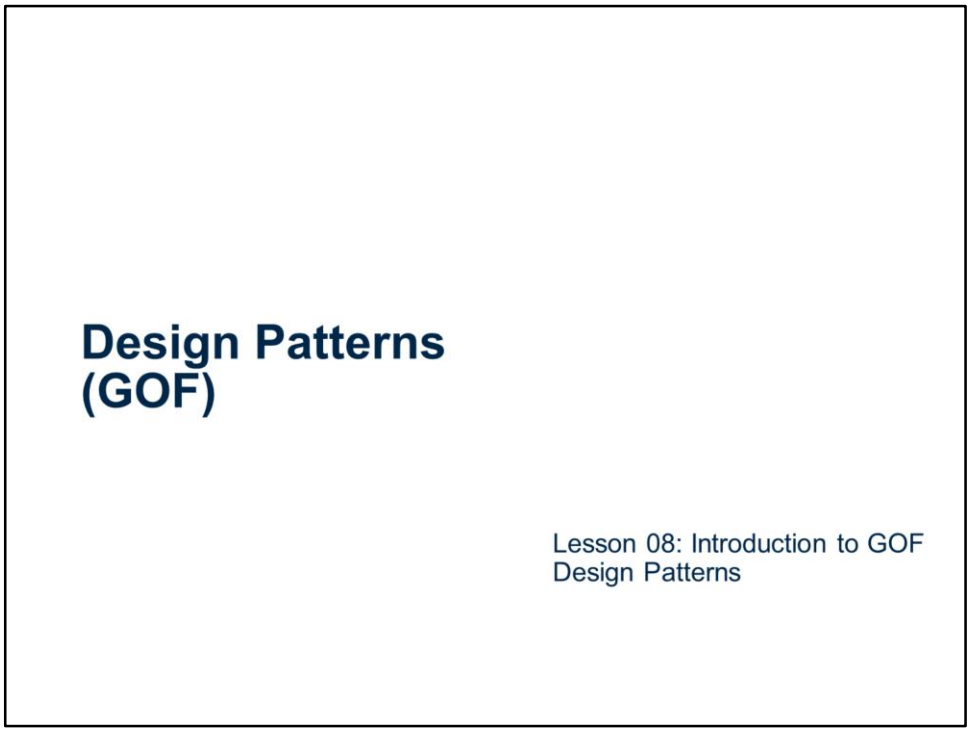
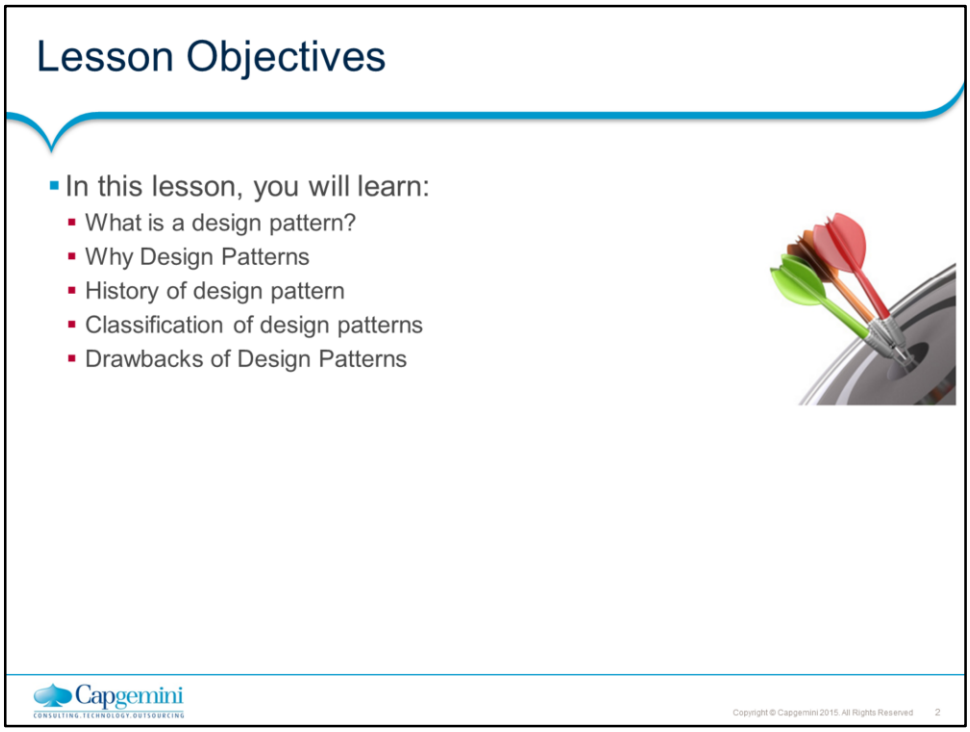
Instructor Notes:



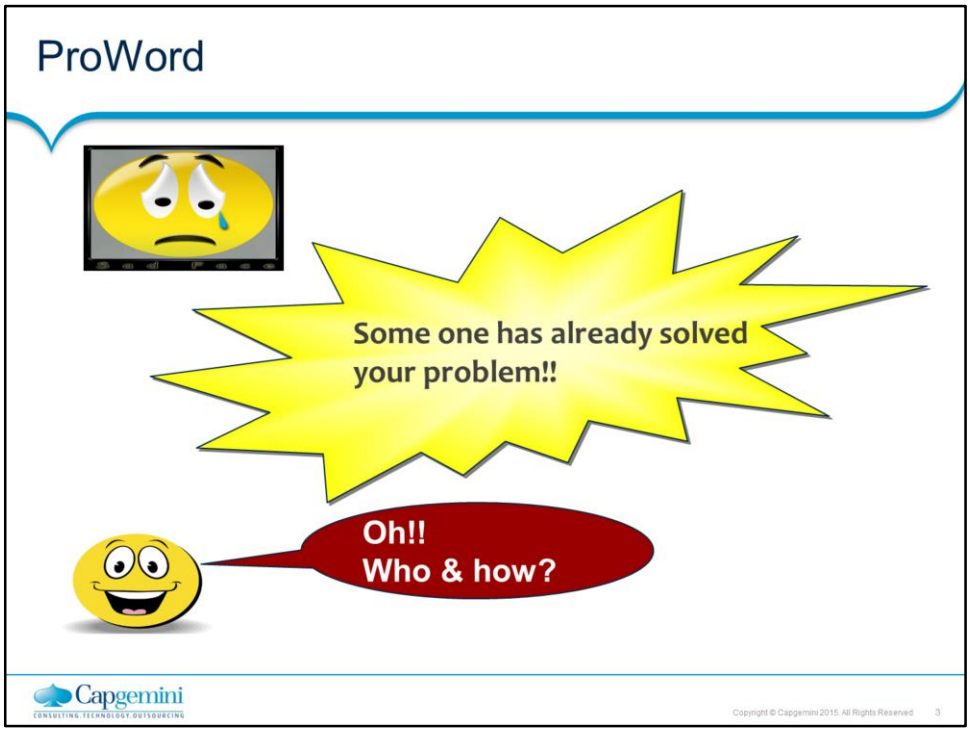
Instructor Notes:

This lesson gives an Introduction on Design Patterns

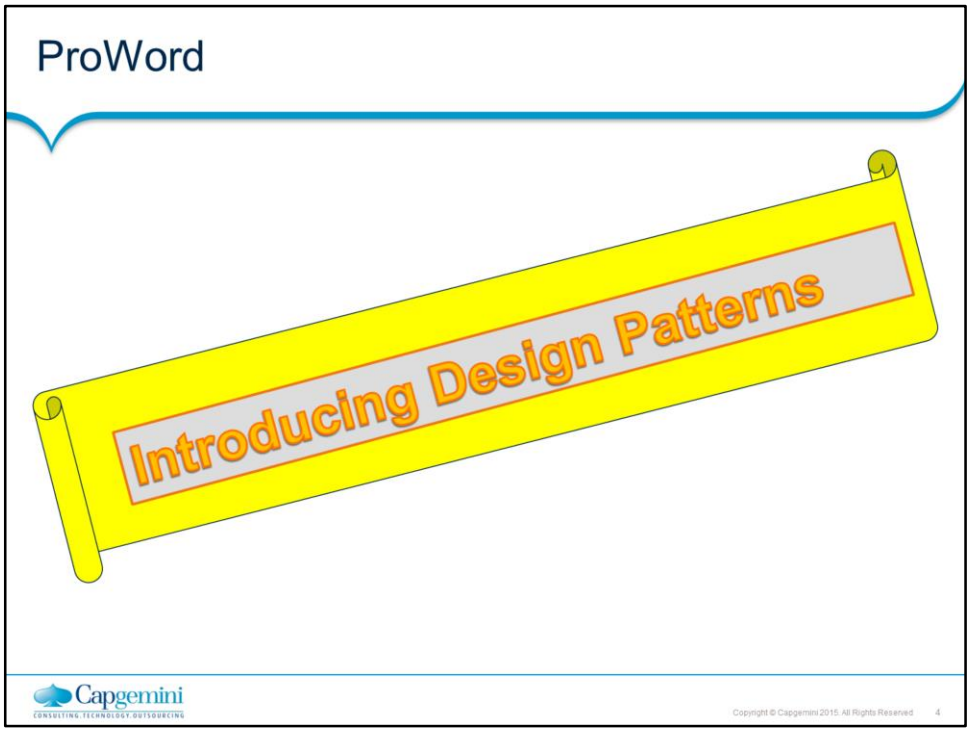


Lesson Objectives:
This lesson introduces GOF (Gang of Four) Design Patterns. The lesson contents are:
Lesson 08: Introduction to GOF Design Patterns
8.1: What is a design pattern?
8.2: Why design patterns?
8.3: History of design patterns
8.4: Classification of Design Patterns
8.5: Drawbacks of Design Patterns

Instructor Notes:

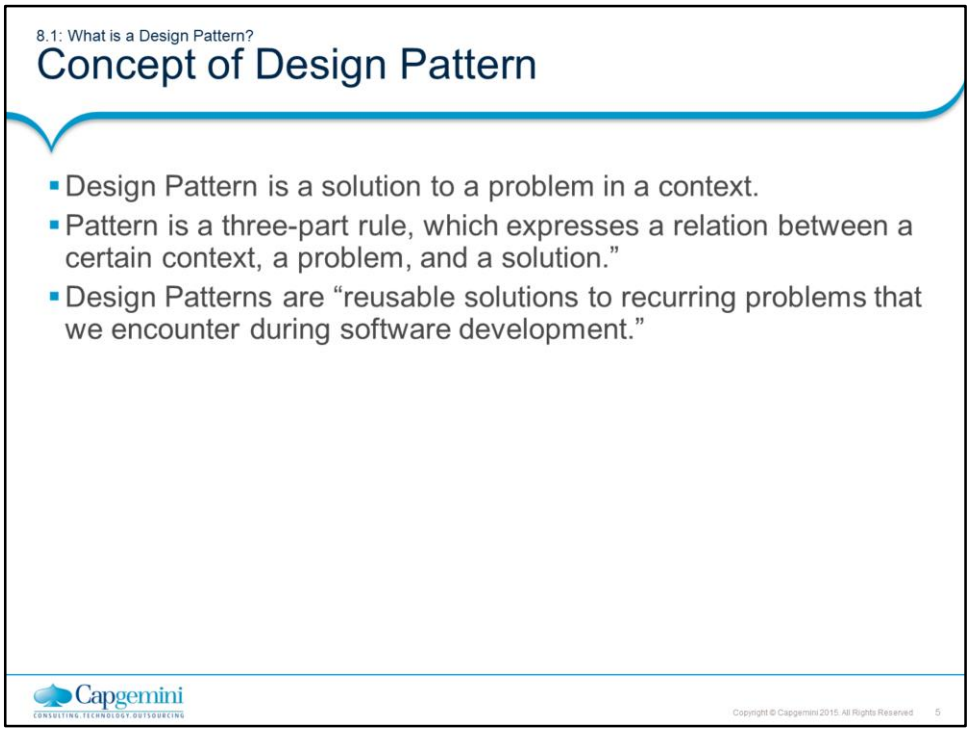


Instructor Notes:



Instructor Notes:

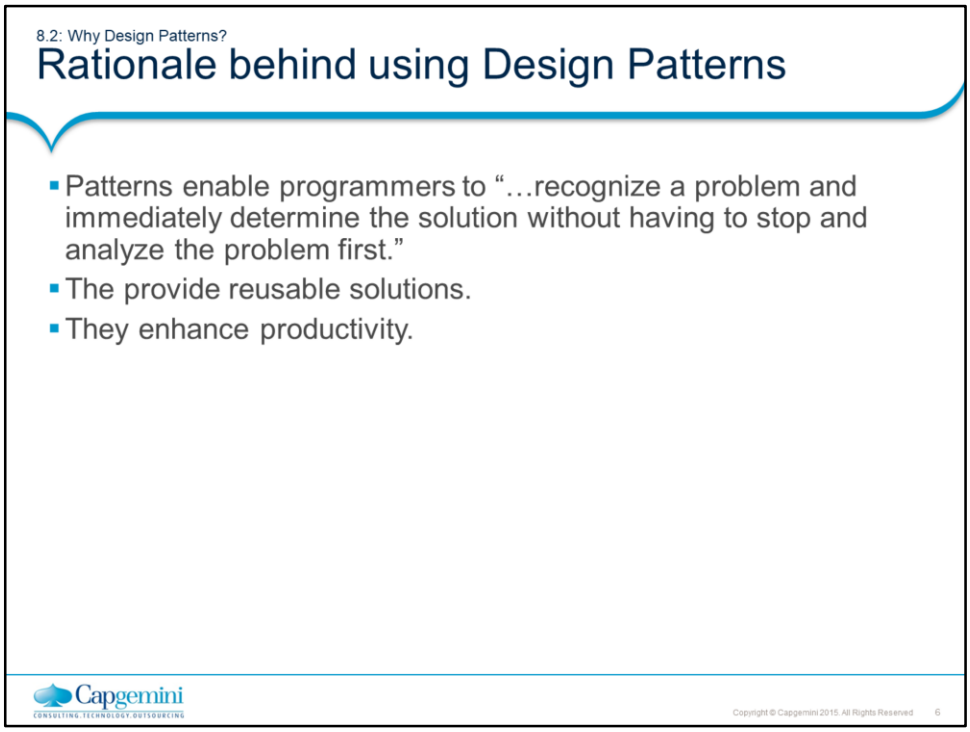
All Design Patterns follow a design principle that says – "Take the parts that vary and encapsulate them, so that later you can alter or extend the parts that vary without affecting those that don't"
All patterns provide a way to let some part of a system vary independently of all other parts.



What is a Design Pattern?
"A 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 using it the same way twice."
Patterns can be applied to many areas of human endeavor, including software development.

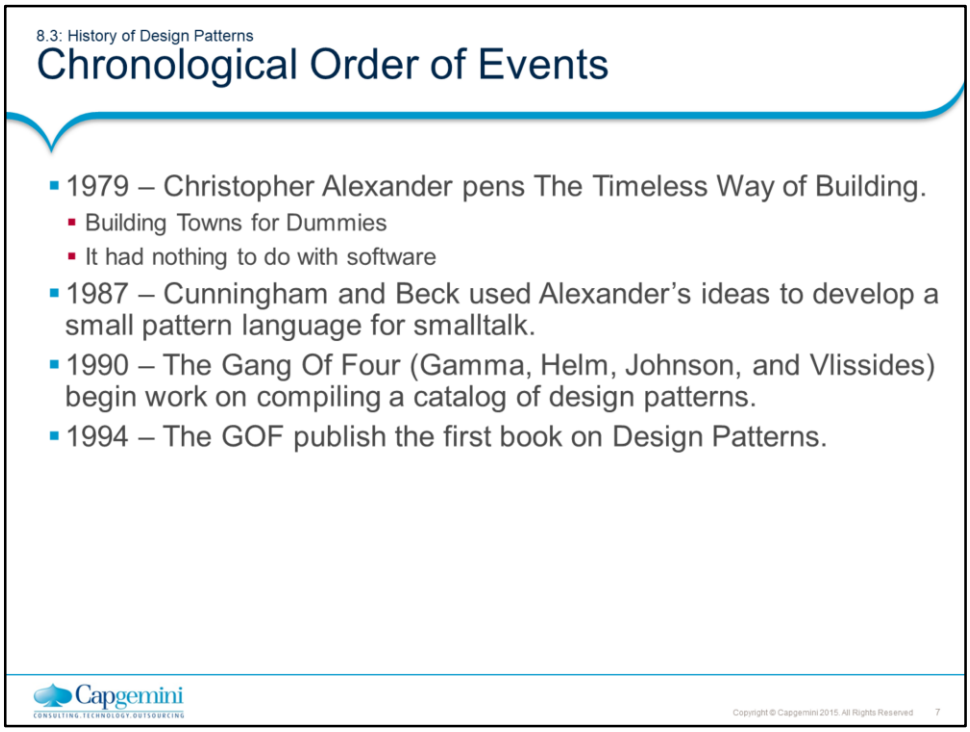
Instructor Notes:

Design Patterns enable large-scale reuse of software architecture and also help document systems.
Patterns explicitly capture expert knowledge and design tradeoffs and make it more widely available.
Patterns help improve a developer communication as they form a common vocabulary.



Why Design Patterns?
Designing object-oriented code is hard, and designing reusable object-oriented software is even harder.
Patterns enable programmers to "…recognize a problem and immediately determine the solution without having to stop and analyze the problem first."
Well structured object-oriented systems have recurring patterns of classes and objects.
The patterns provide a framework for communicating complexities of OO design at a high level of abstraction. Bottom line is productivity.
Experienced designers reuse solutions that have worked in the past.
Knowledge of the patterns that have worked in the past allows a designer to be more productive and the resulting design to be more flexible and reusable.

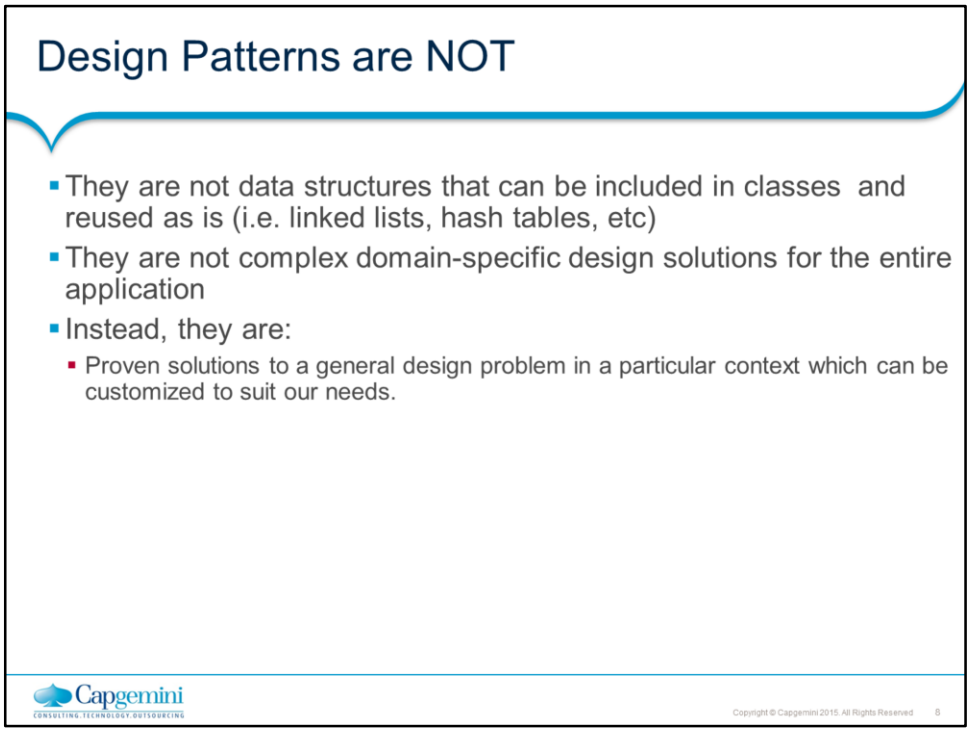
Instructor Notes:



Note:
The history of GOF Design Patterns is listed on the slide.

Instructor Notes:

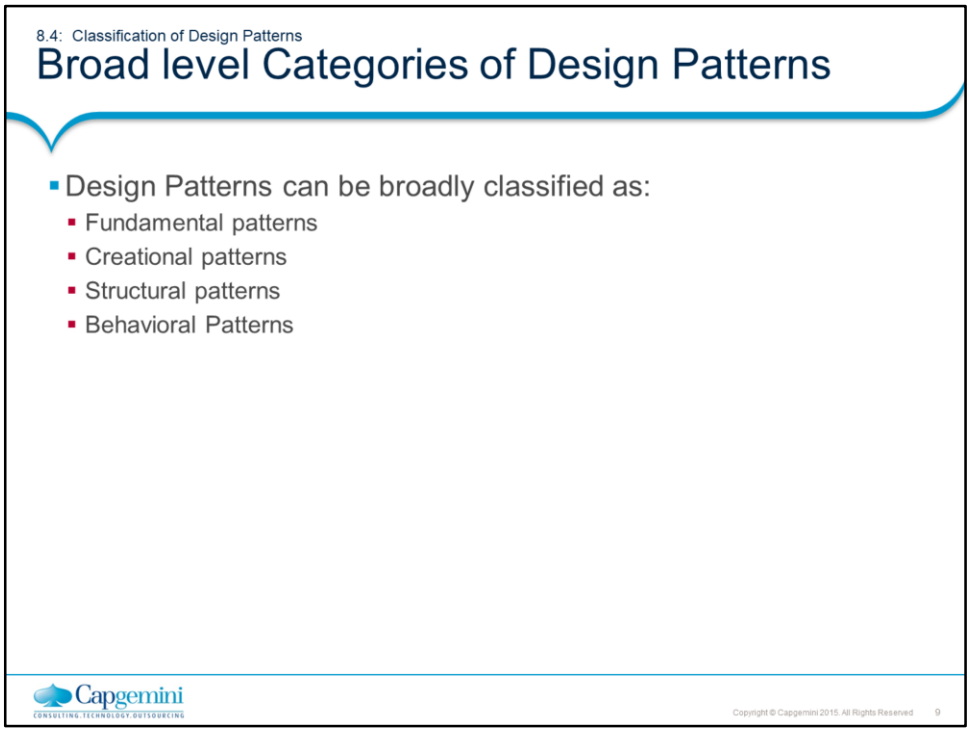
In short, from Design Patterns we do not get “code reuse” rather we get “experience re-use”.



Features of JSP:
Do not ever think that the class structure given by Design Patterns can be used as is unlike Data structures (namely, link lists, hash tables, etc).
They do not provide a domain specific design solution for an entire application.
Instead they provide a proven solution to a general design problem in a particular context (not domain) which can be customized to suit our needs.

Instructor Notes:

Explain the significance of each category of Design Patterns. Mention that the "Fundamental Patterns" are not part of the GOF patterns. They are included here b'cos they form the base/building blocks for the GOF Patterns.

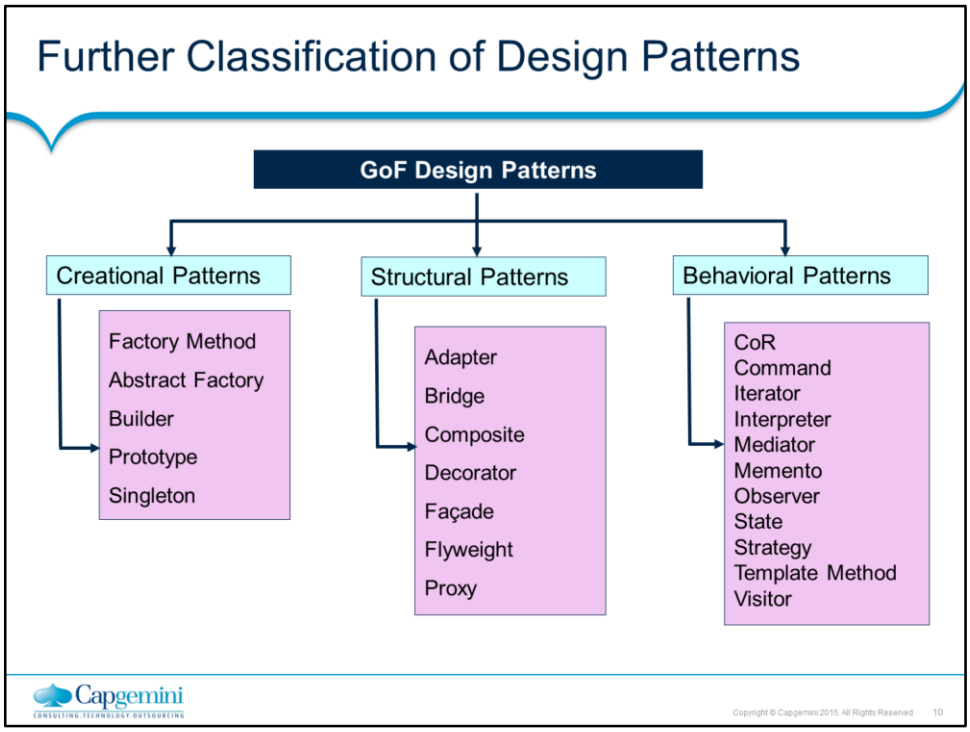


Classification of GOF Design Patterns:
The Design Patterns can be broadly classified as :

Fundamental Patterns: They are the building blocks for the other three categories of Design Patterns.
Creational Patterns: They deal with creation, initializing, and configuring classes and objects.
Structural Patterns: They facilitate decoupling interface and implementation of classes and objects.
Behavioral Patterns: They take care of dynamic interactions among societies of classes and objects. They also give guidelines on how to distribute responsibilities amongst the classes.

Instructor Notes:

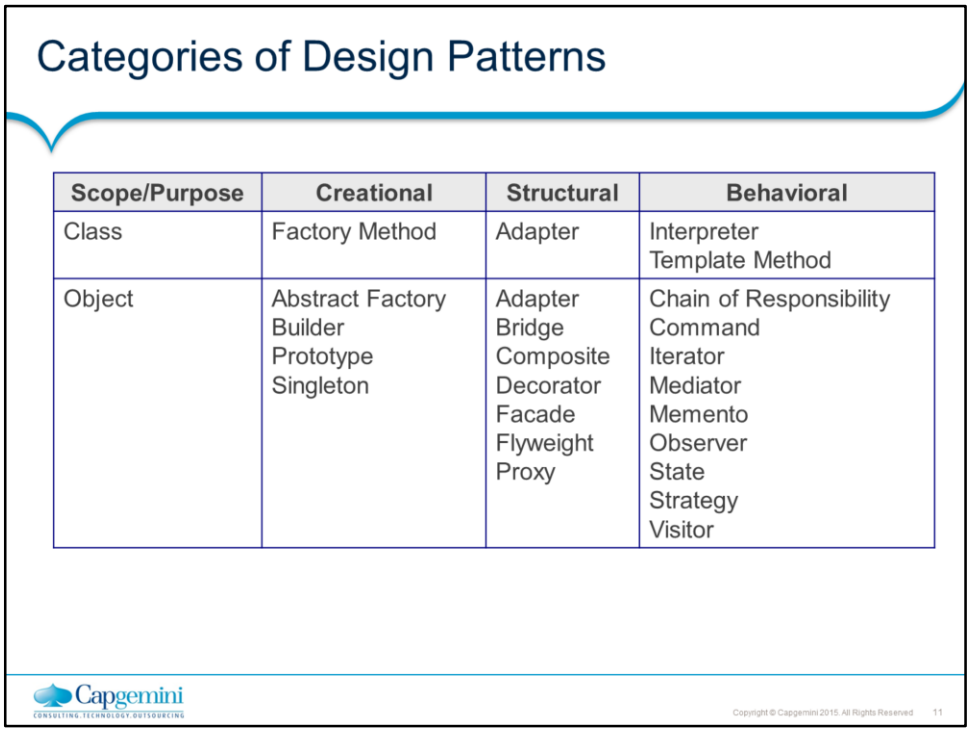
At this stage just mention that there are 23 GOF Design Patterns and the slide lists the GOF patterns as per their classification



Note:
There are 23 GOF Design Patterns.
They have been classified as shown on the slide. Each of the Design Patterns will be explained in detailed in the subsequent lessons.

Instructor Notes:

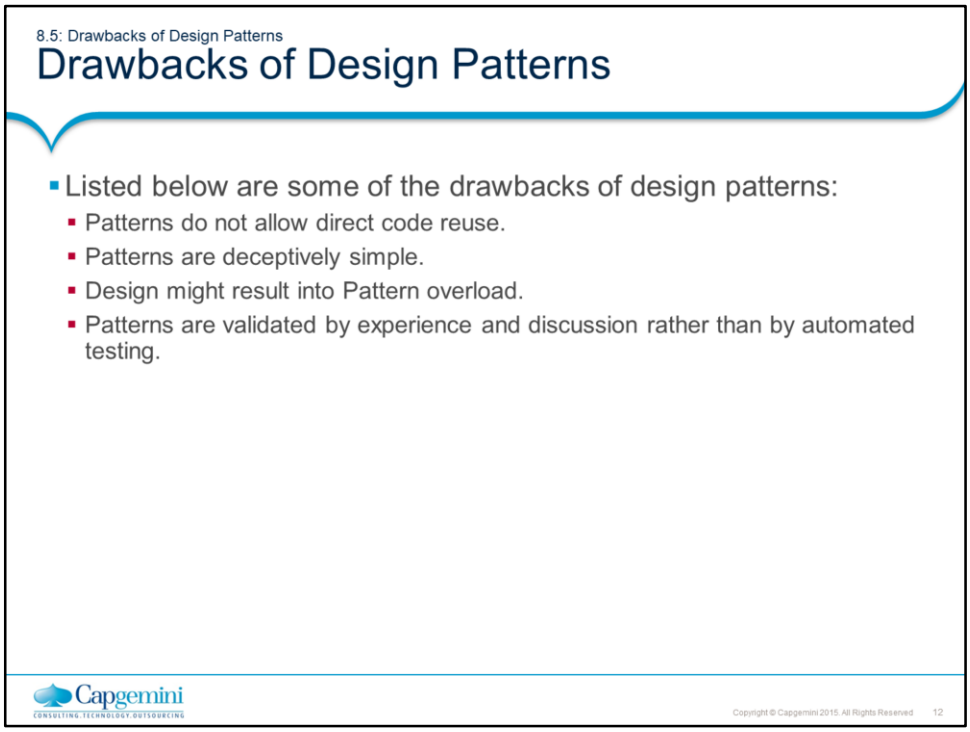
Explain the difference between Class-based & Object-based Design Patterns as explained in notes.



Categories of Design Patterns:
The Design Patterns are categorized as Class based or Object-based.
Class based Design Patterns uses “inheritance” as the basic principle whereas the object based patterns use “composition”.
One of the design principles says, “Favor Composition over Inheritance”.
As seen from the slide, “composition” is being favored by Design Patterns as well as most of the Design Patterns are “Object-based”.

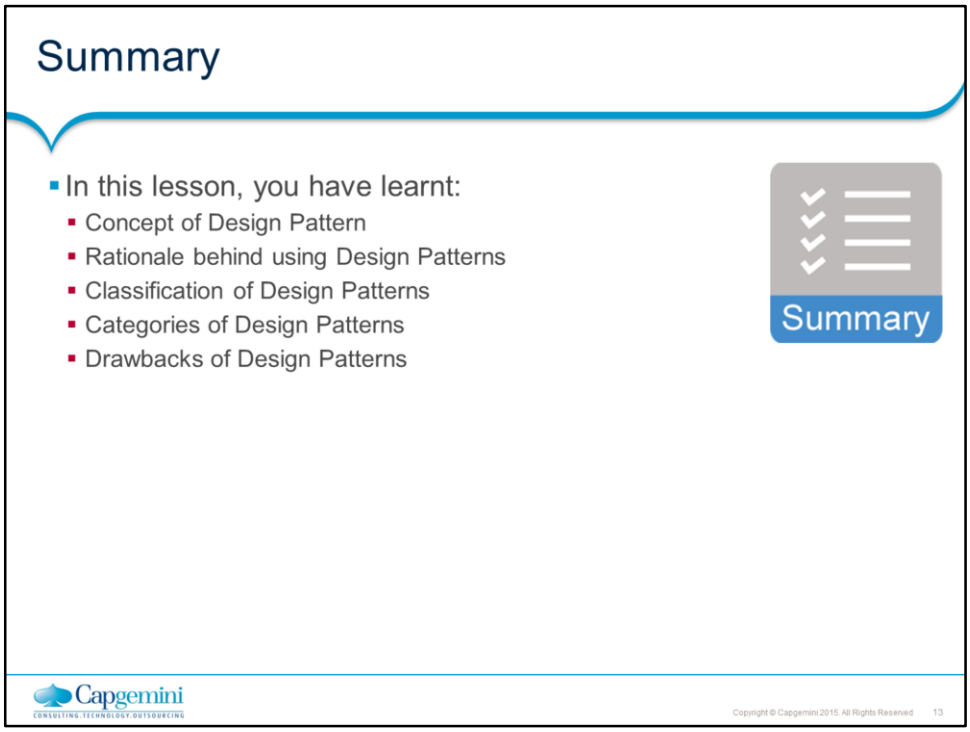
Instructor Notes:

Mention the drawbacks



Note:
Besides the drawbacks mentioned in the slide, Integrating patterns into a software development process is a human-intensive task.

Instructor Notes:



Instructor Notes:

Answers for the Review Questions:

Answer 1: recurring problem; context

Answer 2: Creational, Structural & Behavioral

Answer 3: Class; Object

Review – Questions

- Question 1: Design pattern is a solution to ___ within a particular ___.
- Question 2: Name different types of GOF Design Patterns.
- Question 3: The Design Patterns are categorized with ___ and ___ scope.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 14