Patterns

- A pattern is a common solution to a problem that occurs in many different contexts
- Patterns capture expert knowledge about “best practices” in software design in a form
  - Allows knowledge to be reused
  - Applied in design of many different types of software
- Pattern address the problem of “reinventing the wheel”

History of Patterns

- The use of patterns in software development has its roots in the work of Christopher Alexander, an architect:

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

Design Patterns

- In the late 1980s, several people in the software development community began to apply Alexander's ideas to software
  - *Design Patterns: Elements of Reusable Object-Oriented Software, by Erich Gamma, Richard helm, Ralph Johnson, and John Vlissides (the Gang of Four)*
- Design patterns identify abstractions that are at a higher level than individual classes and objects
  - Construct the software using patterns
    - Singleton Pattern, Proxy Pattern