

# Part 3 - Copy Slide

## Table of Contents

- [Intro](#)
- [Adding to the Template](#)
- [Writing the Code](#)
- [Final Code](#)
- [Downloads](#)
- [Next Steps](#)

## Intro



This is Part 3 of a 3-part tutorial series for the [Employee Training](#) scenario. It is recommended that you complete [Part 1 - Importing Data](#) and [Part 2 - The Continue Modifier](#) before starting this section.



### Following the Sample

There is a downloadable [PowerPointWriter\\_BasicTutorials.zip](#) with completed templates and code. The completed example of the template is available under *EmployeeTraining/templates/part3\_template.pptx*. The code for this part of the tutorial can be found in *Part3.aspx.cs*.

This part focuses on passing a presentation between PowerPointTemplate and PowerPointApplication in order to take advantage of the full range of features available with PowerPointWriter.

## Adding to the Template

We will continue adding to our template from [Part 2](#). The next part of our presentation will contain information about the Team the new employee will be working on.

1. Add a new title slide to the presentation. Give it the title 'Your Team'.
2. Place the data marker `%%=Team.Name` underneath the title. Our template slide is shown below.



Your Team

%%=Team.Name

3. Add a new content slide. Place the company logo in the bottom right corner.
4. The title of this slide will be the data marker %%=**Department.TrainingItem**
5. Create a new list in the body of the slide.
6. Place the data marker %%=**TrainingItems.Details** as the first item in the list.

%%=Department.TrainingItem

▶ %%=TrainingItems.Details

%%=CompanyLogo

This is the 8th and final slide in our template. This slide will be used as a template slide for 4 different data sets. To achieve this we will use the `Slides.Copy` method and the `Slide` data binding property.

## Writing the Code

### Following the Sample

There is a downloadable [PowerPointWriter\\_BasicTutorials.zip](#) with completed templates and code. The completed example of the template is available under `ProjectProposal/templates/part3_template.pptx`. The code for this part of the tutorial can be found in `Part3.aspx.cs`.

We will start with an unprocessed template file containing all of the slides from the 3 parts of this tutorial.

1. Create a `PowerPointApplication` object.

```
PowerPointApplication ppta = new PowerPointApplication();
```

2. Set a `Presentation` object to our template file. This allows us to use all of the methods in the `PowerPointApplication` API including `Slides.CopySlide`.

```
Presentation pres = ppta.Open(Page.MapPath("//templates//part3_template.pptx"));
```

The following steps will create 3 copies of the last slide.

3. Create a [Slide](#) object for the last slide in our template.

```
Slide slideToCopy = pres.Slides[7];
```

4. Call [Slides.CopySlide](#) three times, with each call placing the new slide at the end of the presentation.

```
pres.Slides.CopySlide(slideToCopy, (pres.Slides.Count));  
pres.Slides.CopySlide(slideToCopy, (pres.Slides.Count));  
pres.Slides.CopySlide(slideToCopy, (pres.Slides.Count));
```

Now that our presentation contains all of the slides we want, we can pass it to [PowerPointTemplate](#) to import the data. See [Passing Between Template and Application](#) for more information.

```
PowerPointTemplate pptt = new PowerPointTemplate();  
pptt.Open(ppta, pres);
```

We will reuse the code from [Part 1](#) and [Part 2](#) of this tutorial. We will take out the `Save` call and add then our new code for Part 3.

```
//Code from Part 1  
PowerPointTemplate pptt = new PowerPointTemplate();  
pptt.Open(Page.MapPath("//templates//part1_template.pptx"));  
DataBindingProperties dataProps = pptt.CreateDataBindingProperties();  
string[] columnNamesArray = {"Logo"};  
Byte[] logoArray =  
System.IO.File.ReadAllBytes((Page.MapPath("//data//logo.png")));  
object[] valuesArray = {logoArray};  
pptt.BindData(valuesArray, columnNamesArray, "Company", dataProps);  
string[] employeeColNames = {"Name", "Team"};  
object[] employeeValues = {"Amy Alberts", "Development"};  
pptt.BindData(employeeValues, employeeColNames, "Employee", dataProps);  
DataTable teamData = GetCSVData((Page.MapPath("//data//Team.csv")));  
pptt.BindData(teamData, "Team", dataProps);  
//Code from Part 2  
DataTable dtCompanyHistory =  
GetCSVData(Page.MapPath("//data//CompanyHistory.csv"));  
dataProps.MaxRowsPerSlide = 5;  
pptt.BindData(dtCompanyHistory, "CompanyHistory", dataProps);  
DataTable dtProducts = GetCSVData(Page.MapPath("//data//Products.csv"));  
DataBindingProperties dataProps2 = pptt.CreateDataBindingProperties();  
dataProps2.MaxRowsPerSlide = 2;  
pptt.BindData(dtProducts, "Products", dataProps2);
```

Next, we will bind the data for the slides that were added in part 3.

5. Create an array containing the string "TrainingItem". Because all of our remaining slides use the same data markers, this array can be used as the column names array for multiple `BindData` calls.

```
string[] titleCol = { "TrainingItem" };
```

6. Get the data for the first slide in the 'Your Team' section of the presentation. The title of this slide is 'Team Members'.

```
//Data for the TrainingItem data source
object[] titleArrayTeam = { "Team Members" };
//Get the Team members data
DataTable dtTeamMembers = GetCSVData(Page.MapPath("//data//TeamMembers.csv"));
```

In order to keep the 'Team Members' data from populating all of our copied slides, we need to set the [Slide](#) property to scope to the slide we want. In this case, it is the 8th slide in our presentation.

7. Set the dataProps Slide property to be 7. Slide uses the 0-indexed position of the slide, so this will scope to slide 8.

```
dataProps.Slide = 7;
```

8. Bind the 'Team Members' data to the presentation. This will take two BindData calls, for the two data sources on the slide.

```
pptt.BindData(titleArrayTeam, titleCol, "Department", dataProps);
pptt.BindData(dtTeamMembers, "TrainingItems", dataProps);
```

9. Follow these same steps for the remaining slides. Make sure to change the Slide property before each BindData call. The same DataBindingProperties object can be used for all the BindData calls.

```
//The 9th slide will be 'Agile Development / Scrum'
object[] titleArrayAgile = { "Agile Development / Scrum" };
DataTable dtAgile = GetCSVData(Page.MapPath("//data//Agile.csv"));
dataProps.Slide = 8;
pptt.BindData(titleArrayAgile, titleCol, "Department", dataProps);
pptt.BindData(dtAgile, "TrainingItems", dataProps);

//The 10th slide will be 'Resources and Utilities'
object[] titleArrayResources = { "Resources and Utilities" };
DataTable dtResources = GetCSVData(Page.MapPath("//data//Resources.csv"));
dataProps.Slide = 9;
pptt.BindData(titleArrayResources, titleCol, "Department", dataProps);
pptt.BindData(dtResources, "TrainingItems", dataProps);

//The final slide will be 'Expectations'
object[] titleArrayExpectations = { "Expectations" };
DataTable dtExpectations = GetCSVData(Page.MapPath("//data//Expectations.csv"));
dataProps.Slide = 10;
pptt.BindData(titleArrayExpectations, titleCol, "Department", dataProps);
pptt.BindData(dtExpectations, "TrainingItems", dataProps);
```

10. Finally, Process and Save the finished presentation. Congratulations, the tutorial is complete!

```
pptt.Process();
pptt.Save(Page.Response, "part3_OutPut.pptx", false);
```

## Final Code

---



For information on writing this code, see [Part 1 - Importing Data](#) and [Part 2 - The Continue Modifier](#).

```
PowerPointApplication ppta = new PowerPointApplication();
Presentation pres = ppta.Open(Page.MapPath("~/templates/part3_template.pptx"));
Slide slideToCopy = pres.Slides[7];
pres.Slides.CopySlide(slideToCopy, (pres.Slides.Count));
pres.Slides.CopySlide(slideToCopy, (pres.Slides.Count));
pres.Slides.CopySlide(slideToCopy, (pres.Slides.Count));
PowerPointTemplate pptt = new PowerPointTemplate();
pptt.Open(ppta, pres);
//Code from Part 1
    DataBindingProperties dataProps = pptt.CreateDataBindingProperties();
    string[] columnNamesArray = { "Logo" };
    Byte[] logoArray = System.IO.File.ReadAllBytes((Page.MapPath("~/data/logo.png")));
    object[] valuesArray = { logoArray };
    pptt.BindData(valuesArray, columnNamesArray, "Company", dataProps);
    string[] employeeColNames = { "Name", "Team" };
    object[] employeeValues = { "Amy Alberts", "Development" };
    pptt.BindData(employeeValues, employeeColNames, "Employee", dataProps);
    DataTable teamData = GetCSVData((Page.MapPath("~/data/Team.csv")));
    pptt.BindData(teamData, "Team", dataProps);
//Code from Part 2
    DataTable dtCompanyHistory = GetCSVData(Page.MapPath("~/data/CompanyHistory.csv"));
    dataProps.MaxRowsPerSlide = 5;
    pptt.BindData(dtCompanyHistory, "CompanyHistory", dataProps);
    DataTable dtProducts = GetCSVData(Page.MapPath("~/data/Products.csv"));
    DataBindingProperties dataProps2 = pptt.CreateDataBindingProperties();
    dataProps2.MaxRowsPerSlide = 2;
    pptt.BindData(dtProducts, "Products", dataProps2);
//add the data from part 3
    string[] titleCol = { "TrainingItem" };
    object[] titleArrayTeam = { "Team Members" };
    DataTable dtTeamMembers = GetCSVData(Page.MapPath("~/data/TeamMembers.csv"));
    dataProps.Slide = 7;
    pptt.BindData(titleArrayTeam, titleCol, "Department", dataProps);
    pptt.BindData(dtTeamMembers, "TrainingItems", dataProps);
    object[] titleArrayAgile = { "Agile Development / Scrum" };
    DataTable dtAgile = GetCSVData(Page.MapPath("~/data/Agile.csv"));
    dataProps.Slide = 8;
    pptt.BindData(titleArrayAgile, titleCol, "Department", dataProps);
    pptt.BindData(dtAgile, "TrainingItems", dataProps);
    object[] titleArrayResources = { "Resources and Utilities" };
    DataTable dtResources = GetCSVData(Page.MapPath("~/data/Resources.csv"));
    dataProps.Slide = 9;
    pptt.BindData(titleArrayResources, titleCol, "Department", dataProps);
    pptt.BindData(dtResources, "TrainingItems", dataProps);
    object[] titleArrayExpectations = { "Expectations" };
    DataTable dtExpectations = GetCSVData(Page.MapPath("~/data/Expectations.csv"));
    dataProps.Slide = 10;
    pptt.BindData(titleArrayExpectations, titleCol, "Department", dataProps);
    pptt.BindData(dtExpectations, "TrainingItems", dataProps);
    pptt.Process();
    pptt.Save(Page.Response, "part3_OutPut.pptx", false);
```

## Downloads

You can download the code for the Basic PowerPointWriter Tutorials as a Visual Studio solution, which includes the Project Proposal.

- [PowerPointWriter\\_BasicTutorials.zip](#)

## Next Steps

See our [Quick Guides](#) and [Samples](#) for more advanced topics.