# Notes for language alias extension for Small Basic

Once compiled this extension can be used to create language variants for all default Small Basic commands present in SmallBasicLibrary.

## Requirements

* Each object class must have a different name than the original. For this reason in this template all object class names are prefixed with underscore.
* One C# (\*.cs) file is present for each object class.
* Only European Latin based alphabets can be used for object, method, property and event names, all other intellisense may be in any language script (e.g. Chinese).

## Example

Using the Stack object as an example, the template code is:

using Microsoft.SmallBasic.Library;

namespace SBAliases

{

/// <summary>

/// This object provides a way of storing values just like stacking up a plate. You can push a value to the top of the stack and pop it off. You can only pop the values one by one off the stack and the last pushed value will be the first one to pop out.

/// </summary>

[SmallBasicType]

public static class \_Stack

{

/// <summary>Pushes a value to the specified stack.</summary>

/// <param name="stackName">The name of the stack.</param>

/// <param name="value">The value to push.</param>

public static void PushValue(Primitive stackName, Primitive value)

{

Stack.PushValue(stackName, value);

}

/// <summary>Gets the count of items in the specified stack.</summary>

/// <param name="stackName">The name of the stack.</param>

/// <returns>The number of items in the specified stack.</returns>

public static Primitive GetCount(Primitive stackName)

{

return Stack.GetCount(stackName);

}

/// <summary>Pops the value from the specified stack.</summary>

/// <param name="stackName">The name of the stack.</param>

/// <returns>The value from the stack.</returns>

public static Primitive PopValue(Primitive stackName)

{

return Stack.PopValue(stackName);

}

}

}

The object and function names in **blue** may be changed to any European Latin based alphabet values.

The intellisence and parameter variables in **red** can be changed to any text.

An example of this conversion for Hindi is below:

using Microsoft.SmallBasic.Library;

namespace SBAliases

{

/// <summary>

/// यह ऑब्जेक्ट किसी प्लेट को ढेर करने की तरह ही मूल्यों को संग्रहीत करने का एक तरीका प्रदान करता है। आप किसी मान को स्टैक के शीर्ष पर धकेल सकते हैं और उसे पॉप ऑफ कर सकते हैं। आप केवल मानों को स्टैक से एक-एक करके पॉप कर सकते हैं और अंतिम धक्का दिया गया मान सबसे पहले पॉप आउट होगा.

/// </summary>

[SmallBasicType]

public static class Dher

{

/// <summary>किसी मान को निर्दिष्ट स्टैक पर धकेलता है.</summary>

/// <param name="ढेरकानाम">ढेर का नाम.</param>

/// <param name="कीमत">धकेलने का मूल्य.</param>

public static void PushVailyoo(Primitive stackName, Primitive value)

{

Stack.PushValue(stackName, value);

}

/// <summary>निर्दिष्ट स्टैक में आइटमों की गिनती प्राप्त करता है.</summary>

/// <param name="ढेरकानाम">ढेर का नाम.</param>

/// <returns>निर्दिष्ट स्टैक में वस्तुओं की संख्या.</returns>

public static Primitive GinateePraaptKaren(Primitive stackName)

{

return Stack.GetCount(stackName);

}

/// <summary>निर्दिष्ट स्टैक से मान पॉप करता है.</summary>

/// <param name="ढेरकानाम">ढेर का नाम.</param>

/// <returns>स्टैक से मूल्य.</returns>

public static Primitive PopVailyoo(Primitive stackName)

{

return Stack.PopValue(stackName);

}

}

}

## Compiling

The modified C# source code may be compiled for a dll using the *compile.bat* command included. This file compiles the SBAliases.dll and SBAliases.xml. It can be modified as required, as it is a simple batch command:

@echo off

set name=SBAliases

if not "%~1"=="" (set name=%1)

"C:\Windows\Microsoft.NET\Framework64\v4.0.30319\csc.exe" /target:library /reference:"C:\Program Files (x86)\Microsoft\Small Basic\SmallBasicLibrary.dll" /out:%name%.dll /doc:%name%.xml \*.cs

Pause

You may need to use a different csc (CSharpCompiler) depending on the version of .Net installed.

Compiler warnings CS1572 and CS1573 can safely be ignored.

Alternatively, it can be compiled using the included VS solution.