|  |  |  |
| --- | --- | --- |
| Section 1 | Click Instructions | Talking Points |
| Beacon.pngimage.png  **Click Here** | Open the PartsUnlimited solution. | Rather than trusting to developers to somehow know which components can reference other components, we want to be able to enforce our dependencies. The first step in enforcing the dependencies is to create a dependency diagram.  Note: Visual Studio Enterprise is required to author and edit diagrams, but Visual Studio Pro and Community can be used to open the diagrams in read-only mode. |
| Beacon.pngimage.png  **Click Here** | 1. Click Architecture->New Dependency Validation Diagram. |  |
| Beacon.pngimage.png  **Click Here** | 1. Enter PartsUnlimited.Dependencies for the project name and press OK. |  |
| Beacon.pngimage.png  **Click Here** | 1. In the Solution Explorer, you should see a yellow information block telling you to update your projects for dependency validation. Press the Update button. | The reason for this update is that VS 2017 now uses Roslyn analyzers to perform the dependency validation. Updating the projects installs the analyzers. It will also create a linked item to the dependency diagrams (they will appear in other projects but there is still only a single copy of the diagram - the one in the dependency validation project itself). |
| Beacon.pngimage.png  **Click Here** | 1. In the PartsUnlimited project, expand the References. | You can see the Dependency Validation tools underneath the Analyzers section of the References. |
| Beacon.pngimage.png  **Click Here** | 1. Open the diagram (DependencyValidation1.layerdiagram) and click Class View. |  |
| Beacon.pngimage.png  **Click Here** | 1. Expand the PartsUnlimitedWebsite node to show all the namespaces in the PartsUnlimitedWebsite project. |  |
| Beacon.pngimage.png  **Click Here** | 1. Click and hold PartsUnlimited.Controllers and drag to canvas. |  |
| Beacon.pngimage.png  **Click Here** | 1. Drag PartsUnlimited.ProductSearch to the canvas. |  |
| Beacon.pngimage.png  **Click Here** | 1. Drag PartsUnlimited.ViewModels to the canvas. |  |
| Beacon.pngimage.png  **Click Here** | 1. CS11: Click Here |  |
| Beacon.pngimage.png  **Click Here** | 1. View the Properties for the ViewModels, and change the color to Grey. | You can change the colors of the items on the diagram for clarity. |
| Beacon.pngimage.png  **Click Here** | 1. Right click PartsUnlimited.Controllers |  |
| Beacon.pngimage.png  **Click Here** | 1. Click Add->Dependency. This creates a dependecy line originating in the PartsUnlimited.Controllers namespace. |  |
| Beacon.pngimage.png  **Click Here** | 1. Left-click the PartsUnlimited.ViewModels element. | We've now created an allowed dependency between the Controllers and the View Models. |
| Beacon.pngimage.png  **Click Here** | 1. Press Ctrl-S to save the diagram. If you are prompted that "There are validation errors, continue save?" press Yes. | You can see we already have dependency validation errors. |
| Beacon.pngimage.png  **Click Here** | 1. Open the error list. Double-click on the top error to open SearchController.cs. |  |
| Beacon.pngimage.png  **Click Here** | 1. Highlight the red squiggles. | Not only do we see the errors in the Error List window, we can see them in the code as squigglies! |
| Beacon.pngimage.png  **Click Here** | 1. Switch to the DependencyValidation1.layerdiagram. | We'll now add a dependency from PartsUnlimited.Controllers to PartsUnlimited.ProductSearch to complete our architecture. |
| Beacon.pngimage.png  **Click Here** | 1. Right-click the PartsUnlimited.Controllers element and click Add->Dependency. This creates a dependecy line originating in the PartsUnlimited.Controllers namespace. |  |
| Beacon.pngimage.png  **Click Here** | 1. Now left-click the PartsUnlimited.ProductSearch element. |  |
| Beacon.pngimage.png  **Click Here** | 1. Rebuild the solution. Now close the dependency validation diagram. | Notice that now when we build the solution, there are no longer any errors. Let's close the dependency validation diagram. |
| Beacon.pngimage.png  **Click Here** | 1. In the solution explorer, open PartsUnlimitedWebsite\ProductSearch\StringContainsProductSearch.cs file. Note the comment on line 19, just above the Search method. | We don't even need to have the diagram open in order to see dependency violations. |
| Beacon.pngimage.png  **Click Here** | 1. Change line 20 from 2. public async Task<IEnumerable<Product>> Search(string query) to 3. public async Task<IEnumerable<ProductViewModel>> Search(string query) 4. Resolve the namespace. | What happens if we're coding as part of our day-to-day work and we change violate dependencies? Before, we would be unaware until much later in the process, assuming we uncover the error at all. However, now that we have a diagram in place, we can rely on Live Dependency Validation to quickly warn us about violations as we code. |
| Beacon.pngimage.png  **Click Here** | 1. Hover over the ProductViewModel object (with the squigglies) and read the error. | Right as we code we're prevented from violating dependencies. We can now amend the diagram or table the change - either way, Dependency Validation has saved us from inadvertently violating dependencies.  Once the code is committed to the source control repository, the whole team to benefit from the dependency validation including in Visual Studio Pro and Community where dependency diagrams cannot be created but can be viewed as read-only. |