Facade Pattern Tutorial
Written Date : October 14, 2009

This tutorial is aimed to guide the definition and application of Gang of Four (GoF) facade design pattern. By reading this tutorial, you will know how to develop a model for the facade pattern, and how to apply it in practice.

Modeling Design Pattern with Class Diagram
1. Create a new project Design Patterns.
2. Create a class diagram Facade.
3. Select **Package** from diagram toolbar. Press on the diagram and drag it towards bottom right to form a package representing a subsystem.

4. Name the package **subsystem**.
5. Select **Class** from diagram toolbar. Click inside subsystem to create a class. Name it as *Facade*.

![Diagram](image1)

6. Right-click on *Facade* and select **Add > Operation** from the popup menu.

![Diagram](image2)

7. Name the operation *ClientRequest*. Note that it must be a public operation that enables classes external to the subsystem to access it.

![Diagram](image3)

**Defining Pattern**
1. Select everything on the class diagram.

2. Right-click on the Singleton class and select **Define Design Pattern...** from the popup menu.
3. In the **Define Design Pattern** dialog box, specify the pattern name *Facade*. Keep the file name as is. Click **OK** to proceed.

---

**Applying Design Pattern on Class Diagram**

In this section, we are going to apply the facade pattern in modeling a code generator.

1. Create a new project *Code Generator*.
2. Create a class diagram *Generator*.
3. Right-click on the class diagram and select **Utilities > Apply Design Pattern...** from the popup menu.
4. In the **Design Pattern** dialog box, select **Facade** from the list of patterns.

5. Select **subsystem** in the overview pane.
6. At the bottom pane, rename subsystem to generator. Rename Facade to CodeGenerator and ClientRequest to generate.

7. Click OK to apply the pattern. This is the diagram obtained:

8. We need to fill in the subsystem. Move the mouse cursor over the CodeGenerator class, and drag out Aggregation > Class to empty region in the package to create a class. Name the class ClassBuilder.
9. Right-click on ClassBuilder and select Open Specification from the popup menu.

![Image of ClassBuilder with Open Specification option]

10. In the Class Specification, set Visibility to be package. Click OK to confirm.

![Image of Class Specification window with Visibility set to package]

11. Repeat steps 8 to 10 to create classes AttributeBuilder and OperationBuilder.

![Diagram showing relationships between ClassBuilder, AttributeBuilder, and OperationBuilder]

Resources
1. Design Patterns.vpp
2. Facade.pat
Related Links

- Full set of UML tools and UML diagrams

Visual Paradigm home page
(https://www.visual-paradigm.com/)

Visual Paradigm tutorials
(https://www.visual-paradigm.com/tutorials/)