During my last two years of Secondary High School and my first year of university, I came across many students and Freelance developers. Having a chat with them, I discovered that most of them didn’t know the meaning of:

So, What is a design pattern?

A design pattern is the re-usable form of a solution to a design problem.A Design Pattern is a description or template for how to solve a problem that can potentially be used in a different situation. An organized set of design pattern relates to the same field is called **pattern language.**

All the Design Pattern can seed up to the development process by providing testes and proved development paradigms. In order to achieve flexibility, most o the design patterns introduce an additional level of [**indirection.**](https://en.wikipedia.org/wiki/Indirection)

The documentation for a design pattern describes the context in which the pattern is used.

Reading the documentation, you will know:

* The Pattern Name
* **Intent:** the description of the goal behind the pattern and why use it.
* Situations in which the pattern is usable.
* Graphical representation of the pattern.
* **Participants:** a list of classes and objects used in the pattern and their roles.
* **Consequences:** description of the results and trade-offs caused by using the pattern.
* A description of the implementation of the pattern.
* **Sample code:** An illustration of how to the pattern can be used.

DESIGN Patterns provide a formal way to solve common programming problems.

As have you seen in your career, most software applications tend to grapple with the same basic problems.

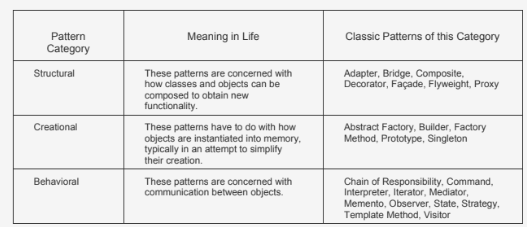
Non-extendable classes:

Most patterns are not tied to a specific language and can be implemented in various frameworks.

Because design pattern is agnostic in nature, patterns are presented using UML diagrams. Optionally, the pattern is illustrated in code using a concise email in a specific programming language (C++, Java etc...).

It is necessary thaty the language used is **not important** as far as a pattern is concerned. Many patterns can also be applied in non-object oriented programming languages such as C.

Categorizing the Classical GOF Design Patterns



source: Intertech-Design-Patterns.pdf (Free online pdf).