



REFACTORING  
· GURU ·

★ Premium Content

⌕ Refactoring

🔗 Design Patterns

→ What is a Pattern

What's a design pattern?

History of patterns

Why should I learn patterns?

Criticism of patterns

Classification of patterns

Catalog

Continued Patterns

★ Shop Now!

🌐 English    ✉ Contact us    👤 Log in

🏠 / [Design Patterns](#)

# What's a design pattern?

**Design patterns** are typical solutions to commonly occurring problems in software design. They are like pre-made blueprints that you can customize to solve a recurring design problem in your code.

You can't just find a pattern and copy it into your program, the way you can with off-the-shelf functions or libraries. The pattern is not a specific piece of code, but a general concept for solving a particular problem. You can follow the pattern details and implement a solution that suits the realities of your own program.

Patterns are often confused with algorithms, because both concepts describe typical solutions to some known problems. While an algorithm always defines a clear set of actions that can achieve some goal, a pattern is a more high-level description of a solution. The code of the same pattern applied to two different programs may be different.

An analogy to an algorithm is a cooking recipe: both have clear steps to achieve a goal. On the other hand, a pattern is more like a blueprint: you can see what the result and its features are, but the exact order of implementation is up to you.

## What does the pattern consist of?

Most patterns are described very formally so people can reproduce them in many contexts. Here are the sections that are usually present in a pattern description:

- **Intent** of the pattern briefly describes both the problem and the solution.
- **Motivation** further explains the problem and the solution the pattern makes possible.
- **Structure** of classes shows each part of the pattern and how they are related.
- **Code example** in one of the popular programming languages makes it easier to grasp the idea behind the pattern.

Some pattern catalogs list other useful details, such as applicability of the pattern, implementation steps and relations with other patterns.

RETURN

READ NEXT

← Design Patterns

History of patterns →

[Home](#)   [Refactoring](#)   [Design Patterns](#)   [Premium Content](#)   [Forum](#)   [Contact us](#)

[f](#)   [✉](#)   [🔄](#)

© 2014-2024 Refactoring.Guru. All rights reserved.  
🖼 Illustrations by Dmitry Zhart

[Terms & Conditions](#)   [Privacy Policy](#)   [Content Usage Policy](#)   [About us](#)

**Ukrainian office:**

🏠 FOP Olga Skobeleva

📍 Abolmasova 7

Kyiv, Ukraine, 02002

✉ Email: [support@refactoring.guru](mailto:support@refactoring.guru)

**Spanish office:**

🏠 Oleksandr Shvets

📍 Avda Pamplona 63, 4b

Pamplona, Spain, 31009

✉ Email: [support@refactoring.guru](mailto:support@refactoring.guru)