

## CHAPTER 1



# Introduction

Over a period of time, software engineers were facing a common problem during the development of various software programs. There were no standards to instruct them how to design and proceed. The issue became significant when a new member (experienced or unexperienced; it does not matter) joined the team and was assigned to do something from scratch or to modify something in the existing product. As already mentioned, since there were no standards, it took a lot of effort to become familiar with the existing product. *Design Patterns* simply addresses this issue and makes a common platform for all developers. *We shall remember that these patterns were intended to be applied in object-oriented designs with the intention of reuse.*

In 1994–95, four Is—Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides—published *Design Patterns—Elements of Reusable Object-Oriented Software* (Addison-Wesley, 1995) in which they initiated the concept of design patterns for software development. These Is became known as the Gang of Four (GoF). They introduced 23 patterns which were developed by experienced software engineers over a very long period of time. As a result, now if any new member joins a development team and he knows that the new system is following some specific design patterns, he can actively participate in the development process with the other members of the team within a very short period of time.

The first concept of real-life design pattern came from the building architect Christopher Alexander. In his experience he came to understand some common problems. Then he tried to address those issues with related solutions (for building design) in a uniform manner. People believe that the software industry grasped those concepts because software engineers can also relate their product applications with these building applications.

*Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice.*

—Christopher Alexander

GoF assures us that though the patterns were described for buildings and towns, the same concepts can be applied for the patterns in object-oriented design. We can substitute the original concepts of walls and doors with objects and interfaces. The common thing in both is that at the core, both types of patterns are solution to problems in some context.

In 1995 the original concepts were discussed with C++. Sun Microsystems released its first public implementation—Java 1.0—in 1995 and then it went through various changes. So, the key point is: Java was relatively new at that time. In this book, we'll try to examine these core concepts with Java. The book is written in Java, but still, if you are familiar with any other popular programming languages (C#, C++ etc.), you can also grasp the concept very easily because I have made as his main focus the design patterns and how we can implement the concept with the basic Java language construct. Again: he has chosen simple, easy-to-remember examples to help you to develop these concepts easily.