

KidCoder™: Game Programming

Visual Basic “PowerPack” Workarounds

Updated Feb 3rd, 2012

This document contains a detailed set of instructions to overcome errors students may encounter when trying to use the “Bouncing Lines” activity starters or solution projects from the *KidCoder™: Game Programming* Second Edition course.

The “Bouncing Lines” activity and a few other course activities uses shapes such as “LineShape” that are part of the Visual Basic “Power Pack”. In rare cases the Visual Basic IDE may not have the Power Pack shapes installed by default (though it should). Or there could be a version mismatch between the Power Pack components on your machine and the course projects. Either condition can produce numerous unusual errors regarding the shape controls.

Overview

Resolving the compile-time errors is a straightforward, though slightly technical process. Here is what you will do:

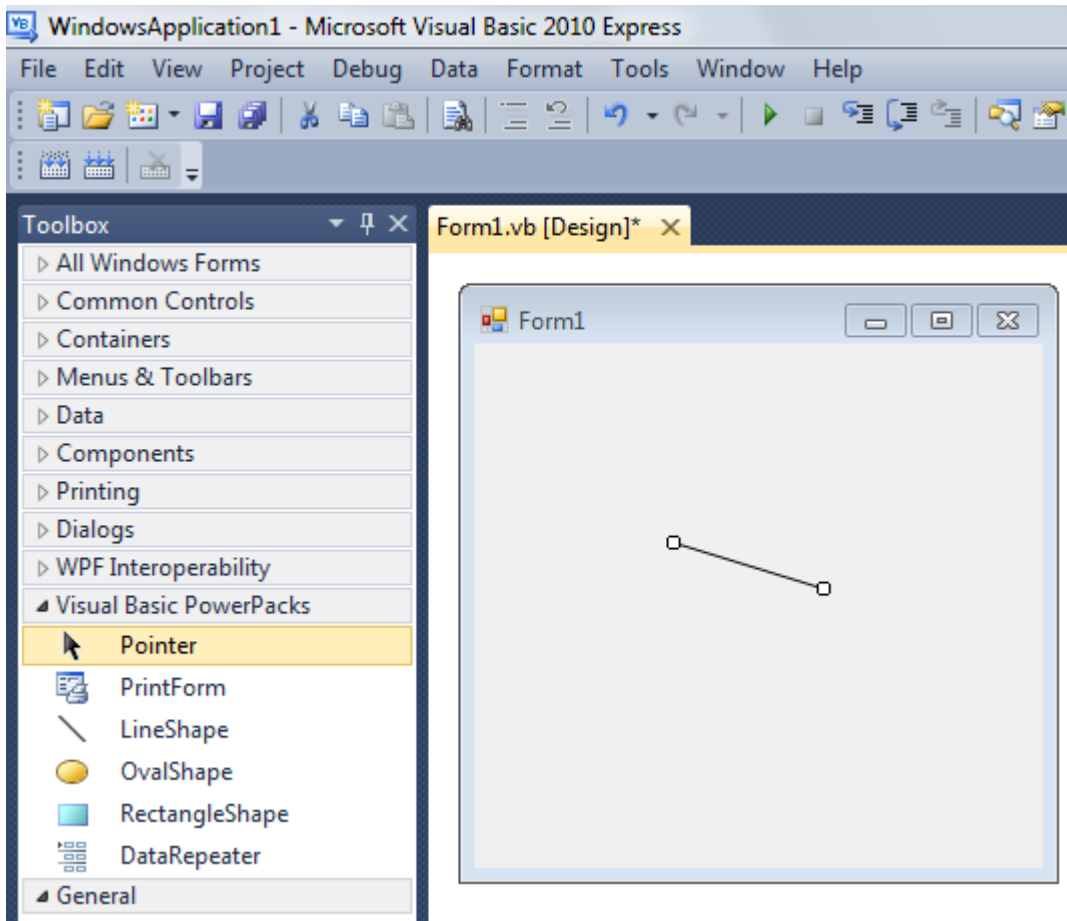
1. Determine if the Power Pack shapes are installed on your machine.
2. If the Power Pack is not installed, perform a special install step to make the shapes available.
3. If the Power Pack is installed, ensure that the Power Pack project references in “Bouncing Lines” and other projects match the version of Power Pack on your computer.

Determine if the Power Pack Shapes are Installed

To see if Power Pack shapes are installed on your computer, perform the following quick test:

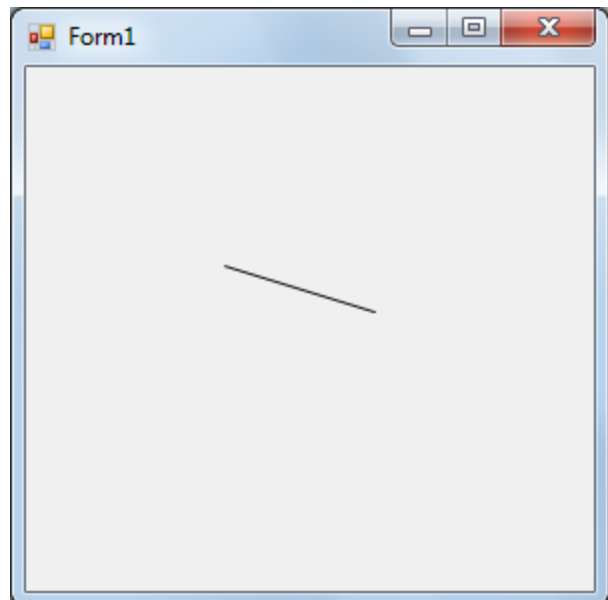
- Create a new empty project, and then go to the Form Design screen.
- On the Toolbox pane to the left, towards the bottom, you should see a section headed "Visual Basic PowerPacks", which contains things like Pointer, PrintForm, LineShape, OvalShape, RectangleShape, and DataRepeater.

- Click on the LineShape and drag a line onto your form as shown below.



- Then simply click on the green “play” arrow button at the top to launch your program, and it should show your form with the line:

If this test is successful, then your installation has what it needs to use the line and other shapes. You can skip the next Power Pack download and installation step and proceed to ensuring that the Power Pack versions match the project settings. If you cannot find the Visual Basic Power Pack shapes in your Toolbox, proceed to the next step!



Installing Missing Power Pack Shapes

If your test above determines that the Power Pack shapes are not installed or not visible on your computer, then:

- You can manually install the PowerPack shape components by downloading from here: <http://msdn.microsoft.com/en-us/vstudio/bb735936.aspx>.

Once installed, confirm that you can complete the quick line test shows in the first step.

It may be possible that your PowerPack is installed but not visible in the Toolbox.

- You can follow the instructions listed here to make them visible if that is the case: <http://download.microsoft.com/download/f/8/f/f8f8f094-85d5-4628-829b-37a320b8c428/readme.htm>

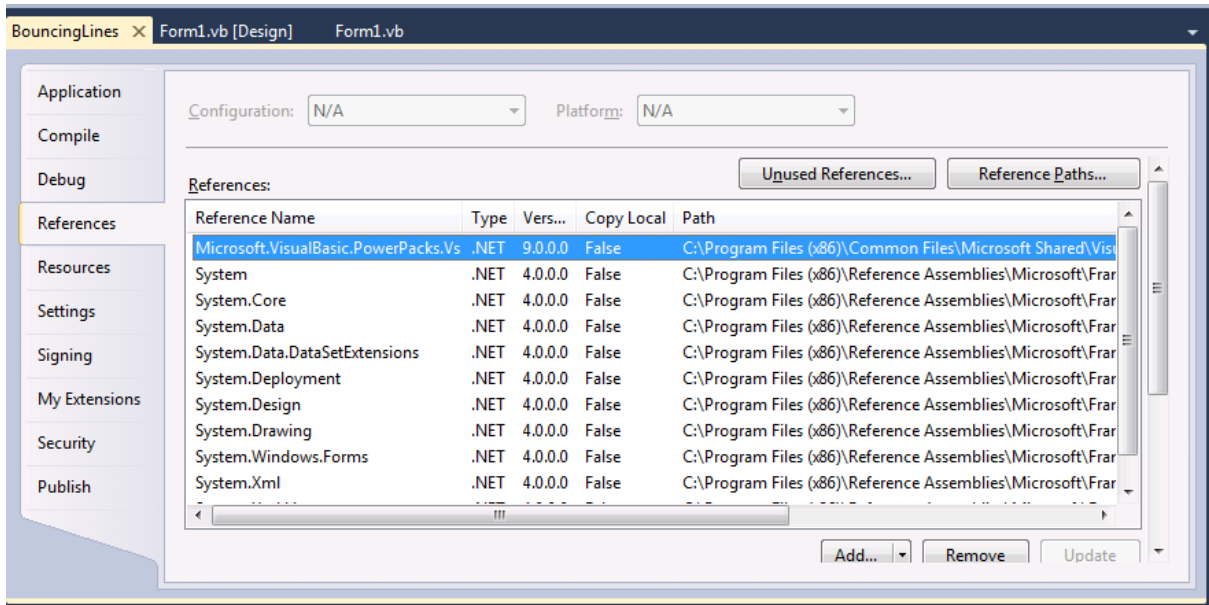
Ensure Power Pack Version Matches

The *KidCoder: Game Programming* second edition course was built when the current version of the Power Packs was “9.0.0”. On some computers you may have an older or newer version that is causing the problem.

This step will change your project settings for “Bouncing Lines” (and you can repeat on others as needed) so the program will use the version of the Power Pack that you have on your computer.

- From your “Solution Explorer”, right-click on the “Bouncing Lines” project and select “Properties”.
- Then click on the “References” tab and you should see a display like shown below.

Notice the “Microsoft.VisualBasic.PowerPacks.Vs line is listed with version 9.0.0.0 originally.



- Remove the reference the original Power Pack by selecting the “Microsoft.VisaulBasic.PowerPacks.Vs” line and clicking the “Remove” button.
- Then click the “Add” button.
 - You will get a dialog with several tabs at the top.
 - Click on “.NET” and then scroll down to find the “Microsoft.VisualBasic.PowerPacks.Vs” entry and click “OK”.
 - If you have multiple versions, pick the highest one.
- Then rebuild your project, and in most cases all of your strange shape-related errors will be resolved!

One additional rare error has been reported after completing the above step. You may see a single error like this:

```
The designer cannot process Form1.designer.vb 27 0
the code at line 26:
Me.ShapeContainer1=New
microsoft.visualBasic.Power
```

To overcome this error:

- Click on the existing “BouncingLine” line component in the Form Design screen and delete it (just hit the delete key). That should then produce a number of errors like this:

Error 1 'BouncingLine' is not declared. It may be inaccessible due to its protection level.

- The step above will also strip away any mention of “ShapeContainer” in your Form1.Designer.vb file. This was the component causing the remaining stray error.
- Then from the Toolbox, drag another LineShape onto your form and reset the **(Name)** property in the Property Sheet to “BouncingLine”.
- Once you save and rebuild your project, the newly injected errors listed above should go away, and any remaining error related to the “ShapeContainer” will go away as well.