

Nant Scripting Language

This example shows how to use boo to enrich your nant build files.

boo includes some [nant tasks](#), more specifically, the boo task allows you to embed boo code directly into your nant build files. The nice thing is that the code has access to the entire nant object model through the Project reference.

Hello, world!

```
<boo>
print("Hello from boo task!")
print("Framework directory: ${Project.TargetFramework.FrameworkAssemblyDirectory}")
</boo>
```

Printing all the properties

```
<boo>
for p as System.Collections.DictionaryEntry in Project.Properties:
    print("${p.Key}: ${p.Value}")
</boo>
```

Creating a .resources file with an image list

```
<target name="create-resources" depends="init">

  <boo>
    import System.IO
    import System.Resources
    import System.Windows.Forms from System.Windows.Forms
    import System.Drawing from System.Drawing

    def MapPath(path):
      return Path.Combine(Project.BaseDirectory, path)

    def CreateImageList():
      imageList = ImageList()
      images = (
        "namespace.png",
        "class.png",
        "interface.png",
        "enum.png",
        "field.png",
        "property.png",
        "method.png"
      )

      for image in images:
        fname = MapPath("resources/ClassBrowserIcons/${image}")
        imageList.Images.Add(Image.FromFile(fname))

      return imageList

    using imageList=CreateImageList():
    using
writer=ResourceWriter(MapPath("build/BooExplorer.DocumentOutline.resources")):
      writer.AddResource("_imageList", imageList.ImageStream)
  </boo>

</target>
```