



Home ▾

Design Patterns ▾

Servlets ▾

JDBC ▾

Friday, 27 December 2013

## All Design Patterns Links

Tablets,IPad,Mobile Devices or Android OS use Firefox browser to view this blog [Other browsers have some issues to display this blog and playing the embedded youtube video's].So use Firefox browser to View this blog.

- **Design Patterns - Introduction**

- [Design Patterns](#)
- [Design patterns - catalog](#)
- **Enterprise Application Architecture Patterns**
  - [Domain Logic Patterns](#)
  - [Data Source Architectural Patterns](#)
  - [Object Relational Behavioral Patterns](#)
  - [Object-Relational Structural Patterns](#)
  - [Object Relational Metadata Mapping Patterns](#)
  - [Web Presentation Patterns](#)
  - [Distribution Patterns](#)
  - [Offline Concurrency Patterns](#)
  - [Base Design Pattern](#)
  - [Session State Design pattern](#)
  - [Concurrency patterns](#)

- **J2EE patterns**

- [J2EE patterns - Presentation Tier](#)

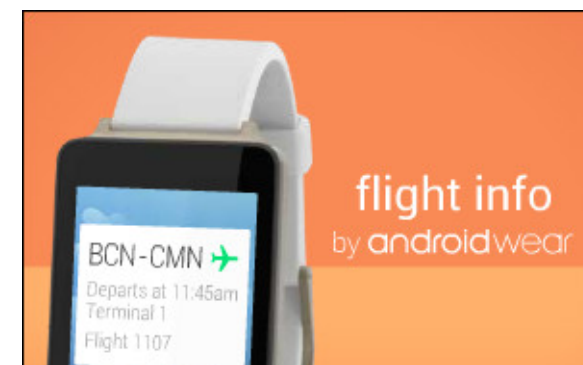
Search  
in this blog .....

Search

Try unlimited music.  
First 30 days on us.

 Get Started

 Google play



- [J2EE patterns - Business Tier](#)
- [J2EE patterns - Integration Tier](#)
- [Creational Design patterns](#)
- [Structural design patterns](#)
- [Behavioral design patterns](#)
- **J2EE Design Patterns**
  - **J2EE patterns**
    - [J2EE patterns - Presentation Tier](#)
    - [J2EE patterns - Business Tier](#)
    - [J2EE patterns - Integration Tier](#)
  - **Front Controller Design Pattern**
    - [Front Controller Design Pattern - Introduction](#)
    - [Front Controller Design Pattern - Class and Sequence Diagram](#)
    - [Front Controller Design Pattern - Implementation](#)
    - [Front Controller Design Pattern - Key Points](#)
  - **Intercepting Filter Design Pattern**
    - [Intercepting Filter Design Pattern - Introduction](#)
    - [Intercepting Filter Design Pattern - Class and Sequence Diagram](#)
    - [Intercepting Filter Design Pattern - Implementation](#)
    - [Intercepting Filter Design Pattern - KeyPoints](#)
  - **Business Delegate Design Pattern**
    - [Business Delegate Design Pattern - Introduction](#)
    - [Business Delegate Design Pattern - Class and Sequence Diagram](#)
    - [Business Delegate Design Pattern - Implementation](#)
    - [Business Delegate Design Pattern - KeyPoints](#)
  - **Service Locator Design Pattern**
    - [Service Locator Design Pattern - Introduction](#)



## Share It



## Blog Archive

- 2014 (216)
- ▼ 2013 (229)
  - ▼ December (44)
    - [Strategy Design pattern - Implementation \[Sort\]](#)
    - [Strategy Design pattern - Implementation \[Search\]](#)
    - [Strategy Design pattern - Implementation \[Payment\]...](#)
    - [Strategy Design pattern - Implementation \[Travel\]](#)
    - [Strategy Design pattern - Real Time Example \[Searc...](#)
    - [Strategy Design pattern - Real Time Example \[Sorti...](#)
    - [Strategy Design pattern - Real Time Example \[Trave...](#)
    - [Strategy Design pattern - Real Time Example \[Payme...](#)
    - [All Design Patterns Links](#)
    - [Mediator Design pattern - Implementation \[Chat Roo...](#)
    - [Mediator Design pattern - Real Time Example \[Chat ...](#)
    - [Mediator Design pattern - Implementation \[Facebook...](#)
    - [Strateav Design pattern - Keypoints](#)

- [Service Locator Design Pattern - Class and Sequence Diagram](#)
- [Service Locator Design Pattern - Implementation](#)
- [Service Locator Design Pattern - KeyPoints](#)
- **Context Object Design Pattern**
  - [Context Object Design Pattern - Introduction](#)
  - [Context Object Design Pattern - Keyoints](#)
- **Data Access Object Design Pattern**
  - [Data Access Object Design Pattern](#)
  - [Data Access Object Design Pattern - KeyPoints](#)
- **Creational Design patterns**
  - [Creational Design patterns](#)
  - **Object Pool Design Pattern**
    - [Object Pool Design Pattern - Introduction](#)
    - [Object Pool Design Pattern - Implementation](#)
    - [Connection Pooling \[Example of Object Pool Design Pattern\]](#)
    - [Object Pool Design Pattern - KeyPoints](#)
  - **Singleton Design Pattern**
    - [Singleton Design Pattern - Introduction](#)
    - [Singleton Design Pattern - Implementation](#)
    - [Singleton Design Pattern - Java Code](#)
    - [Singleton Design Pattern - key Points](#)
  - **Factory Design Pattern**
    - [Factory Design Pattern - Introduction](#)
    - [Factory Design Pattern - Implementation](#)
    - [Factory Design Pattern - key Points](#)
  - **Abstract Factory Design Pattern**
    - [Abstract Factory Design Pattern - Introduction](#)

[Strategy Design pattern - Implementation \[Compress...](#)

[Strategy Design pattern - Sequence Diagram](#)

[Strategy Design pattern - Class Diagram](#)

[Strategy Design pattern - Real Time Example \[Compr...](#)

[Strategy Design pattern - Introduction](#)

[Mediator Design pattern - Real Time Example \[ATC\]](#)

[Mediator Design pattern - Real Time Example \[Faceb...](#)

[Mediator Design pattern - When to Use](#)

[Mediator Design pattern - Introduction](#)

[Decorator Design pattern - Real Time Example \[Car\]...](#)

[Decorator Design pattern - Sequence Diagram](#)

[Decorator Design pattern - Class Diagram](#)

[Mediator Design pattern - Class Diagram](#)

[Decorator Design pattern - Real Time Example \[Pizz...](#)

[Decorator Design pattern - Real Time Example \[Ice ...](#)

[Decorator Design pattern - Real Time Example \[Dosa...](#)

[Mediator Design pattern - Keyoints Down Arrow](#)

[Decorator Design pattern - Keyoints](#)

[Decorator Design pattern - Introduction](#)

[Template Design pattern or](#)

- [Abstract Factory Design Pattern - Implementation](#)
- [Abstract Factory Design Pattern - key Points](#)

- **Prototype Design Pattern**

- [Prototype Design Pattern - Introduction](#)
- [Prototype Design Pattern - Class Diagram](#)
- [Prototype Design Pattern - Implementation](#)
- [Prototype Design Pattern - KeyPoints](#)

- **Builder Design Pattern**

- [Builder Design Pattern - Introduction](#)
- [Builder Design pattern - Real Time Example \[Meal Package\]](#)
- [Builder Design pattern - Real Time Example \[Animal Toys\]](#)
- [Builder Design pattern - Real Time Example \[Beverage\]](#)
- [Builder Design Pattern - Class Diagram](#)
- [Builder Design Pattern - Sequence Diagram](#)
- [Builder Design Pattern - Implementation \[Beverage\]](#)
- [Builder Design Pattern - Implementation \[Animal Toy\]](#)
- [Builder Design Pattern - KeyPoints](#)

- **Behavioral design patterns**

- [Behavioral design patterns](#)

- **Iterator Design Pattern**

- [Iterator Design Pattern - Introduction](#)
- [Iterator Design Pattern - Implementation](#)
- [Iterator Design Pattern - KeyPoints](#)

- **Observer Design Pattern**

- [Observer Design Pattern - Introduction](#)
- [Observer Design Pattern - Real time Example](#)
- [Observer Design Pattern - Class and Sequence Diagram](#)

[Template Method Design ...](#)

[Template Design pattern or  
Template Method Design ...](#)

[Template Design pattern or  
Template Method Design ...](#)

[Template Design pattern or  
Template Method Design ...](#)

[Template Design pattern or  
Template Method Design ...](#)

[Template Design pattern or  
Template Method Design ...](#)

[Template Design pattern or  
Template Method Design ...](#)

[Builder Design pattern -  
Implementation \[Animal To...](#)

[Builder Design pattern - Real Time  
Example \[Bevera...](#)

[Builder Design pattern - Real Time  
Example \[Animal...](#)

[Builder Design pattern -  
Implementation \[Beverage\]...](#)

- November (37)
- October (22)
- September (26)
- August (100)

0



Share 2

YouTube 1K

- [Observer Design Pattern - Implementation](#)

- [Observer Design Pattern - Key Points](#)

- **Chain of Responsibility Design Pattern**

- [Chain of Responsibility Design Pattern - Introduction](#)

- [Chain of Responsibility Design Pattern - Real time Example](#)

- [Chain of Responsibility Design Pattern - Class and Sequence Diagram](#)

- [Chain of Responsibility Design Pattern - Implementation\(One Receiver\)](#)

- [Chain of Responsibility Design Pattern - Implementation\(one or more Receiver\)](#)

- [Chain of Responsibility Design Pattern - Key Points](#)

- **Memento Design pattern**

- [Memento Design pattern - Introduction](#)

- [Memento Design pattern - Real time Example](#)

- [Memento Design pattern - Class Diagram](#)

- [Memento Design pattern - Sequence Diagram](#)

- [Memento Design pattern - Implementation](#)

- [Memento Design pattern - KeyPoints](#)

- **Template Design pattern or Template Method Design pattern**

- [Template Design pattern or Template Method Design pattern - Introduction](#)

- [Template Design pattern or Template Method Design pattern - Real Time Example \[Coffee\]](#)

- [Template Design pattern or Template Method Design pattern - Real Time Example \[Car\]](#)

- [Template Design pattern or Template Method Design pattern - Class Diagram](#)

- [Template Design pattern or Template Method Design pattern - Implementation \[Building a House\]](#)

- [Template Design pattern or Template Method Design pattern - Implementation \[Coffee\]](#)

- [Template Design pattern or Template Method Design pattern - KeyPoints](#)

- **Mediator Design pattern**

- [Mediator Design pattern - Introduction](#)

- [Mediator Design pattern - When to Use](#)



### Google+ Followers

Ram N

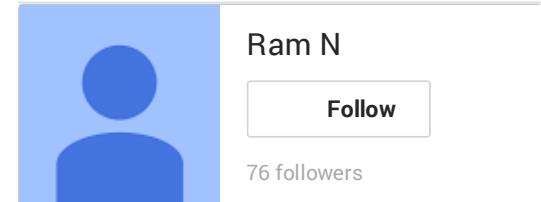
Add to circles



76 have me in circles

[View all](#)

### Google+ Badge



- [Mediator Design pattern - Real Time Example \[Facebook\]](#)
- [Mediator Design pattern - Real Time Example \[ATC\]](#)
- [Mediator Design pattern - Real Time Example \[Chat Room\]](#)
- [Mediator Design pattern - Class Diagram](#)
- [Mediator Design pattern - Implementation \[Facebook\]](#)
- [Mediator Design pattern - Implementation \[Chat Room\]](#)
- [Mediator Design pattern - KeyPoints](#)

#### • Strategy Design pattern

- [Strategy Design pattern - Introduction](#)
- [Strategy Design pattern - Real Time Example \[Compress files\]](#)
- [Strategy Design pattern - Real Time Example \[Payment\]](#)
- [Strategy Design pattern - Real Time Example \[Travel\]](#)
- [Strategy Design pattern - Real Time Example \[Sorting\]](#)
- [Strategy Design pattern - Real Time Example \[Search\]](#)
- [Strategy Design pattern - Class Diagram](#)
- [Strategy Design pattern - Sequence Diagram](#)
- [Strategy Design pattern - Implementation \[Compress files\]](#)
- [Strategy Design pattern - Implementation \[Travel\]](#)
- [Strategy Design pattern - Implementation \[Payment\]](#)
- [Strategy Design pattern - Implementation \[Search\]](#)
- [Strategy Design pattern - Implementation \[Sort\]](#)
- [Strategy Design pattern - KeyPoints](#)

#### • Visitor Design pattern

- [Visitor Design pattern - Introduction](#)
- [Visitor Design pattern - Real time Example \[Kids\]](#)
- [Visitor Design pattern - Real time Example \[Products\]](#)
- [Visitor Design pattern - Class Diagram](#)
- [Visitor Design pattern - Implementation \[Kids\]](#)
- [Visitor Design pattern - Implementation \[Products\]](#)



#### Subscribe To



#### Follow by Email

#### Tutorials

[All Java EE Tutorial Links](#) (3)

[Design Patterns](#) (279)

[Design Patterns Tutorial](#) (1)

[JAVA](#) (1)

[JDBC](#) (54)

[JDBC Tutorial](#) (1)

[Mysql](#) (2)

[Servlets](#) (108)

[Servlets Tutorial](#) (1)

#### Popular Posts



[Prototype Design Pattern - Introduction](#)

[All Design Patterns Links](#)

- [Visitor Design pattern - KeyPoints](#)

- **State Design pattern**

- [State Design pattern - Introduction](#)
- [State Design pattern - Real time example \[ATM\]](#)
- [State Design pattern - Real time example \[TV Remote\]](#)
- [State Design pattern - Real time example \[Vending Machine\]](#)
- [State Design pattern - Real time example \[Project\]](#)
- [State Design pattern - Class Diagram](#)
- [State Design pattern - Implementation \[ATM\]](#)
- [State Design pattern - Implementation \[Vending Machine\]](#)
- [State Design pattern - Implementation \[Project\]](#)
- [State Design pattern - Implementation \[LED TV\]](#)
- [State Design pattern - Keypoints](#)

- **Command Design pattern**

- [Command Design pattern - Introduction](#)
- [Command Design pattern - Real time example \[Hotel\]](#)
- [Command Design pattern - Real time example \[Menu\]](#)
- [Command Design pattern - Class Diagram](#)
- [Command Design pattern - Sequence Diagram](#)
- [Command Design pattern - Object Creation and flow](#)
- [Command Design pattern - Implementation \[Menu\]](#)
- [Command Design pattern - Key points](#)

- **Interpreter Design pattern**

- [Interpreter Design pattern - Introduction](#)
- [Interpreter Design pattern - Real time example \[Google Translator\]](#)
- [Interpreter Design pattern - Real time example \[JVM\]](#)
- [Interpreter Design pattern - Class Diagram](#)
- [Interpreter Design pattern - Implementation \[Date\]](#)



[JDBC : Store Image](#)



[Prototype Design pattern - Class Diagram](#)



[Prototype Design pattern - Implementation](#)



[JDBC Transaction Management](#)



[JDBC Transaction Management Demo](#)

## [Servlets Tutorial](#)



[JDBCRowSet](#)



[JDBC : Retrieve Image](#)

Total Pageviews



115,587

- [Interpreter Design pattern - Key points](#)
- **Structural design patterns**
  - [Structural design patterns](#)
  - **Filter or Criteria Design Pattern**
    - [Filter or Criteria Design Pattern - Introduction](#)
    - [Filter or Criteria Design Pattern - Implementation](#)
    - [Filter or Criteria Design Pattern - Key Points](#)
  - **Composite Design Pattern**
    - [Composite Design Pattern - Introduction](#)
    - [Composite Design Pattern - Class Diagram](#)
    - [Composite Design Pattern - Implementation](#)
    - [Composite Design Pattern - Key Points](#)
  - **Adapter Design Pattern**
    - [Adapter Design pattern - Introduction](#)
    - [Adapter Design Pattern - Real Time Exmaple\(Translator\)](#)
    - [Adapter Design pattern - Real Time Example \(Card Reader\)](#)
    - [Adapter Design pattern - Real Time Example \(Mobile Charger\)](#)
    - [Adapter Design pattern - Real Time Example \(Universal Adaptor\)](#)
    - [Adapter Design pattern - Real Time Example \[JDBC Driver\]](#)
    - [Adapter Design pattern - Class diagram](#)
    - [Adapter Design Pattern - Sequence diagram](#)
    - [Adapter Design Pattern - Implementation \[Language Translator\]](#)
    - [Adapter Pattern - Implementation \[Object Adapter\]](#)
    - [Adapter Pattern - Implementation \[Class Adapter\]](#)
    - [Adapter Design Pattern - Implementation \[Mobile Charger\]](#)
    - [Adapter Design Pattern - Key Points](#)
  - **Proxy Design Pattern**



- [Proxy Design Pattern - Introduction](#)
- [Proxy Design pattern - Real Time Example\[ATM\]](#)
- [Proxy Design pattern - Real Time Example \[Proxy Server\]](#)
- [Proxy Design pattern - Class Diagram](#)
- [Proxy Design pattern - Sequence Diagram](#)
- [Proxy Design pattern - Implementation \[Protection Proxy\]](#)
- [Proxy Design pattern - Implementation \[Remote Proxy\]](#)
- [Proxy Design pattern - Implementation\[Virtual Proxy\]](#)
- [Proxy Design Pattern - Key Points](#)

- **Flyweight Design pattern**

- [Flyweight Design pattern - Introduction](#)
- [Flyweight Design pattern - Real Time Example](#)
- [Flyweight Design pattern - Class Diagram](#)
- [Flyweight Design pattern - Sequence Diagram](#)
- [Flyweight Design pattern - Implementation](#)
- [Flyweight Design pattern - Key Points](#)

- **Decorator Design pattern**

- [Decorator Design pattern - Introduction](#)
- [Decorator Design pattern - Real Time Example \[Dosa\]](#)
- [Decorator Design pattern - Real Time Example \[Ice Cream\]](#)
- [Decorator Design pattern - Real Time Example \[Pizza\]](#)
- [Decorator Design pattern - Real Time Example \[Car\]](#)
- [Decorator Design pattern - Class Diagram](#)
- [Decorator Design pattern - Sequence Diagram](#)
- [Decorator Design pattern - Implementation \[Dosa\]](#)
- [Decorator Design pattern - Implementation \[Pizza\]](#)
- [Decorator Design pattern - Implementation \[Ice Cream\]](#)
- [Decorator Design pattern - Implementation \[Car\]](#)
- [Decorator Design pattern - Implementation \[Shape\]](#)

- [Decorator Design pattern - Key Points](#)
- **Bridge Design pattern**
  - [Bridge Design pattern - Introduction](#)
  - [Bridge Design pattern - Real time example \[Send Message\]](#)
  - [Bridge Design pattern - Real time example \[Publish Message\]](#)
  - [Bridge Design pattern - Real time example \[Shape\]](#)
  - [Bridge Design pattern - Real time example \[TV\]](#)
  - [Bridge Design pattern - Class Diagram](#)
  - [Bridge Design pattern - Implementation \[Shape\]](#)
  - [Bridge Design pattern - Implementation \[Send Message\]](#)
  - [Bridge Design pattern - Implementation \[Publish Message\]](#)
  - [Bridge Design pattern - Implementation \[LED TV\]](#)
  - [Bridge Design pattern - Keypoints](#)
- **Concurrency Design patterns**
  - [Concurrency patterns](#)
  - [Thread Pool Design Pattern](#)
  - [Monitor or synchronization Design Pattern](#)
  - [Lock Design Pattern](#)
  - [Scheduler Design Pattern](#)
  - [Read-write lock Design Pattern](#)
  - **Messaging Design Pattern(MDP)**
    - [Messaging Design Pattern\(MDP\) - Introduction](#)
    - [Messaging Design Pattern\(MDP\) - Asynchronous messaging](#)
    - [Messaging Design Pattern\(MDP\) - Two Way Messaging](#)
    - [Messaging Design Pattern\(MDP\) - Implementation of Proxy](#)
    - [Messaging Design Pattern\(MDP\)-Implementation of Adapter](#)
    - [Messaging Design Pattern\(MDP\) - Implementation of Webservice](#)

- **Reactor Design Pattern**
  - [Reactor Design Pattern - Introduction](#)
  - [Reactor Design Pattern - Implementation](#)
  - [Reactor Design Pattern vs Observer Design Pattern](#)
- **Double-checked locking Design Pattern**
  - [Double-checked locking Design Pattern - Introduction](#)
  - [Double-checked locking Design Pattern - Implementation](#)
  - [Double-checked locking Design Pattern-Out-of-order writes](#)
- **Offline Concurrency Patterns**
  - [Offline Concurrency Patterns](#)
  - [Optimistic Offline Lock Design Pattern](#)
  - [Pessimistic Offline Lock Design Pattern](#)
  - [Coarse-Grained Lock Design Pattern](#)
  - [Implicit Lock Design Pattern](#)
- **Session State Design pattern**
  - [Session State Design pattern](#)
  - [Client Session State Design Pattern](#)
  - [Server Session State Design Pattern](#)
  - [Database Session State Design Pattern](#)
- **Distribution Patterns**
  - [Distribution Patterns](#)
  - [Data Transfer Object Design Pattern](#)
  - [Remote Facade Design Pattern](#)
- **Base Design Pattern**
  - [Base Design Pattern](#)
  - [Gateway Design Pattern](#)
  - [Layer Supertype Design Pattern](#)

- [Registry Design Pattern](#)
- [Money Design Pattern](#)
- [Plugin Design Pattern](#)
- [Special Case Design Pattern](#)
- [Separated Interface Design Pattern](#)
- [Service Stub Design Pattern](#)
- [Record Set Design Pattern](#)
- [Mapper Design Pattern](#)
- [Value Object Design Pattern](#)

- **Web Presentation Patterns**

- [Web Presentation Patterns](#)
- [Model View Controller Design Pattern](#)
- [Application Controller Design Pattern](#)
- [Page Controller Design Pattern](#)
- [Template View Design Pattern](#)
- [Transform View Design Pattern](#)
- [Two Step View Design Pattern](#)

- **Front Controller Design Pattern**

- [Front Controller Design Pattern - Introduction](#)
- [Front Controller Design Pattern - Class and Sequence Diagram](#)
- [Front Controller Design Pattern - Implementation](#)
- [Front Controller Design Pattern - Key Points](#)

- **Domain Logic Patterns**

- [Domain Logic Patterns](#)
- [Transaction Script Design Pattern](#)
- **Domain Model Design Pattern**
  - [Domain Model Design Pattern](#)
  - [Domain Model Design Pattern - Samples](#)

- [Table Module Design Pattern](#)
- [Table Module Design Pattern vs Transaction Script Pattern](#)
- [Service Layer Design Pattern](#)
- **Object Relational Structural Patterns**
  - [Object-Relational Structural Patterns](#)
  - [Identity Field Design Pattern](#)
  - [Foreign Key Mapping Design Pattern](#)
  - [Association Table Mapping Design Pattern](#)
  - [Dependent Mapping Design Pattern](#)
  - [Embedded Value Design Pattern](#)
  - [Serialized LOB Design Pattern](#)
  - [Single Table Inheritance Design Pattern](#)
  - [Class Table Inheritance Design Pattern](#)
  - [Concrete Table Inheritance Design Pattern](#)
  - [Inheritance Mappers Design Pattern](#)
- **Object Relational Behavioral Patterns**
  - [Object-Relational Behavioral Patterns](#)
  - [Identity Map Design Pattern](#)
  - **Unit of Work Design Pattern**
    - [Unit of Work Design Pattern Introduction](#)
    - [Unit of Work Design Pattern Implementation](#)
  - **Lazy Load Design Pattern**
    - [Lazy Load Design Pattern - Introduction](#)
    - [Lazy Load Design Pattern - Implementation](#)
- **Object Relational MetadataMapping Patterns**
  - [Object-Relational Metadata Mapping Patterns](#)
  - [Metadata Mapping Design Pattern](#)

- [Query Object Design Pattern](#)
- [Repository Design Pattern](#)
- **Data Source Architectural Patterns**
  - [Data Source Architectural Patterns](#)
  - [Table Data Gateway Design Pattern](#)
  - [Row Data Gateway Design Pattern](#)
  - [Active Record Design Pattern](#)
  - [Data Mapper Design Pattern](#)
- [More Design Patterns ...](#)

Posted by [Ram N](#) at [22:23](#)  +2 Recommend this on Google

Labels: [All Java EE Tutorial Links](#), [Design Patterns](#), [Design Patterns Tutorial](#)

## 6 comments:



**[NGSHadoop](#)** [12 July 2014 22:19](#)

Hi Ram, Thanks for the nice blog on design pattern. Its very helful. Where can I download ppt and java code for all design pattern?

[Reply](#)



**[Ram N](#)**  [13 July 2014 04:33](#)

Thanks for the feedback :). I didn't upload the code and ppt. For each design pattern I have provided the implementation [e.g. <http://ramj2ee.blogspot.in/2013/11/proxy-design-pattern-implementation.html>] you can see the sample code and you can download the images which is there is each page and make a ppt based on your need.

[Reply](#)

▼ [Replies](#)



**[NGSHadoop](#)** [13 July 2014 06:11](#)

Thanks Ram :) ..Where is Facade pattern video?



**Ram N**  13 July 2014 21:18

Here is the link..

<http://ramj2ee.blogspot.in/2013/08/remote-facade-design-pattern.html>

---

[Reply](#)



**Gaurav** 23 July 2014 10:49

Nice Blog Ram. I just glanced through some videos and sure will return to your blog :)

[Reply](#)

▼ [Replies](#)



**Ram N**  24 July 2014 03:16

Thanks for the feedback :)

---

[Reply](#)

Enter your comment...

Comment as:

Select profile...

Publish

Preview



[Newer Post](#)

[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)

Template images by [PLAINVIEW](#). Powered by [Blogger](#).