

# Fitting Data on to Multiple Slides

## Introduction

In order to ensure that data sets with multiple rows fit neatly in a PowerPoint presentation, PowerPointWriter allows for customization of the way data is imported.

This guide will explain how to use the various PowerPointWriter options for fitting data on to multiple slides.

This guide assumes a basic knowledge of [Data Markers](#) and [Importing Multiple Rows of Data](#) into a presentation.

- [Introduction](#)
- [MaxRowsPerSlide](#)
- [The Continue Modifier](#)
- [RepeatSlide](#)
- [Special Cases](#)
  - [Repeating an entire slide for each row of data](#)
  - [Repeating a slide that already has the continue modifier](#)

## MaxRowsPerSlide

The [MaxRowsPerSlide](#) property allows the user to control how many rows of data are imported into any slide. Note that when importing multiple row of data, the data marker must be placed in a list entry or table row. For more information see [Importing Multiple Rows of Data](#).

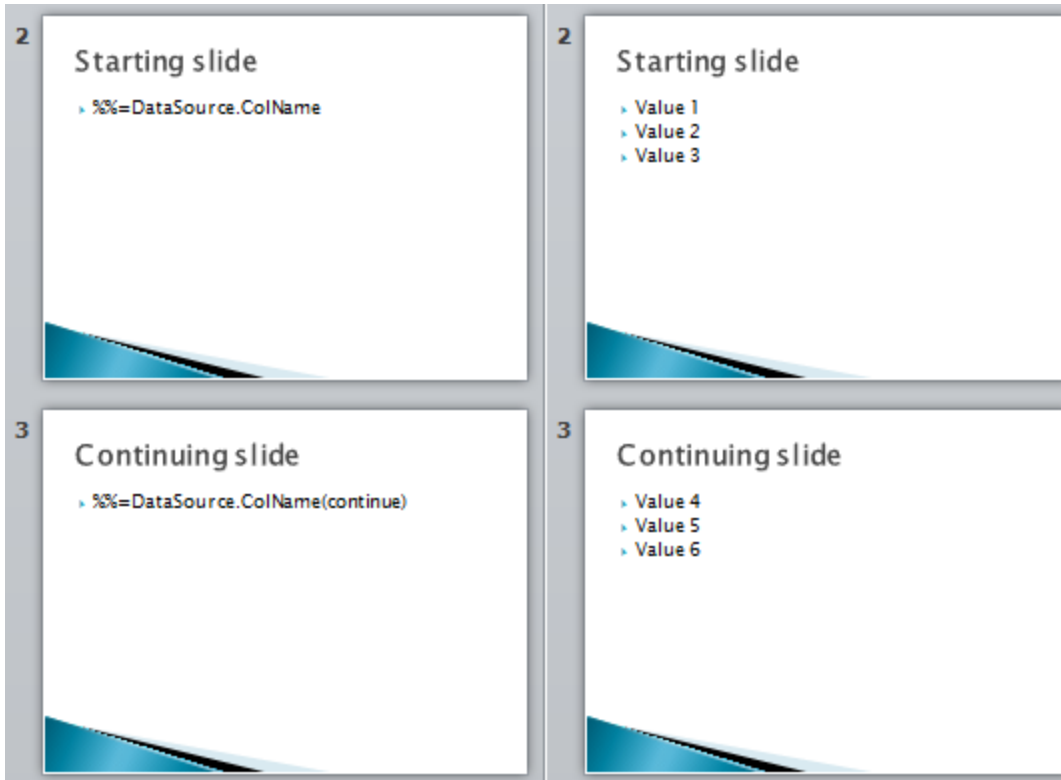
```
DataBindingProperties DataProps = pptt.DataBindingProperties();  
DataProps.MaxRowsPerSlide = 5;
```

When used by itself, MaxRowsPerSlide will stop importing data when the initial row limit has been reached, and no more data from that data set will be imported into the presentation. For example, if your data set has 12 rows of data, but MaxRowsPerSlide is set to 5, PowerPointWriter will import 5 rows of data initially. To import all of the data in a data set, MaxRowsPerSlide must be used with the *Continue* modifier or *RepeatSlide* marker.

## The Continue Modifier

The *Continue* modifier allows data from a single data source to be imported across multiple slides. When data is being imported into a data source, and the MaxRowsPerSlide limit is reached, PowerPointWriter will look in the presentation for the next copy of the same data marker, but with the *Continue* modifier. It will then begin importing data at the new location, starting with the next row of data in the data source.

The *Continue* modifier should be placed in parentheses at the end of a data marker: `%%=DataSource.DataCol(continue)`. The data marker with the *Continue* modifier must also be placed in a list or table to enable PowerPointWriter's automatic repeating behavior. If it is not in a list or table, only 1 additional row of data will be imported.



If a presentation contains multiple appearances of the same data marker and the *Continue* modifier is not used, PowerPointWriter will begin importing data from the first row in the data source for each instance of the data marker.

The continue modifier can also be used to import multiple batches of data onto a single slide:

<ul style="list-style-type: none"> <li>• %%=RepeatingData2.List : %%=RepeatingData2.List2</li> </ul>	<ul style="list-style-type: none"> <li>• %%=RepeatingData2.List(continue) : %%=RepeatingData2.List2(continue)</li> </ul>
--	--

<ul style="list-style-type: none"> <li>• Connecticut : 3.6 million residents</li> <li>• Maine : 1.3 million residents</li> <li>• Massachusetts : 6.6 million residents</li> </ul>	<ul style="list-style-type: none"> <li>• New Hampshire : 1.3 million residents</li> <li>• Rhode Island : 1 million residents</li> <li>• Vermont : 0.6 million residents</li> </ul>
---	--

PowerPointWriter will attempt to import all the data, but there must be enough data markers with *Continue* modifiers to accommodate all the data, or data rows will be lost. For example, if you have 12 rows of data and MaxRowsPerSlide is set to 5, then you need the original data marker and two additional data markers with the *Continue* modifier to import all 12 rows (5 rows, 5 rows, 2 rows).

If you are unsure of how many rows to import, we recommend that you use the *RepeatSlide* marker.

## RepeatSlide

PowerPoint's RepeatSlide marker makes it easy to import data onto multiple slides with the same layout. With RepeatSlide it is not necessary to know the size of the data set you are importing.

On a slide marked with the RepeatSlide marker, PowerPointWriter will import data onto the slide until the MaxRowsPerSlide limit is reached. Next, PowerPointWriter will create a copy of the slide and insert the copy into the presentation in place. All of the data markers on the copied slide are treated as though they have the *Continue* modifier applied.

Then using the regular *Continue* modifier behavior, PowerPointWriter will search for the next instance of the data marker with the *Continue* modifier. This will of course be the copied slide that was just created and PowerPointWriter will continue to import data onto the new slide. This behavior will continue until all the data has been imported.

To use RepeatSlide:

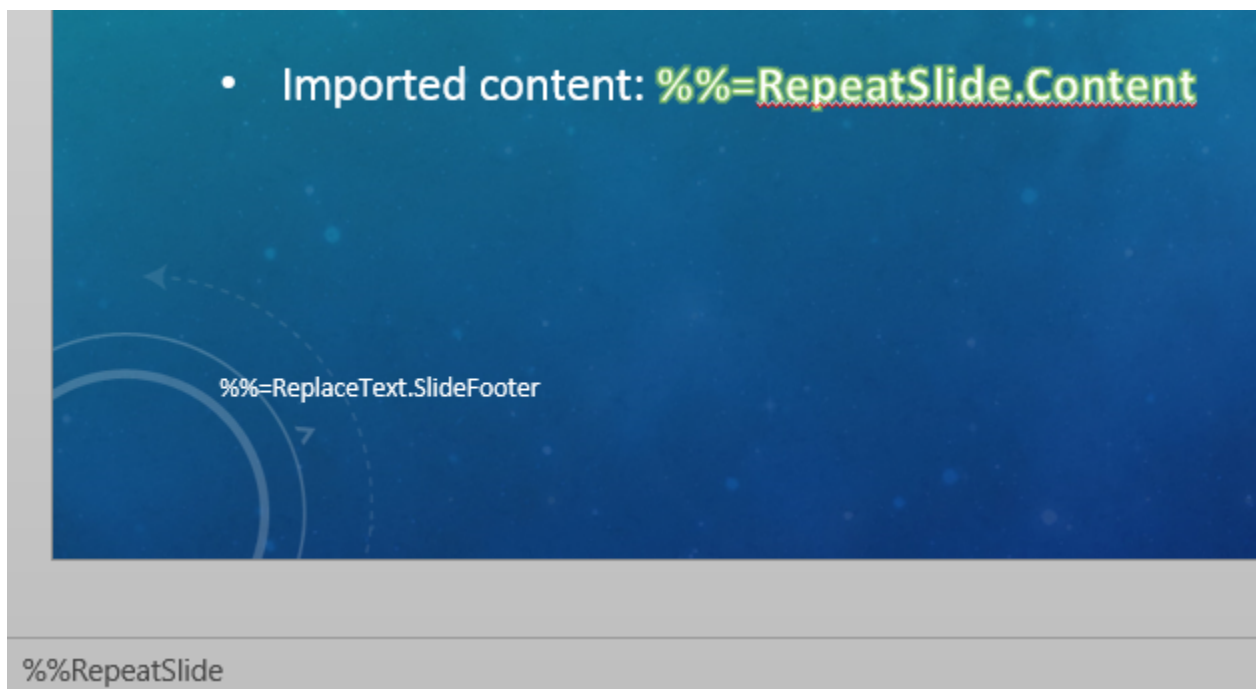
1. Make sure that MaxRowsPerSlide is set to avoid importing all the data at once
2. Place the marker **%%RepeatSlide** as the first item in the notes section of the slide you wish to see repeated. There must be a space between the RepeatSlide marker and the rest of the text in the notes section, but you can have additional text in the notes section.

## Special Cases

### Repeating an entire slide for each row of data

To achieve mail merge-like behavior:

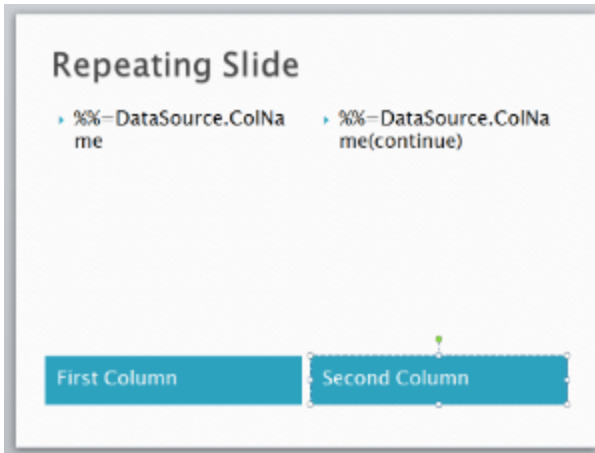
1. Make sure that MaxRowsPerSlide is set to 1
2. Add %%RepeatSlide to the slide you want to repeat
3. Add data markers: the data markers do not need to be in a list entry or table row.



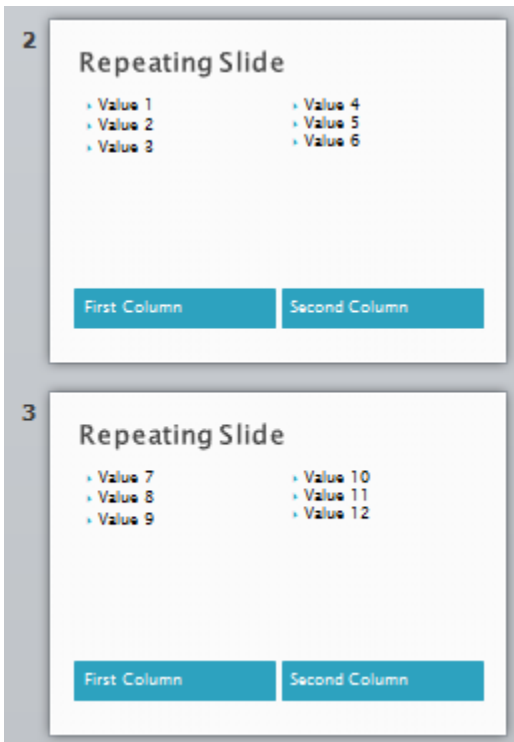
### Repeating a slide that already has the continue modifier

PowerPointWriter will always attempt to import data at the next available data marker with the *Continue* modifier.

When the *RepeatSlide* behavior is engaged, it makes a copy of the repeating slide and sets all the data markers to use the *Continue* modifier. If you are importing multiple times on the same slide, this behavior will continue for each repeated slide:

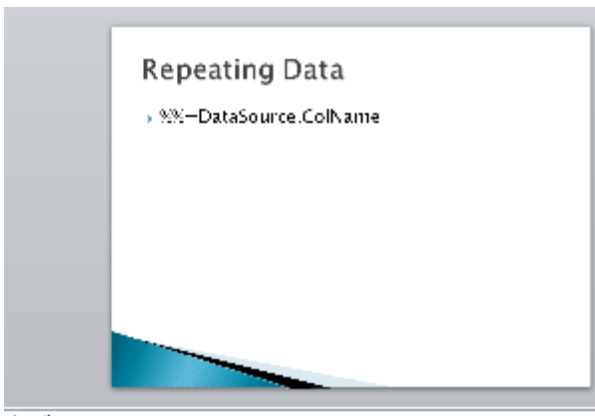


Output file:



You can also start by importing data on a non-repeating slide and then continue on a slide that is set to repeat:

First slide (not set to repeat)



Second slide (set to repeat):

## Repeating Data(cont.)

- ▶ `%%=DataSource.ColName(continue)`