

Hello World using PowerPointTemplate

Table of Contents

- [Hello World with PowerPointTemplate](#)
- [Setting up the template file](#)
- [Writing the code](#)
- [Final Code](#)
- [Downloads](#)

Hello World with PowerPointTemplate

PowerPointWriter's PowerPointTemplate approach allows you to write data to a template presentation that contains [data markers](#). The data markers tell PowerPointWriter where to bind specific sets of data. This tutorial will show you the basics of dynamically inserting data into a presentation using PowerPointTemplate by taking custom text from a web form textbox and inserting it into a template file.

Setting up the template file

1. Create a new .PPTX file. Save it as Hello World.pptx.

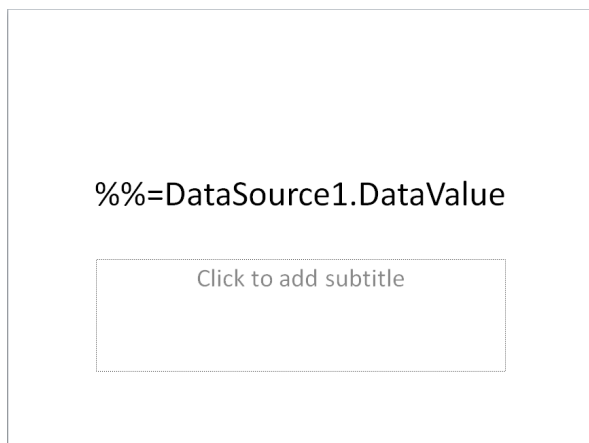


The Hello World Tutorial can be found in the downloadable [PowerPointWriter_HelloWorldC#](#). In the *Hello World* sample web application, the completed template file is located in \templates\Hello World.pptx.

2. We are going to bind a single string value to a slide in a template file. To do this, we will first need to add a data marker to the slide where we want the value to appear.

All PowerPointWriter data markers follow the syntax `%%=DataSourceName.ColumnName`. In this example, we will use the data marker `%%=DataSource1.DataValue` to import the data into the presentation. For more information on data markers see [How to use Data Markers](#).

Below is a screenshot of the completed template file from the *Hello World* sample:



3. Now the template file is done. Next is writing the code to bind the string value to the data marker.

Writing the code



In the sample code, the completed file is: `PowerPointTemplate_HelloWorld.aspx.cs`. The corresponding web form is `PowerPointTemplate_HelloWorld.aspx`.

1. Include the `SoftArtisans.OfficeWriter.PowerPoint` namespace in the code behind.

```
using SoftArtisans.OfficeWriter.PowerPointWriter;
```

2. Instantiate the `PowerPointTemplate` object.

```
PowerPointTemplate PPTT = new PowerPointTemplate();
```

3. Open the template file with `PowerPointTemplate.Open`.

The `PowerPointTemplate` object corresponds to a single template file, so a given `PowerPointTemplate` instance can only have one template file open.

```
PPTT.Open(Page.MapPath("templates\\Hello World.pptx"));
```

4. Create a `DataBindingProperties` object

```
DataBindingProperties DataProps = PPTT.CreateDataBindingProperties();
```

The `DataBindingProperties` object can be used to change the behavior of how data is imported. For example, if we were importing multiple rows of data, we can use the `DataBindingProperties.MaxRowsPerSlide` property to limit the number of rows that are imported. In this sample, we won't be changing any of the import properties, but we still need the `DataBindingProperties` object to bind data.

5. Get the data and create 2 arrays to hold the values and column names of the data you are importing. The column names should match the name of your data marker.

```
string value = DataValueBox.Text.Trim();  
string[] values = { value };  
string[] colNames = { "DataValue" };
```

6. Call `PowerPointTemplate.BindData()` to bind the data to the data marker. The data source name should match the value in your data marker.

```
PPTT.BindData(values, colnames, "DataSource1", DataProps);
```

Note that we need to pass the `DataBindingProperties` object, even though none of the `DataBindingProperties` are active.

7. Call `PowerPointTemplate.Process()` to insert the data into the file

```
PPTT.Process();
```

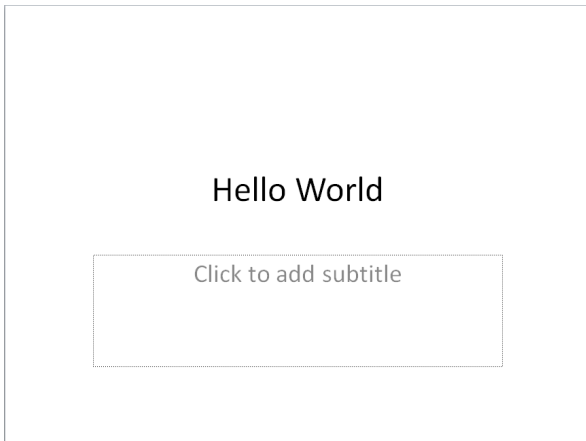
`PowerPointTemplate.Process()` handles everything relating to inserting the data into the file.

8. Save the output

```
PPTT.Save(Page.Response, "Output.pptx", false);
```

There are several options for `PowerPointTemplate.Save` including: save to disk, save to memory stream, stream back to the client inline, and stream back to the client as an attachment. In this case, we're streaming the presentation back to the client as an attachment. Also, `PowerPointWriter` does not convert between file formats, so it is important that the file extension on the output file matches the file extension of the original template file.

9. Go to the web form page, **PowerPointTemplate_HelloWorld.aspx**, to try out the sample. In the output file, you will see that the data marker has been replaced with the custom text entered in the form.



Congratulations, you have completed *Hello World* for PowerPointTemplate!

Final Code

```
using SoftArtisans.OfficeWriter.PowerPointWriter;
...
PowerPointTemplate PPTT = new PowerPointTemplate();
PPTT.Open(Page.MapPath("Hello World.pptx"));
DataBindingProperties DataProps = PPTT.CreateDataBindingProperties();
string value = DataValueBox.Text.Trim();
string[] values = { value };
string[] colNames = { "DataValue" };
PPTT.BindData(values, colNames, "DataSource1", DataProps);
PPTT.Process();
PPTT.Save(Page.Response, "Output.pptx", false);
```

Downloads

You can download the code for the hello world tutorial as a Visual Studio solution.

- [PowerPointWriter_HelloWorldC#](#)