Composite Pattern Tutorial

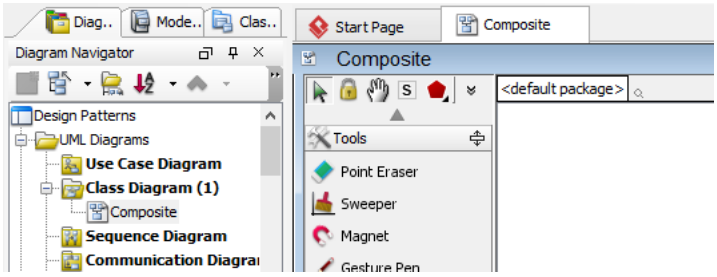
Written Date : October 7, 2009

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

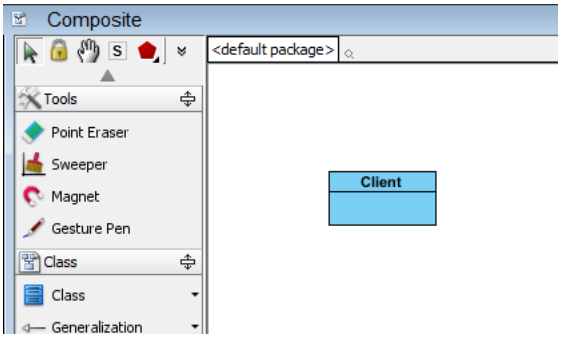
Modeling Design Pattern with Class Diagram

1. Create a new project *Design Patterns*.

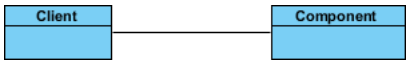
2. Create a class diagram *Composite*.



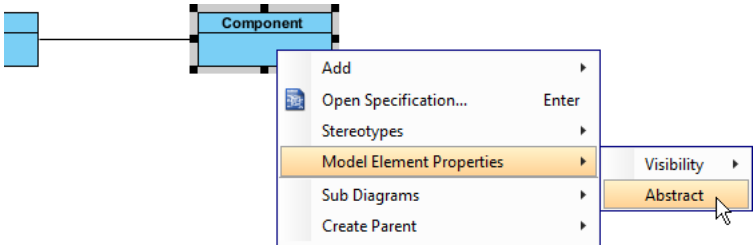
3. Select **Class** from diagram toolbar. Click on the diagram to create a class. Name it as *Client*.



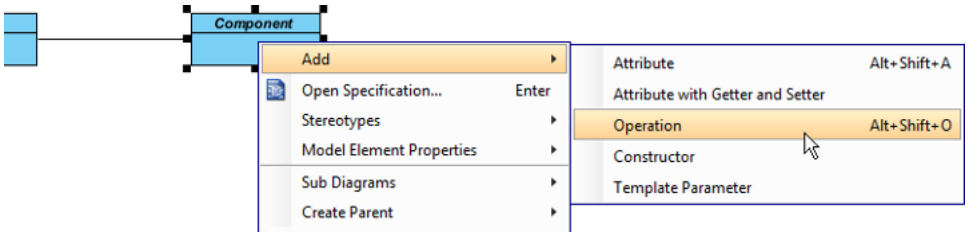
4. Move the mouse cursor over the *Client* class, and drag out **Association** > **Class** to create an associated class *Component*.



5. Right click on *Component*, and select **Model Element Properties** > **Abstract** to set it as abstract.

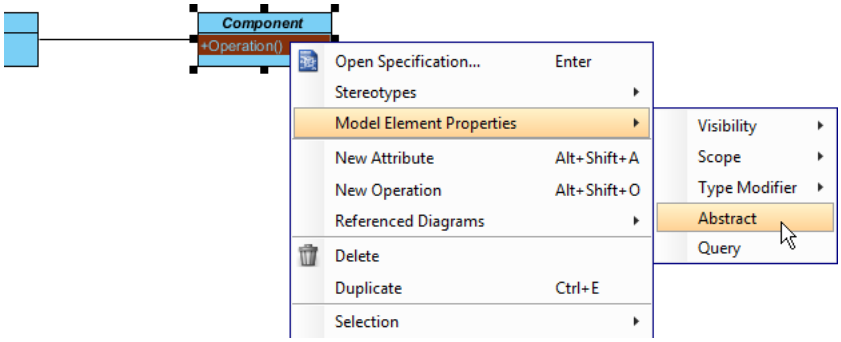


6. Right click on the *Component* class, and select **Add** > **Operation** from the popup menu.

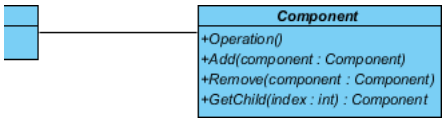


7. Name the operation *Operation()*.

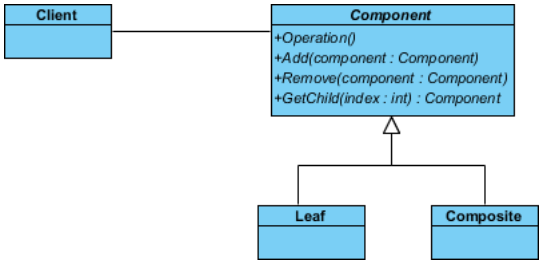
8. Right click on *Operation*, and select **Model Element Properties** > **Abstract** to set it as abstract.



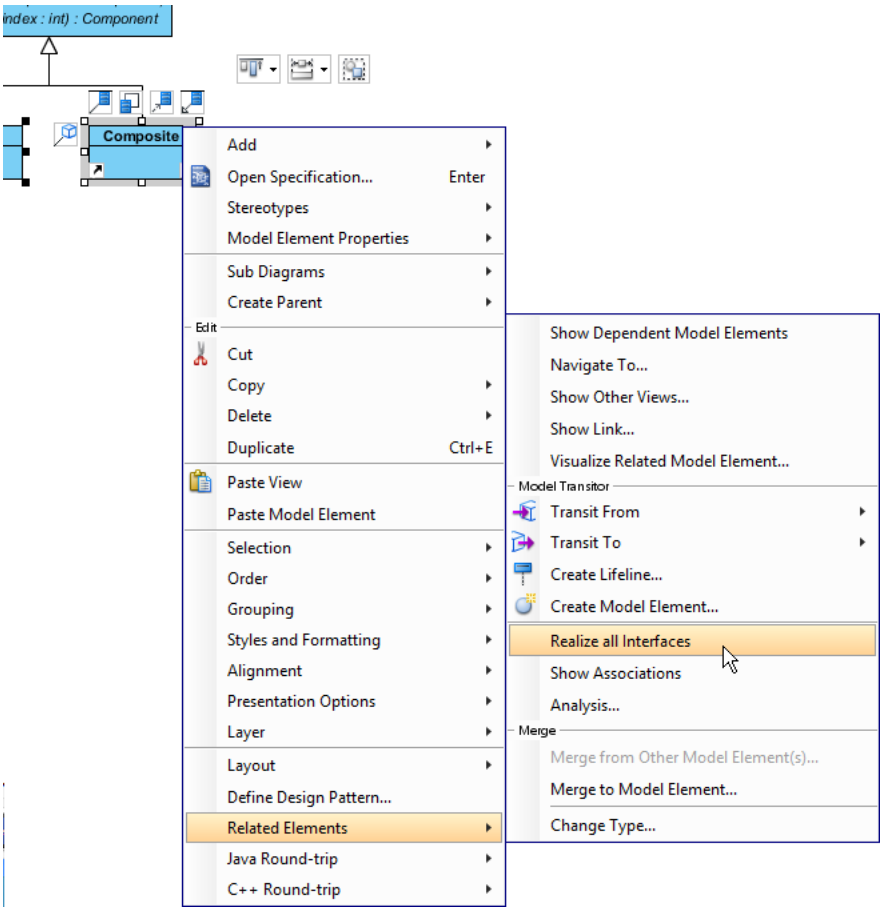
9. Repeat steps 6 to 8 to create operations *Add(component : Component)*, *Remove(component : Component)*, *GetChild(index : int) : Component*.



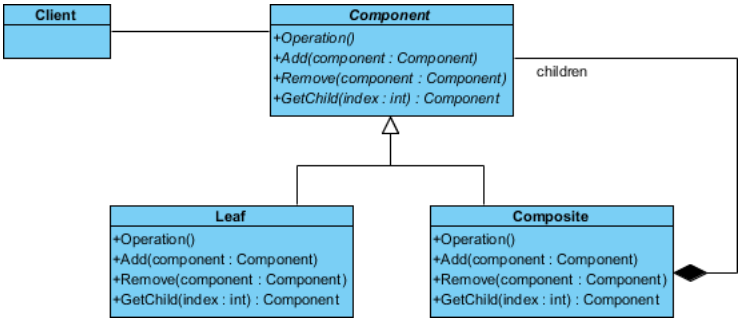
10. Move the mouse cursor over the *Component* class, and drag out **Generalization** > **Class** to create a subclass *Leaf*. Repeat this step to create another subclass *Composite*, from *Component*.



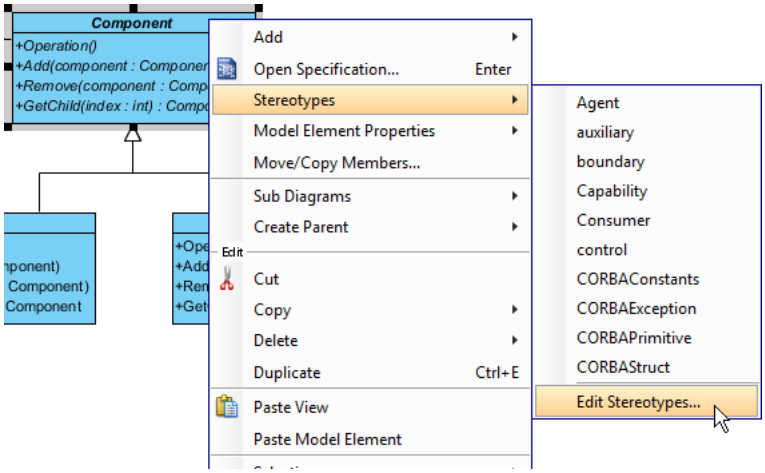
11. *Leaf* and *Composite* will inherit the operations from *Component*. Select *Leaf* and *Component*, right click on them and select **Related Elements** > **Realize all Interfaces** from the popup menu.



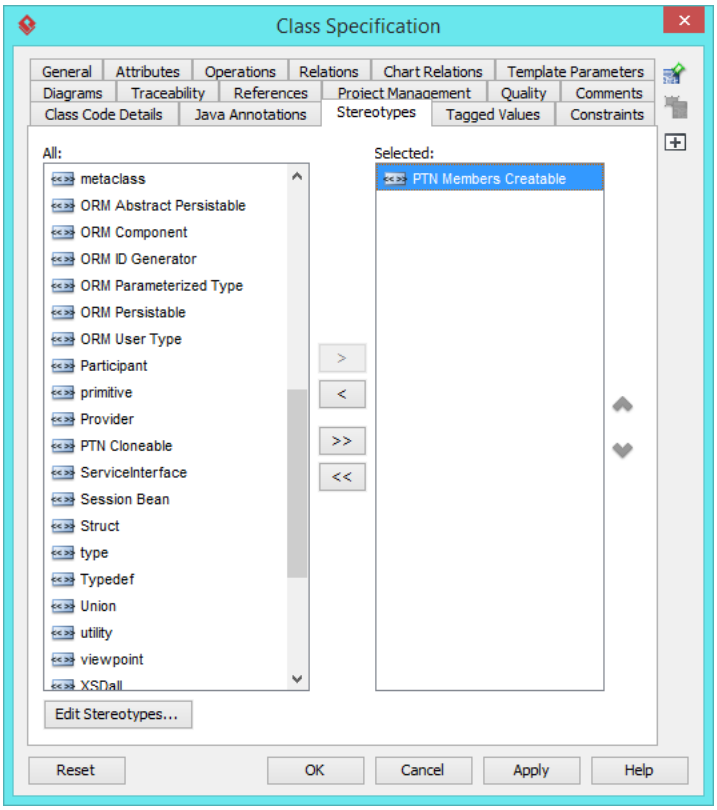
12. Move the mouse cursor over the *Component* class, and drag out **Composition** > **Class** to *Component*. Name the Component's role as *children*.



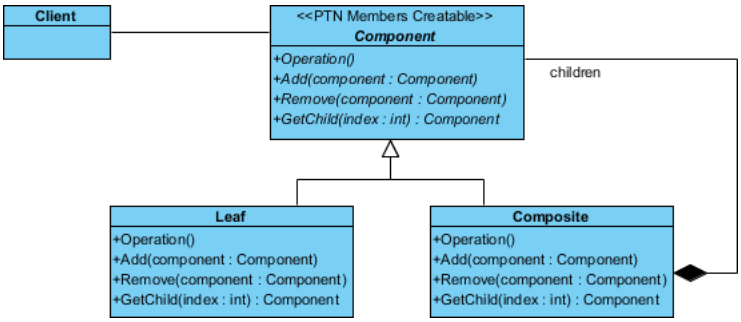
13. In practice, there may be multiple operations in *Component*. To represent this, stereotype the class *Component* as **PTN Members Creatable**. Right click on *Component* and select **Stereotypes** > **Stereotypes...** from the popup menu.



14. In the **Stereotypes** tab of the **Class Specification** dialog box, select **PTN Members Creatable** and click > to assign it to *Component* class. Click **OK** to confirm.

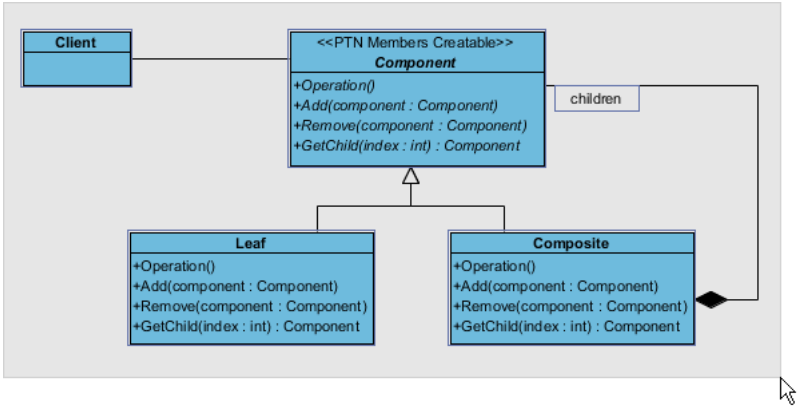


Up to now, the diagram should look like this:

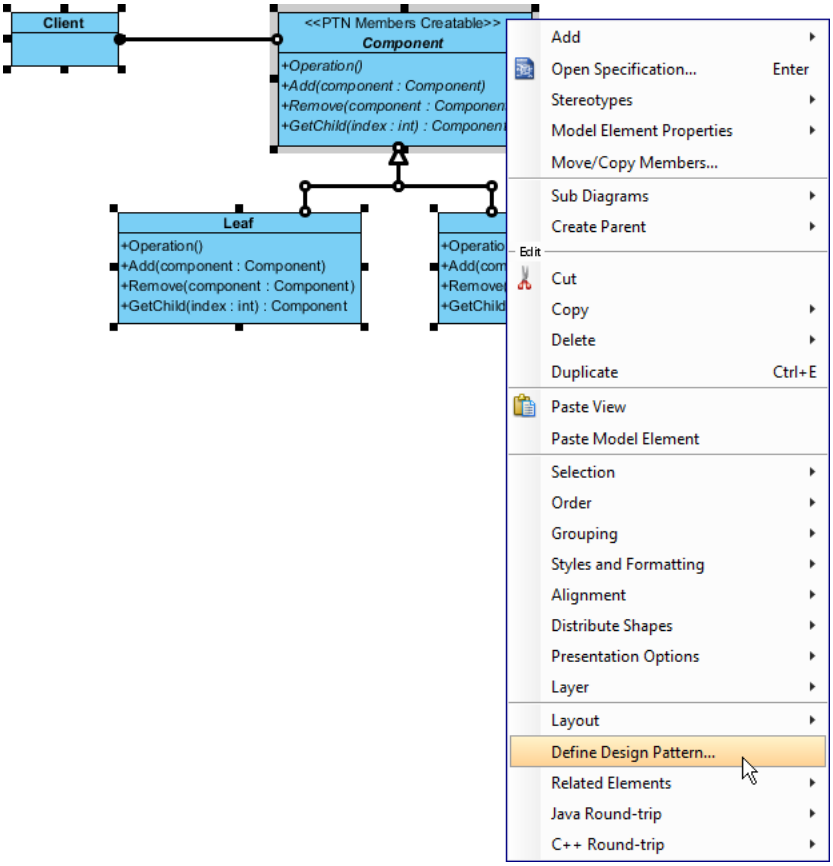


Defining Pattern

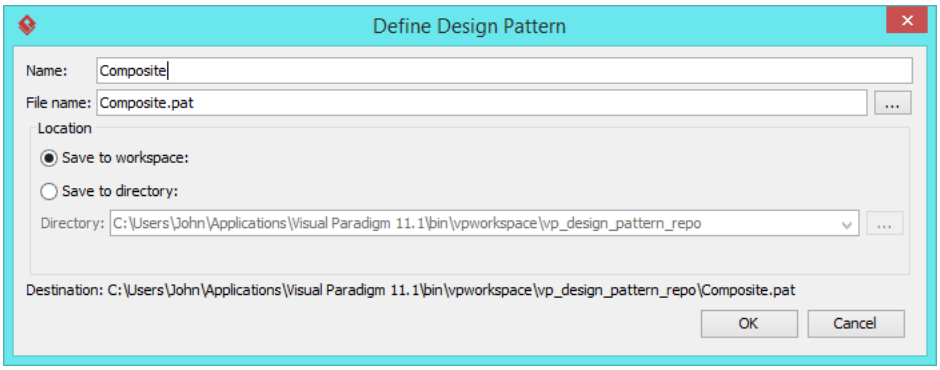
1. Select all classes on the class diagram.



2. Right click on the selection and select **Define Design Pattern...** from the popup menu.



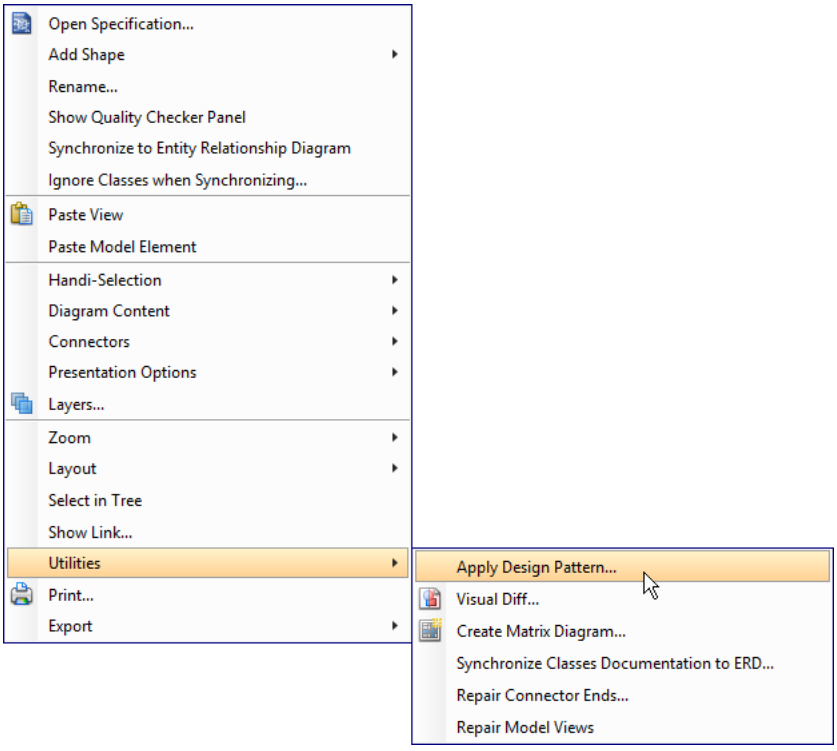
3. In the **Define Design Pattern** dialog box, specify the pattern name *Composite*. Keep the file name as is. Click **OK** to proceed.



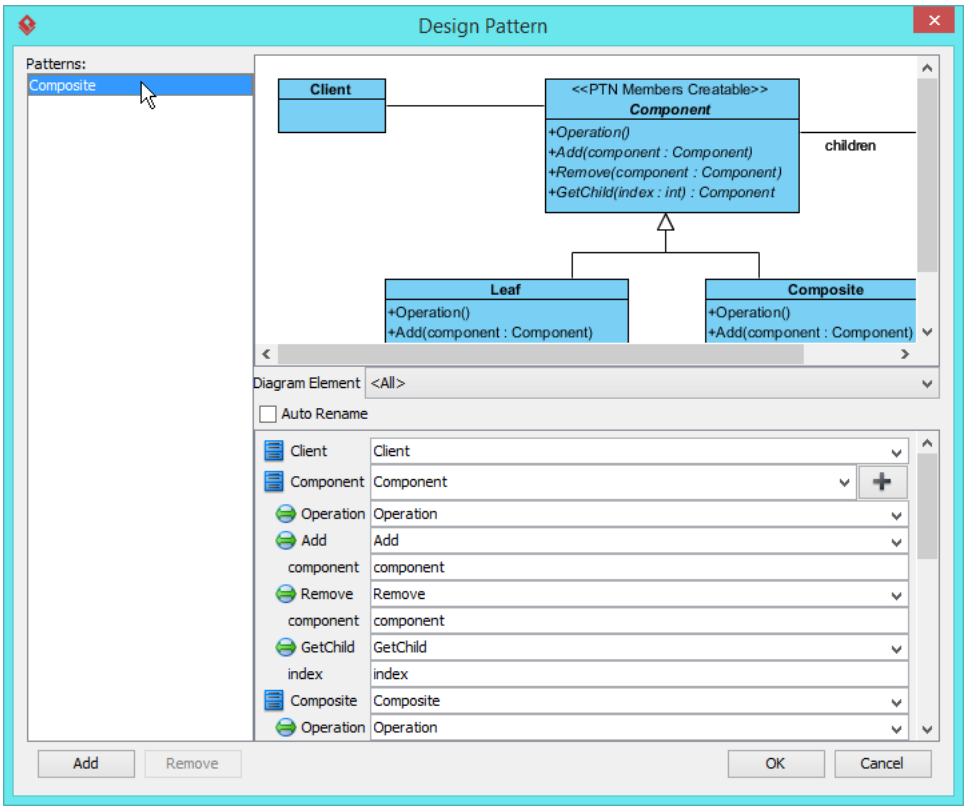
Applying Design Pattern on Class Diagram

In this section, we are going to apply the composite pattern to model a furniture shop's furniture catalog.

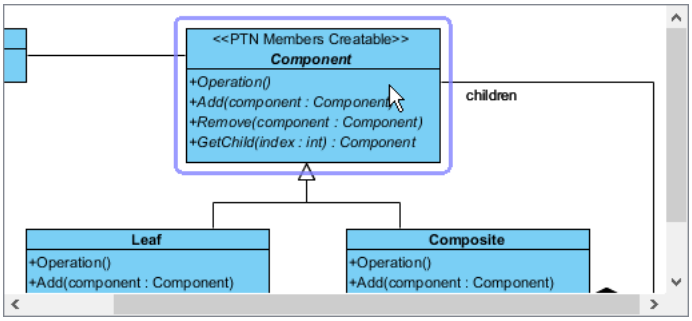
1. Create a new project *Furniture Shop*.
2. Create a class diagram *Domain Model*.
3. Right click on the class diagram and select **Utilities** > **Apply Design Pattern...** from the popup menu.



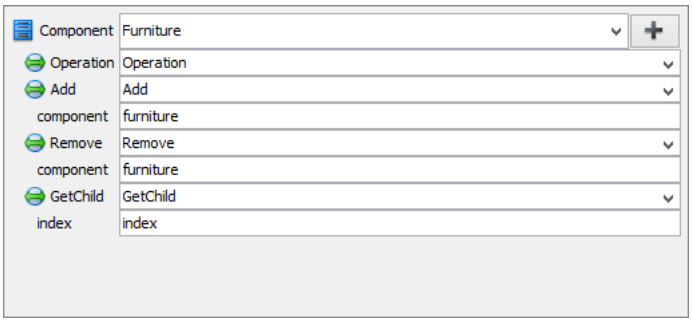
4. In the **Design Pattern** dialog box, select *Composite* from the list of patterns.



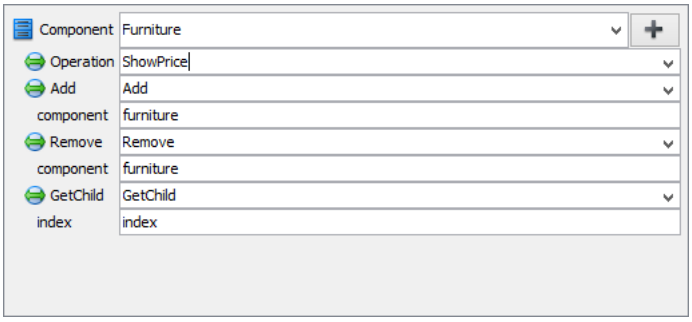
5. Click on *Component* in the overview.



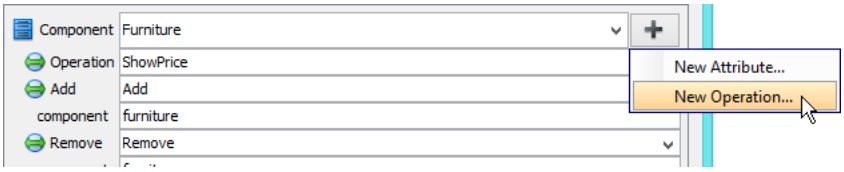
6. Rename *Component* to *Furniture*, and the parameters *component* in various operations to *furniture* at the bottom pane.



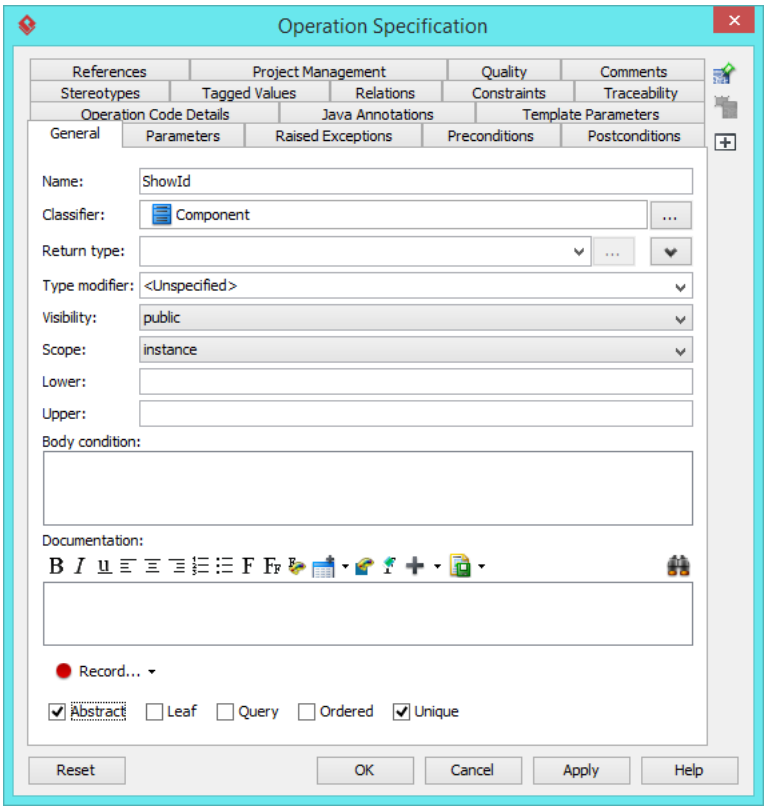
7. Rename *Operation* to *ShowPrice*.



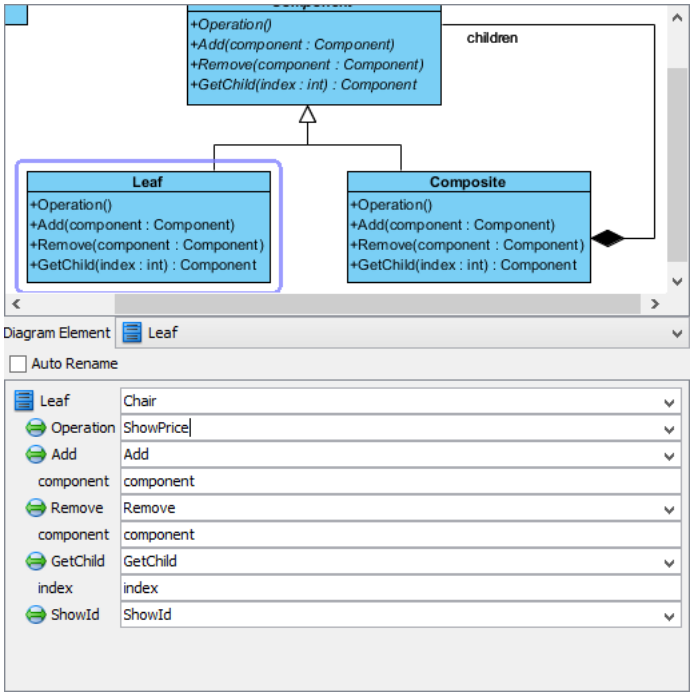
8. Besides the operation *ShowPrice*, we also need one more operation for *ShowId*. Keep *Component* selected, click on the + button at the bottom pane, and select **New Operation...** from the popup menu.



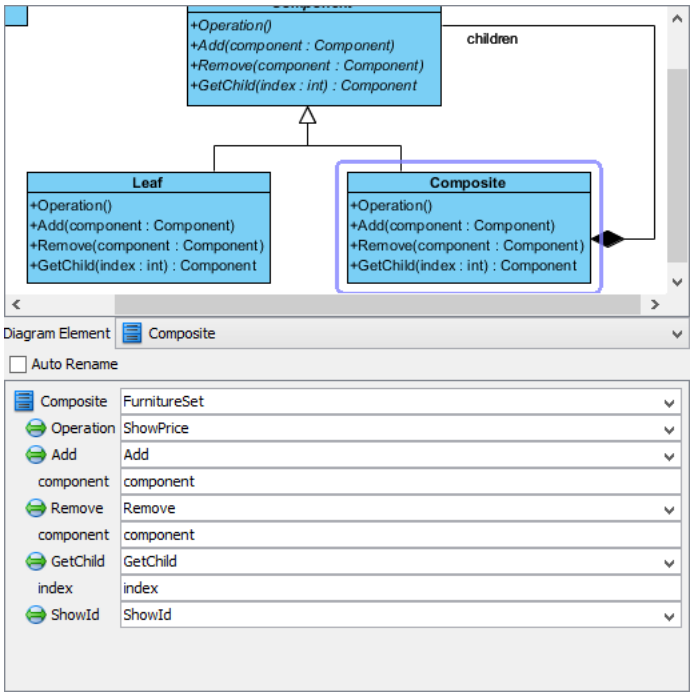
9. In the **Operation Specification** dialog box, name the operation *ShowId*. Check **Abstract** at the bottom of dialog box.



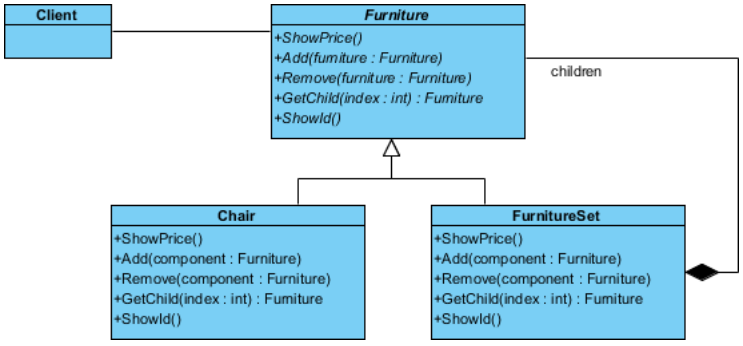
10. Select *Leaf* in overview, and rename it as *Chair* at the bottom pane. Rename also the operation *Operation* to *ShowPrice*. Note that if the option **Auto Rename** is on, rename of operation is not needed as this will be done automatically.



11. Select *Composite* in overview, and rename it as *FurnitureSet* at the bottom pane. Rename also *Operation* to *ShowPrice*. Click **OK** to apply the pattern to diagram.



This is the result:



Resources

1. Composite.pat
2. Design Patterns.vpp

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/)