

Programming for engineers II

Master in Sustainable Production Creation

Gustavo Martín Larrea Gallegos
[gustavo.larrea@list.lu]

University of Luxembourg

Master in Sustainable Product Creation

April 1st , 2025



Design patterns

> Design patterns are general, typical and repeatable solutions to commonly occurring software design problems.

Design patterns

- > Design patterns are general, typical and repeatable solutions to commonly occurring software design problems.
- > You can see them as recipes of how to solve a recurring design problem in the code.

Design patterns

- > Design patterns are general, typical and repeatable solutions to commonly occurring software design problems.
- > You can see them as recipes of how to solve a recurring design problem in the code.
- > IMPORTANT: A design pattern is not a finished design that can be copied and pasted into the code. It is a general concept for solving a particular problem.

Design patterns

- > Design patterns are general, typical and repeatable solutions to commonly occurring software design problems.
- > You can see them as recipes of how to solve a recurring design problem in the code.
- > IMPORTANT: A design pattern is not a finished design that can be copied and pasted into the code. It is a general concept for solving a particular problem.
- > The pattern concept should not be copied, but merely referenced.

[\[A great source here\]](#)

Creational patterns

There are distinct creational patterns that provide various creation mechanisms

- > Factory method

- > Builder method

More ...

[\[more about creational patterns\]](#)

Creational patterns

There are distinct creational patterns that provide various creation mechanisms

- > Factory method -> Provides an interface for creating objects in a superclass.
- > Builder method -> Allows to construct object step by step.

More ...

[\[more about creational patterns\]](#)

Structural patterns

These patterns help us with the way how we assemble objects and classes.

> Decorators

> Facade

More ...

[\[more about structural patterns\]](#)

Structural patterns

These patterns help us with the way how we assemble objects and classes.

- > Decorators -> Take a functionality, and add an additional functionality on top
- > Facade -> Programs can be complex, build a facade to show only the necessary elements

More ...

[\[more about structural patterns\]](#)

Behavioral patterns

These patterns are focus on algorithms and approaches to assign responsibilities between objects

- > Iterator pattern

- > Strategy pattern

More ...

[\[more about behavioral patterns\]](#)