

Solid principle L-5

Problem occurs when we build Real world application.

→ maintainability

→ Readability.

→ BUGS.

S O L I D.

S :- Single Responsibility principle (SRP)

O :- Open - Close principle (OCP)

L :- Liskov Substitution principle (LSP)

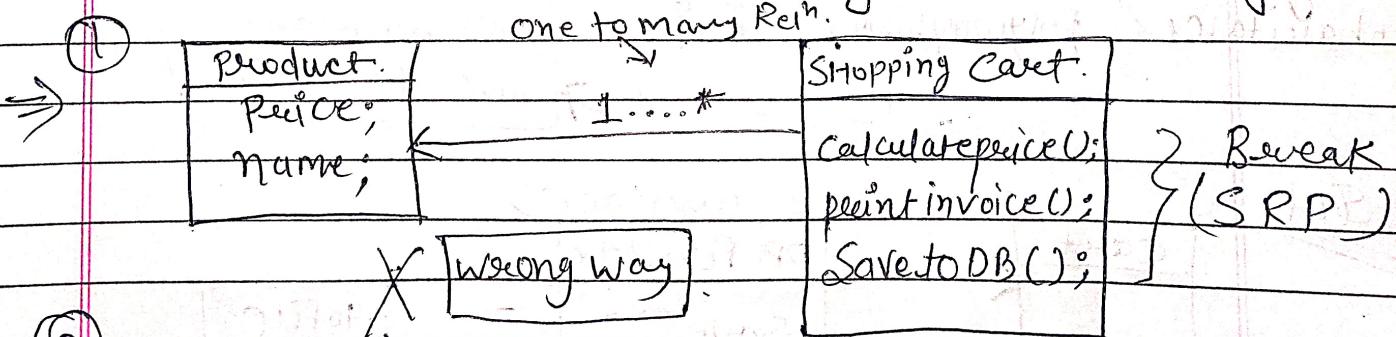
I :- Interface Segregation principle (ISP)

D :- Dependency inversion principle (DIP)

| S :- |

→ A class Should have only one reason to change.

→ A class Should do only one thing.



⇒

Product
Price, Name

ShoppingCart
CalculateTotalPrice()

Invoice Printer
PrintInvoice()

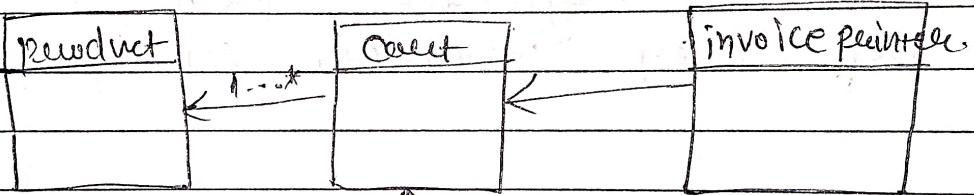
CartToDB Store
ShopInCart() SaveToDB()

11 Code

→ No matter how many methods we use in single class, it matters that all method works on same task.

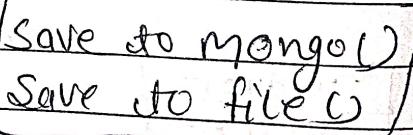
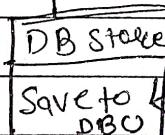
O:- Open-Close Principle.

→ A class should be open for extension but close for modification.



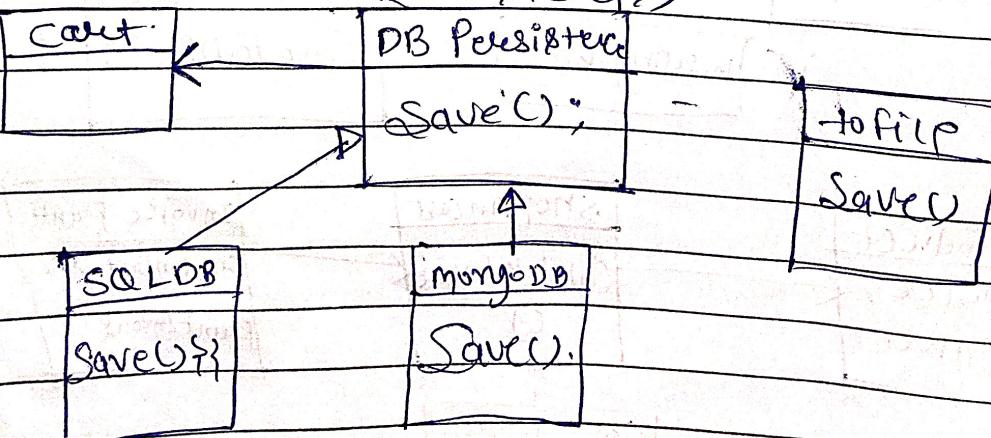
We can apply this through Abstraction, Inheritance & Polymorphism.

Modification is not allowed in class.



Sol.

<Abstract>



L:- Liskov Substitution Principle.

→ Subclasses Should be substitutable for their Base Classes.

