



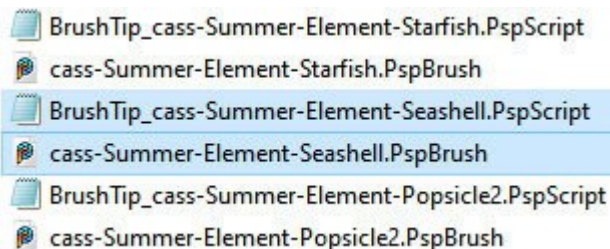
Brushes need two files. Really?

Description

If you use brushes in PaintShop Pro, you might not care much about the file system that you need. However, if you purchase, download, or share brushes, this is important information. Let's have a look at the files needed for a brush to work correctly in PaintShop Pro.

What are those files?

If you look at the brushes that you already have, you will notice one called **[brushname].pspbrush**, and one called **BrushTip_[brushname].pspscript**. Here is an example of those brushes being listed in Windows Explorer.

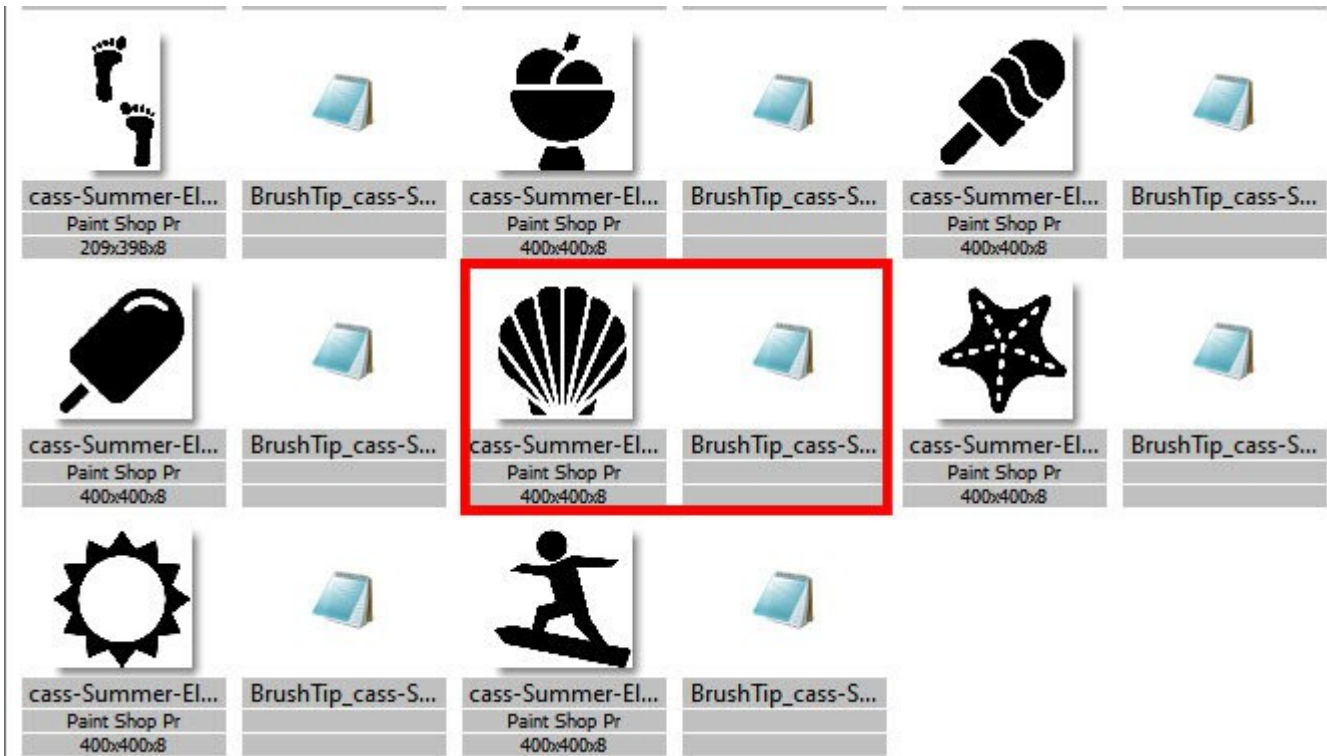


The file with the .pspbrush extension is, as you would expect, the information about the shape of the brush. So if the shape is a star, coffee mug, or a grungy design, that is the file that will contain that "design".

On the other hand, the file ending with .pspscript is a collection of information about the brush settings and behavior. Even though it seems like it is a script, don't put it in the scripts folder.

Let's look at the brush file

The .pspbrush file cannot be viewed in Windows Explorer, but if you have **XnView**, it will display the design for each brush.



You can see that both files are listed, but only the **.pspbrush** file has a design. The designs you see here are the same you will see in the **Brushes** download list in your PaintShop Pro.



Let's look at the **BrushTip** file

Although it does not seem logical, the **BrushTip_** file does not show the shape of the brush tip. Instead, it includes very specific information about that brush. You can open this file with Notepad since it is just a text file. Let's look at the information in this particular file.

```
BrushTip_cass-Summer-Element-Seashell.PspScript - Notepad
File Edit Format View Help
|from JascApp import *

def ScriptProperties():
    return {
        'Author': u'Cassel',
        'Copyright': u'CreationCassel',
        'Description': u''
    }

def Preset_BrushTip():
    return {
        'Shape': App.Constants.BrushShape.Custom,
        'CustomBrush': u'cass-Summer-Element-Seashell',
        'Size': 400,
        'Hardness': 100,
        'Step': 100,
        'Density': 100,
        'Thickness': 100,
        'Rotation': 0
    }
```

It is not a big file and most of the information listed here should be familiar.

The **Author** and **Copyright** information are information you typed when you exported the brush (or someone else). If you never put that information and you share your brushes, it will be harder for the user to know where they come from.

BrushTip_cass-Summer-Element-Seashell.PspScript - Notepad

File Edit Format View Help

```
from JascApp import *
```

```
def ScriptProperties():
```

```
    return {
```

```
        'Author': u'Cassel',  
        'Copyright': u'CreationCassel',  
        'Description': u''
```

```
    }
```

```
def Preset_BrushTip():
```

```
    return {
```

```
        'Shape': App.Constants.BrushShape.Custom,  
        'CustomBrush': u'cass-Summer-Element-Seashell',  
        'Size': 400,  
        'Hardness': 100,  
        'Step': 100,  
        'Density': 100,  
        'Thickness': 100,  
        'Rotation': 0
```

```
    }
```

Here is the brush name. Notice that it has to match perfectly the other file name.

BrushTip_cass-Summer-Element-Seashell.PspScript - Notepad

File Edit Format View Help

```
from JascApp import *
```

```
def ScriptProperties():  
    return {  
        'Author': u'Cassel',  
        'Copyright': u'CreationCassel',  
        'Description': u''  
    }
```

```
def Preset_BrushTip():  
    return {  
        'Shape': App.Constants.BrushShape.Custom,  
        'CustomBrush': u'cass-Summer-Element-Seashell',  
        'Size': 400,  
        'Hardness': 100,  
        'Step': 100,  
        'Density': 100,  
        'Thickness': 100,  
        'Rotation': 0  
    }
```

Finally, you can see the default settings associated with this particular brush.

BrushTip_cass-Summer-Element-Seashell.PspScript - Notepad

File Edit Format View Help

```
from JascApp import *
```

```
def ScriptProperties():
```

```
    return {  
        'Author': u'Cassel',  
        'Copyright': u'CreationCassel',  
        'Description': u''  
    }
```

```
def Preset_BrushTip():
```

```
    return {  
        'Shape': App.Constants.BrushShape.Custom,  
        'CustomBrush': u'cass-Summer-Element-Seashell',  
        'Size': 400,  
        'Hardness': 100,  
        'Step': 100,  
        'Density': 100,  
        'Thickness': 100,  
        'Rotation': 0  
    }
```

These are the basic settings, and this brush is a simple one. Let's see a more detailed brush file:

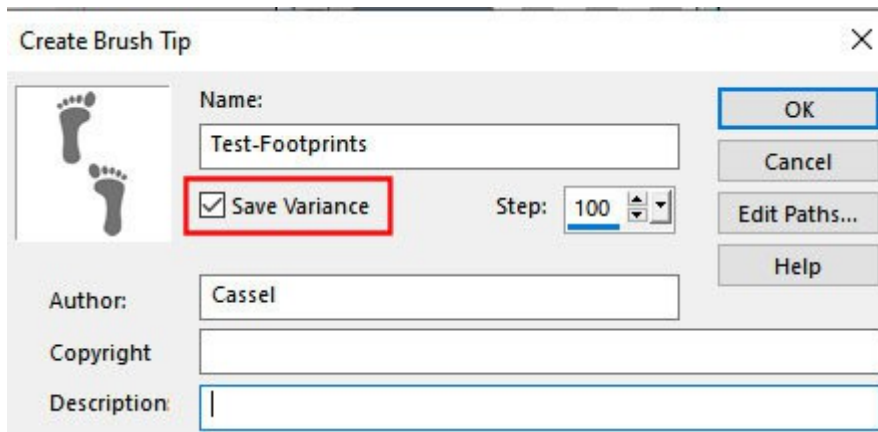

```
from PSPApp import *

def ScriptProperties():
    return {
        'Author': u'Cassel',
        'Copyright': u'',
        'Description': u''
    }

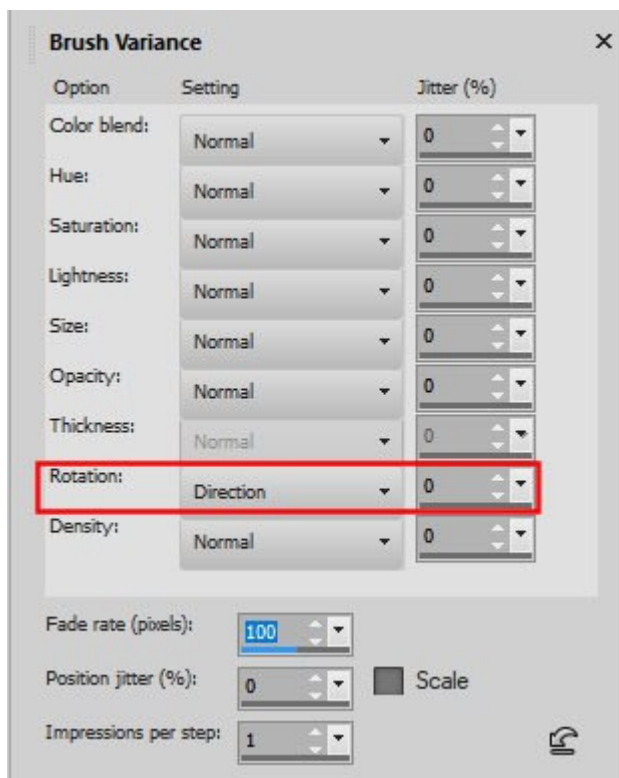
def Preset_BrushTip():
    return {
        'Shape': App.Constants.BrushShape.Custom,
        'CustomBrush': u'Test-Footprints',
        'Size': 377,
        'Hardness': 100,
        'Step': 100,
        'Density': 100,
        'Thickness': 100,
        'Rotation': 0,
        'BrushVariance': {
            'SizeVariance': App.Constants.VarianceMethod.None,
            'SizeJitter': 0,
            'OpacityVariance': App.Constants.VarianceMethod.None,
            'OpacityJitter': 0,
            'DensityVariance': App.Constants.VarianceMethod.None,
            'DensityJitter': 0,
            'ThicknessVariance': App.Constants.VarianceMethod.None,
            'ThicknessJitter': 0,
            'RotationVariance': App.Constants.VarianceMethod.Direction,
            'RotationJitter': 0,
            'ColorBlendVariance': App.Constants.VarianceMethod.None,
            'ColorBlendJitter': 0,
            'HueVariance': App.Constants.VarianceMethod.None,
            'HueJitter': 0,
            'SaturationVariance': App.Constants.VarianceMethod.None,
            'SaturationJitter': 0,
            'LightnessVariance': App.Constants.VarianceMethod.None,
            'LightnessJitter': 0,
            'PositionJitter': 0,
            'UseScaledPositionJitter': False,
            'ImpressionsPerStep': 1,
            'FadeRate': 100
        },
        'BrushTrayShow': False
    }
```

This whole section comes from the **Brush Variance** palette. When you create and export a new custom brush, you have this little checkbox. If you check it, all the settings inside of the **Brush Variance**

palette will also be included in the **BrushTip_** file.



In this example, I changed the **Rotation** setting to **Direction**.



And you can see that particular setting in the **BrushTip_** file:


```
from PSPApp import *

def ScriptProperties():
    return {
        'Author': u'Cassel',
        'Copyright': u'',
        'Description': u''
    }

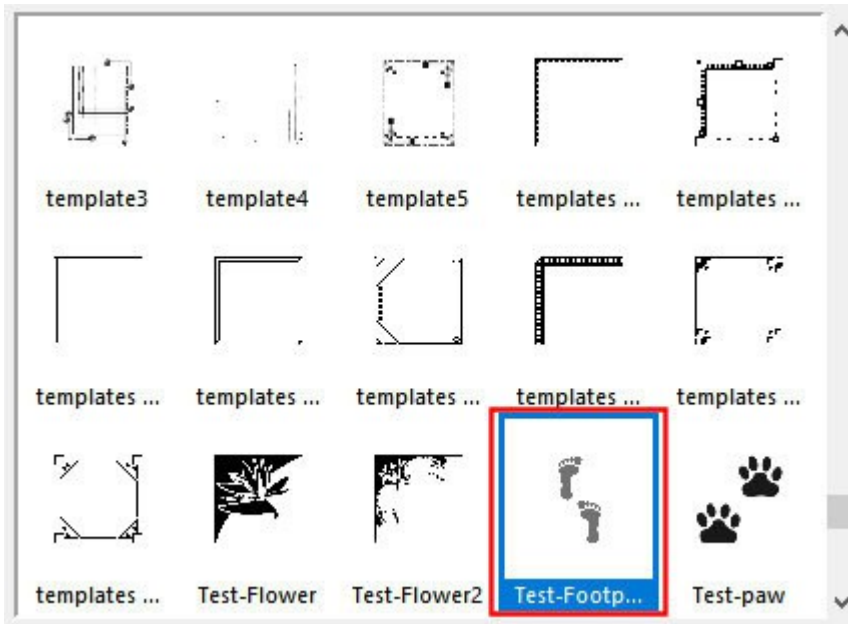
def Preset_BrushTip():
    return {
        'Shape': App.Constants.BrushShape.Custom,
        'CustomBrush': u'Test-Footprints',
        'Size': 377,
        'Hardness': 100,
        'Step': 100,
        'Density': 100,
        'Thickness': 100,
        'Rotation': 0,
        'BrushVariance': {
            'SizeVariance': App.Constants.VarianceMethod.None,
            'SizeJitter': 0,
            'OpacityVariance': App.Constants.VarianceMethod.None,
            'OpacityJitter': 0,
            'DensityVariance': App.Constants.VarianceMethod.None,
            'DensityJitter': 0,
            'ThicknessVariance': App.Constants.VarianceMethod.None,
            'ThicknessJitter': 0,
            'RotationVariance': App.Constants.VarianceMethod.Direction,
            'RotationJitter': 0,
            'ColorBlendVariance': App.Constants.VarianceMethod.None,
            'ColorBlendJitter': 0,
            'HueVariance': App.Constants.VarianceMethod.None,
            'HueJitter': 0,
            'SaturationVariance': App.Constants.VarianceMethod.None,
            'SaturationJitter': 0,
            'LightnessVariance': App.Constants.VarianceMethod.None,
            'LightnessJitter': 0,
            'PositionJitter': 0,
            'UseScaledPositionJitter': False,
            'ImpressionsPerStep': 1,
            'FadeRate': 100
        },
        'BrushTrayShow': False
    }
```

As you can see, that is all the settings the user will need to use the brush the way the author intended.

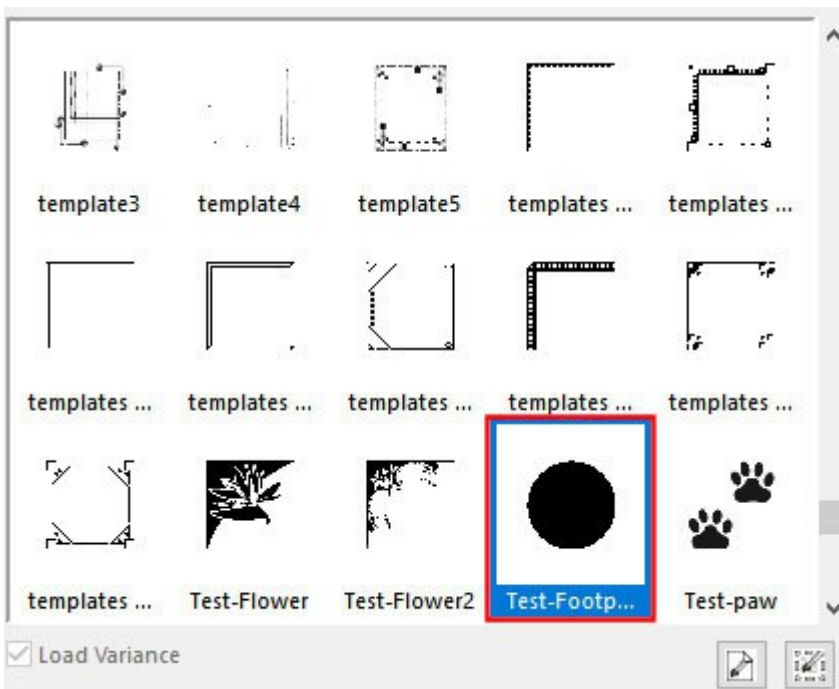
Of course, the user can change any of those settings once the brush is loaded.

What if one file is missing?

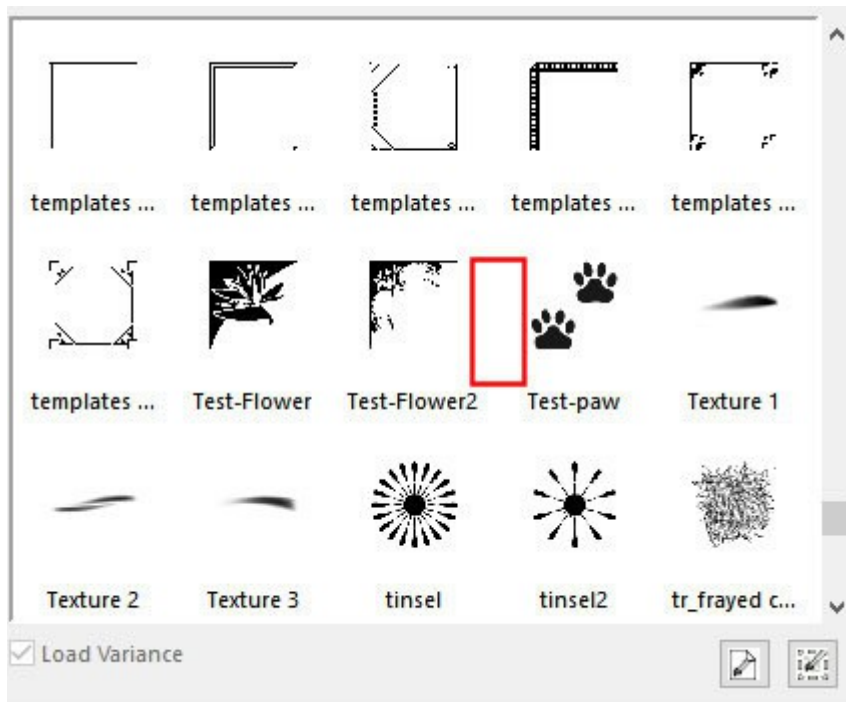
If you have both files in place, you will see the brush in the list, as expected:



However, you might find some old brushes and you don't have both files associated together. If you are missing the **.pspbrush** file, you will see a simple circle in your brushes list:



On the other hand, if you have the **.pspbrush** file but are missing the **BrushTip_** file, the brush won't even be listed in your drop-down.

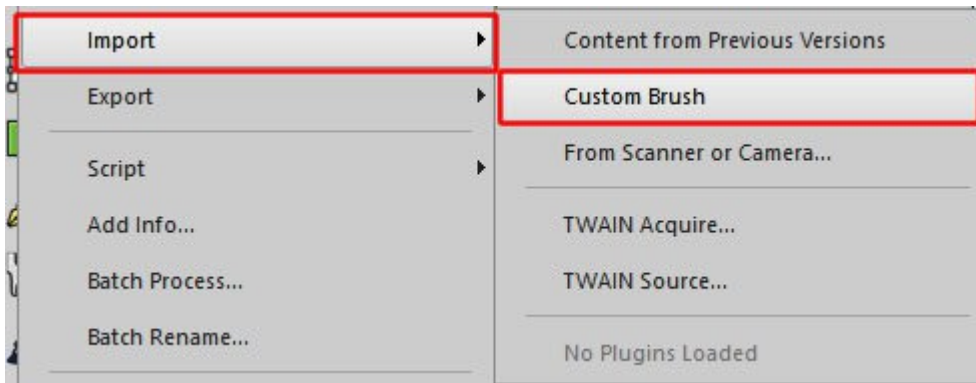


Remember also that both files have to match. If you happen to change the file name for the **.pspbrush** file, you need to edit the **BrushTip_** file with the correct name. You can just do that in Notepad. Make sure you also adjust the name when you save the file. **THEY HAVE TO MATCH**. If you edit one without adjusting the other, you will just end up with two mismatched files and the brush won't appear.

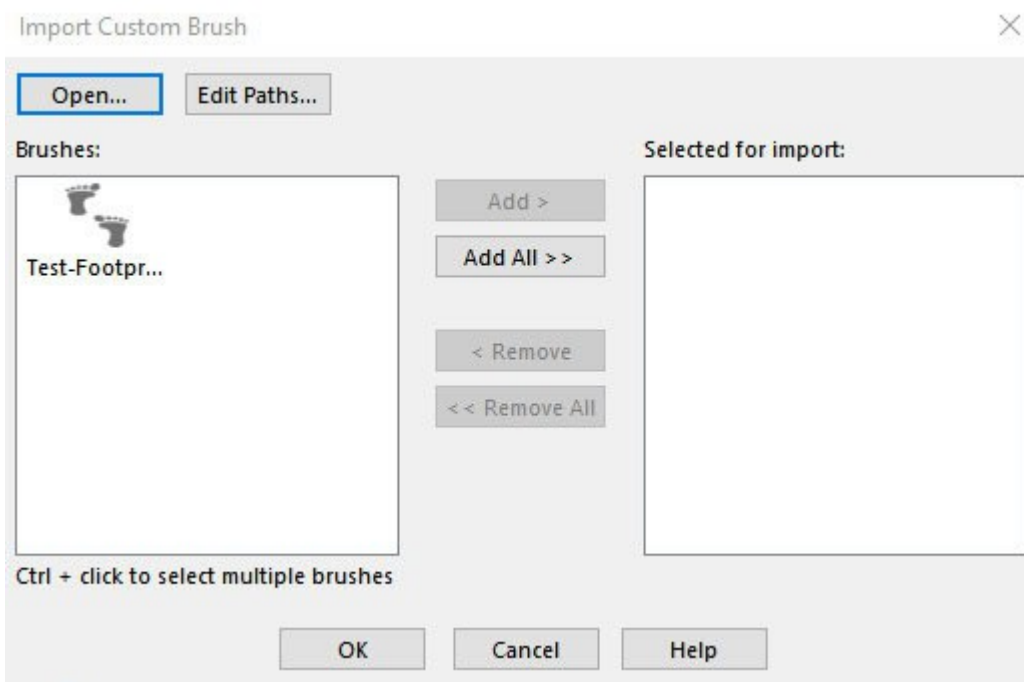
What if I only have the **.pspbrush** file?

Lucky for you, PaintShop Pro can create that **BrushTip_** file. In fact, it will create **BOTH** files, in the correct folder. That means that you have to start with the **.pspbrush** file in a **DIFFERENT** folder than the **Brushes** folder. Don't worry, it can be a temporary location since you will end up with a duplicate. PaintShop Pro just won't want to create both files if one already exists in the **Brushes** folder.

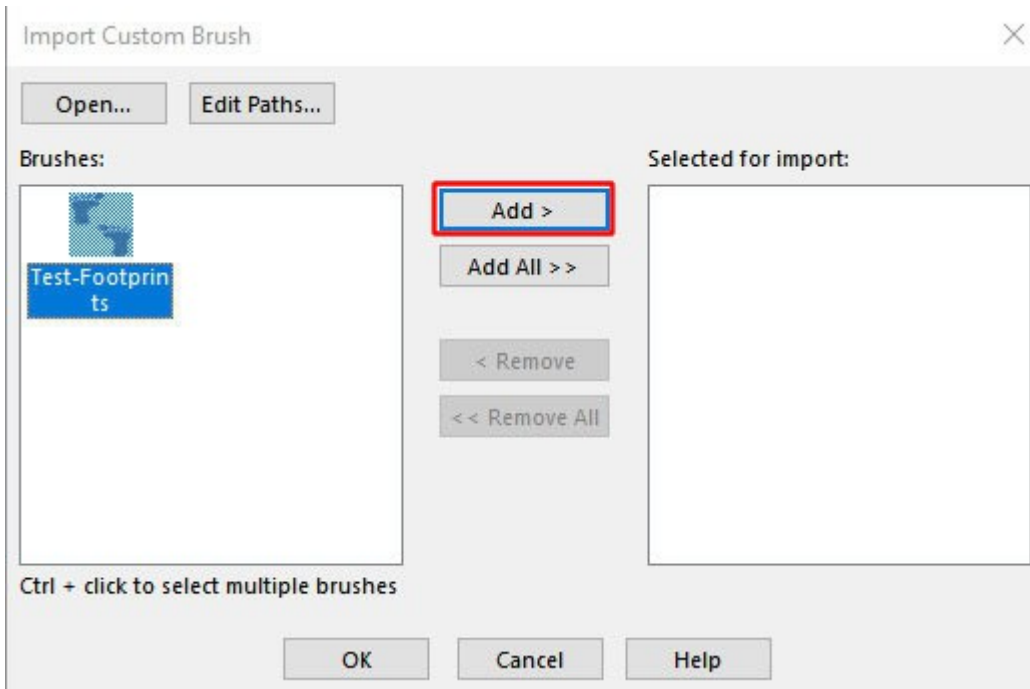
Got to **File > Import > Custom brushes**.



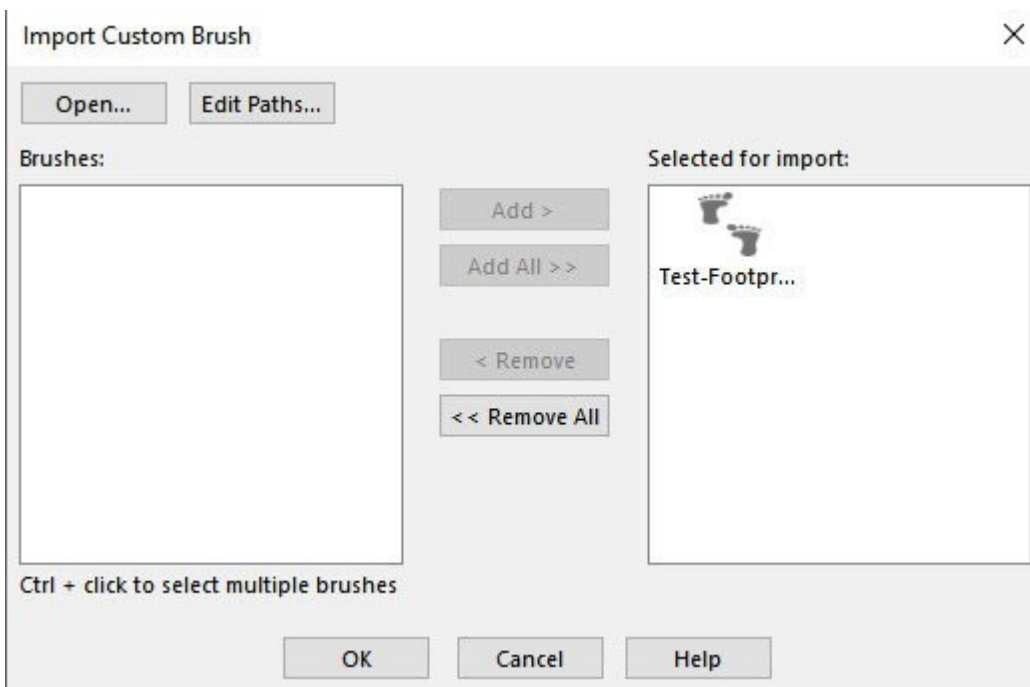
This will open the import dialog window. Click on **Open** and navigate to the .pspbrush file you want to create.



Highlight the brush you want to import and click **Add >** .



Once the brush appears on the right pane, click **OK**.



Now, the brush has both files created and will now appear in the drop-down list of brushes.

Note that the Import function is the same thing you would do if you had ABR brushes since they don't have the .psp files. You can see that process in a video [HERE](#).

Now that you understand what those two files are, for the PaintShop Pro brushes, you will make sure to have them both!

Date Created

August 5, 2024

Author

myclone