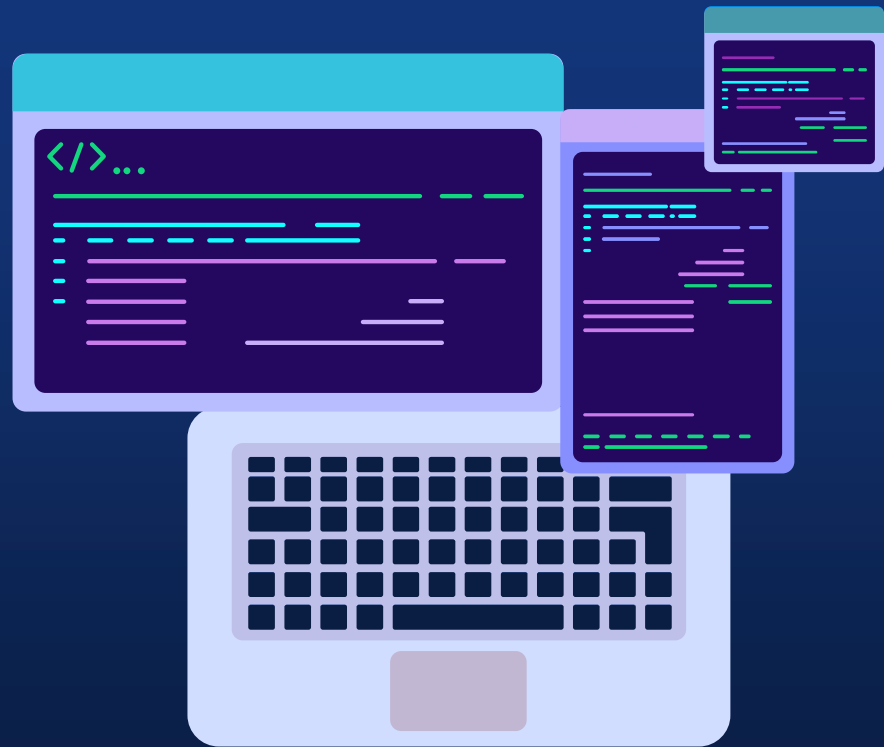


Software Design and Important concepts



Mentor: Einar Rocha

CONTENT



01

OOP Pillars

Inheritance, Polymorphism
Encapsulation, Abstraction



02

Clean Code

Meaningful Names,
Functions, Unit test
Code Smells...

03

SOLID

Single Responsibility
Open closed
Liskov Substitution
Interface Segregation
Dependency Inversion

04

Design patterns

Singleton, Factory Method
Strategy, Observer
Builder...



03

SOLID



Purpose



To create understandable, readable, and testable code that many developers can collaboratively work on.



Agenda

The Single-Responsibility Principle

Responsibility...

The Open/Closed Principle

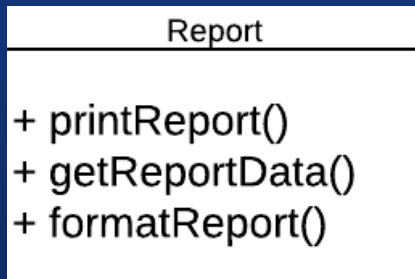
Open for extension
Closed for modification



The Single-Responsibility Principle



A class should have only one reason to change



`printReport()`

Pdf
UI?

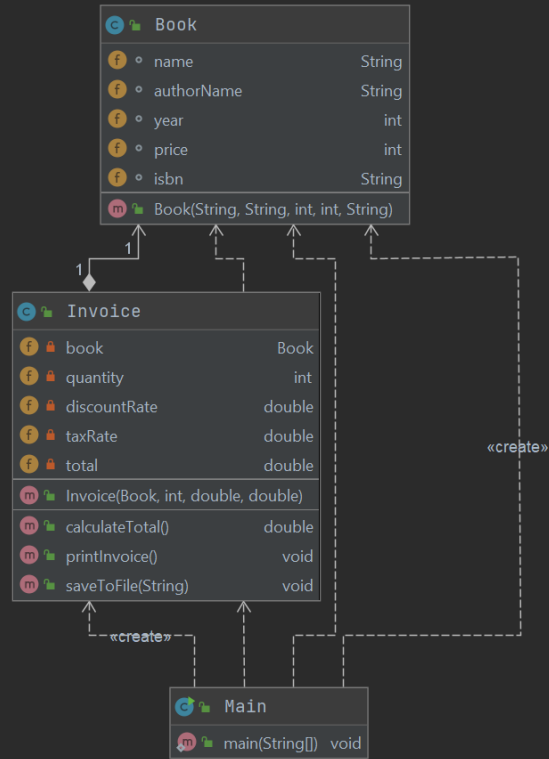
`getReportData()`

XML
JSON
CSV
Database?

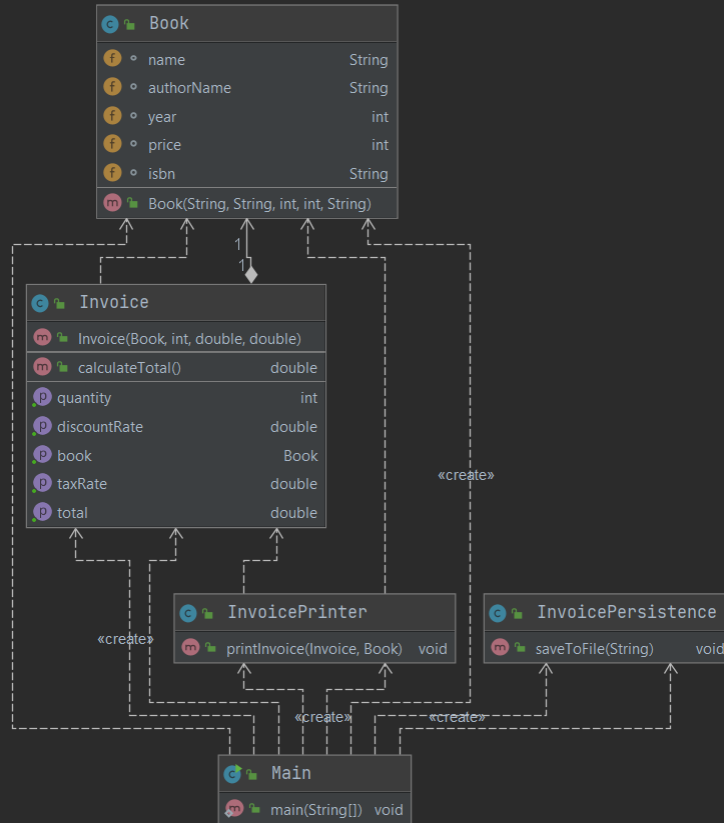
`formatReport()`


UI
HTML
XLS?

Refactor

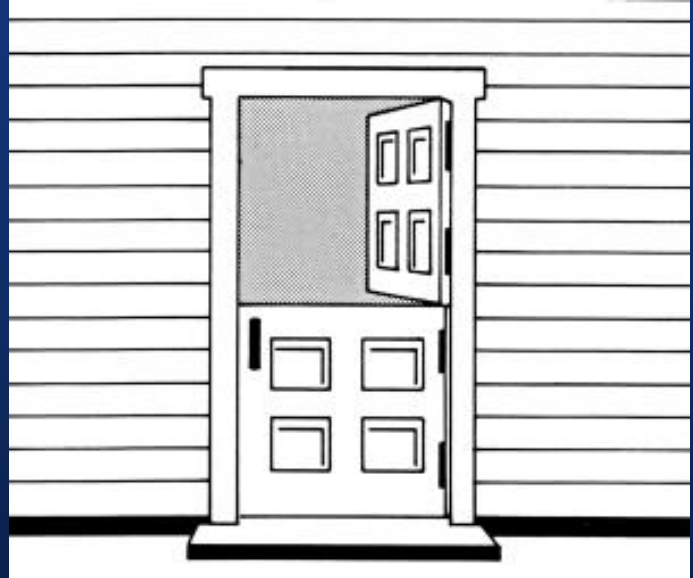


Refactor

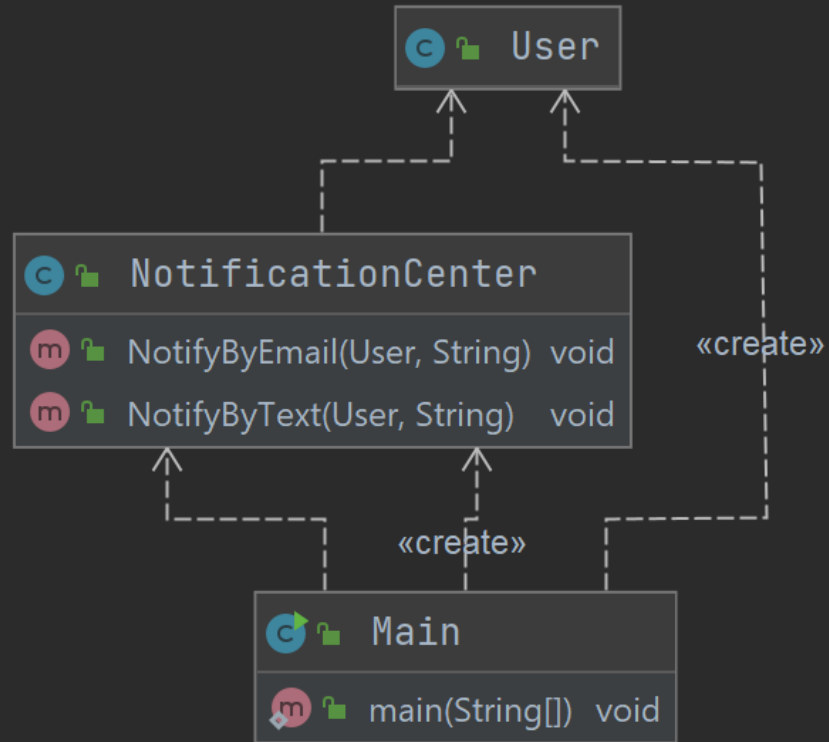




The Open/Closed Principle



Refactor



Refactor

