# Erasyl Meirambekov

# SE-2423

Report: Builder Pattern with Clean Code Principles

In this task, I implemented the Builder pattern using a simple product class called BookingTickets. The goal was to demonstrate the correct use of the pattern together with principles from clean code.

Immutability

All fields of BookingTickets are declared private final, which makes the object immutable once created.

Builder Class

A static nested Builder class is used to construct objects step by step.

Methods return this to enable method chaining.

Clean Code Practices

Naming conventions: Class and method names (BookingTickets, build(), priceInDollars).

Readability: Builder methods use descriptive names (setTitle, setDuration, setPriceInDollars).

toString() Method

Overridden for easier debugging and logging, formatted for clarity.

Conclusion:

This implementation shows how the Builder pattern can simplify object creation, especially when dealing with many optional parameters. Applying *Clean Code* principles improved readability, maintainability, and correctness of the code.