names**: Class, method, and variable names clearly express their purpose (e.g., ReportBuilder, setTitle, setContent).

**Small Methods**: Each method performs only one action, avoiding unnecessary complexity.

**Fluent Interface**: Method chaining improves readability and makes the object creation process look natural.

**Separation of Concerns**: The Report class is responsible only for representing the data, while the ReportBuilder class handles the object construction process.

**Code**



Изображение выглядит как текст, снимок экрана, Шрифт

Содержимое, созданное искусственным интеллектом, может быть неверным.

Изображение выглядит как текст, снимок экрана

Содержимое, созданное искусственным интеллектом, может быть неверным.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Содержимое, созданное искусственным интеллектом, может быть неверным.

Main

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Содержимое, созданное искусственным интеллектом, может быть неверным.

This project successfully demonstrates how the **Builder design pattern** can be applied in Java using the principles of **Clean Code**. By separating object construction from representation, the solution achieves better readability, maintainability, and clarity.