



PDF SHARE FORMS

Online, Offline, OnDemand

PDF forms and SharePoint are better together

Mapping repeatable fields to SharePoint list

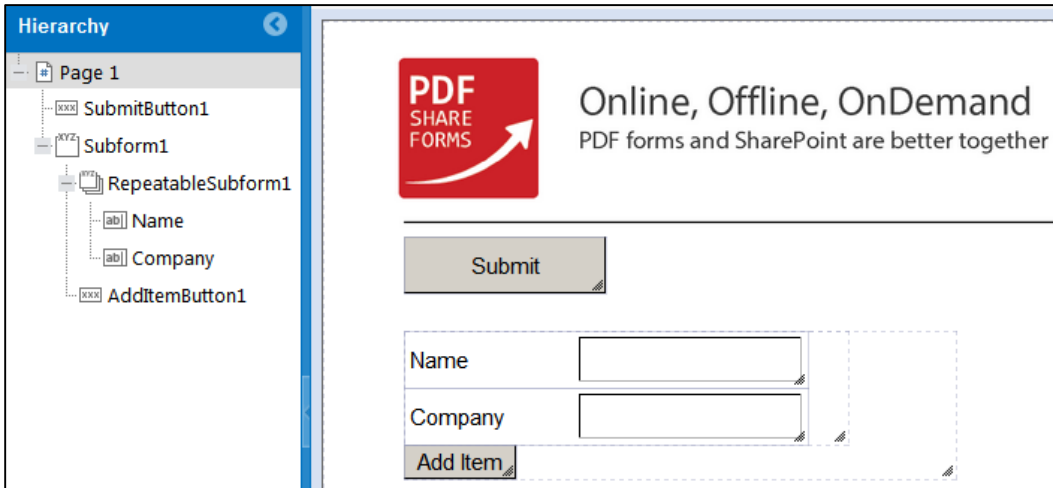
Contents

PDF Share Forms Enterprise for SharePoint 2013	2
Saving information from Repeatable section to SharePoint list.....	2
Loading information from SharePoint list to Repeatable section	4
PDF Share Forms Enterprise for SharePoint 2010	6
Saving information from Repeatable section to SharePoint list.....	6
Loading information from SharePoint list to Repeatable section	8

PDF Share Forms Enterprise for SharePoint 2013

This guide will show how to save data from Repeatable section of PDF form to SharePoint List and load it back from the list to Repeatable section.

Step 1. Template

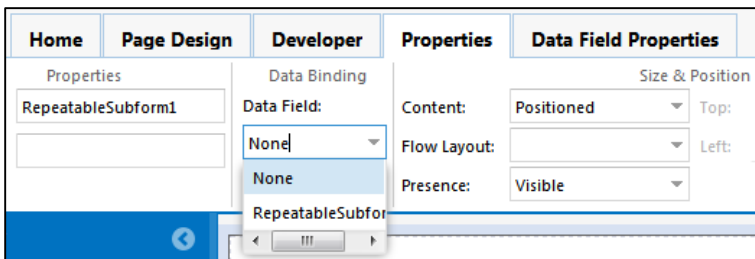


This template has repeatable sub form and two text fields in it. Those two text fields will be repeated using “Add Item” button.

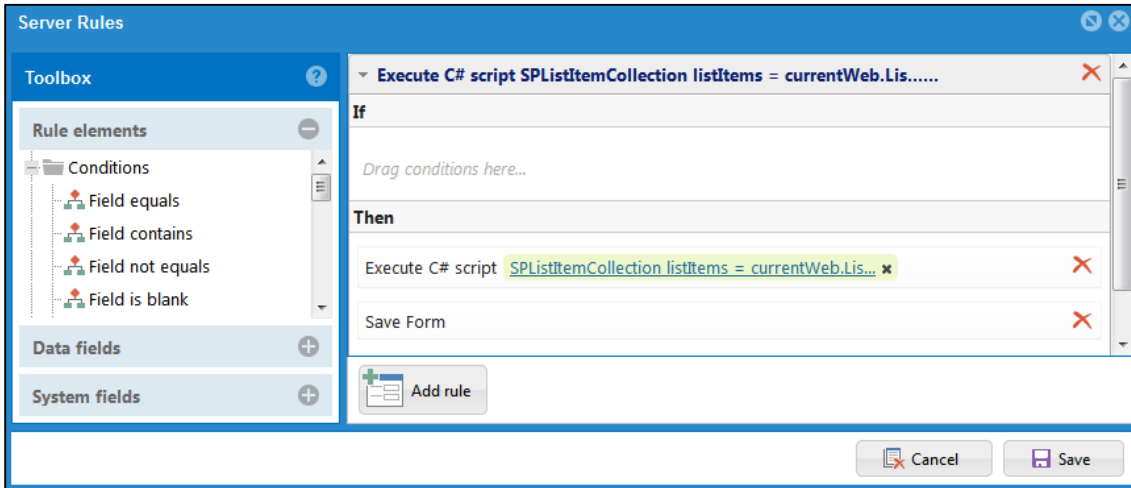
NOTE: Repeatable Subform must be placed in flawed Subform field.

Saving information from Repeatable section to SharePoint list

Step 2. Remove **RepeatableSubform1** Data Binding. Choose **RepeatableSubform1** → **Properties** → **Data Binding**



Step 3. Add action to Form Submit event. Navigate to **Developer** → **Form Submit**



Add **“Execute C# script”** action above **“Save Form”** action and place inside it following code:

```
SPListItemCollection listItems =currentWeb.Lists["TestRepeatable"].Items; //name must be specified
foreach (XElement node in data.Root.Elements("RepeatableSubform1")) {
SPListItem item = listItems.Add();
item["Title"] = node.Element("Name").Value; //field name and column must be specified
item["Company"] = node.Element("Company").Value;
item.Update(); }
```

This code will take data from all appeared text fields to specified list fields of **“TestRepeatable”** list.

NOTE:

1. List will not be created automatically, in this guide was used custom list **“TestRepeatable”** with following columns:
 - Name – **“single line of text”** column, made from **“Title”** column, will be used to store data from **“Name”** text field;
 - Company – **“single line of text”** column, will be used to store data from **“Company”** text field.
2. **“Name”** column has internal name **“Title”**, that’s why in script we are placing it instead of **“Name”**:

```
item["Title"] = node.Element("Name").Value;
```

To check internal name of the list’s column proceed to **List** → **List Settings** → **Columns** → **Edit column**:

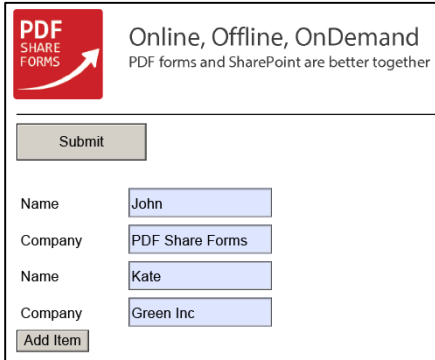
Internal name will be at the end of the URL of edit column page. In this guide URL for **“Name”** column is following:

http://yourSite/_layouts/FldEdit.aspx?List={4A1BDE6C-5640-41F9-BDAC-F88BA89212D6}&Field=Title

which means that **“Title”** is its internal name.

Step 4. Save template and deploy.

Step 5.1. Open and fill-in a form



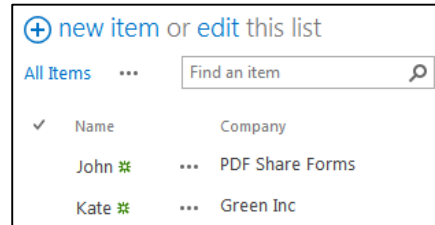
PDF SHARE FORMS Online, Offline, OnDemand
PDF forms and SharePoint are better together

Submit

Name: John
Company: PDF Share Forms
Name: Kate
Company: Green Inc
Add Item

Step 5.2. Click on save button to submit form.

Step 5.3. Open “TestRepeatable” list



+ new item or edit this list

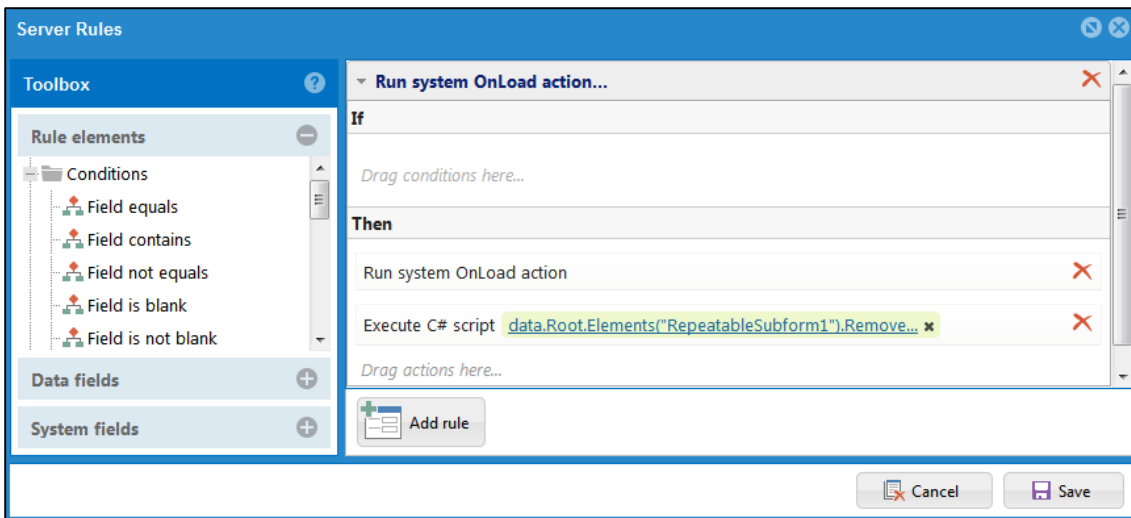
All Items Find an item

Name	Company
John	PDF Share Forms
Kate	Green Inc

List contains the information that was added in PDF form.

Loading information from SharePoint list to Repeatable section

Step 6. Add a server-side script to Form Load event. Navigate to Developer → Form Load



Server Rules

Toolbox

- Rule elements
 - Conditions
 - Field equals
 - Field contains
 - Field not equals
 - Field is blank
 - Field is not blank
 - Data fields
 - System fields

Run system OnLoad action...

If

Drag conditions here...

Then

Run system OnLoad action

Execute C# script `data.Root.Elements("RepeatableSubform1").Remove...`

Drag actions here...

Add rule

Cancel Save

Place “Execute script” under “Run system OnLoad action”, add this code in C# script action

```
data.Root.Elements("RepeatableSubform1").Remove();
SPList list = currentWeb.Lists["TestRepeatable"]; //list name should be specified
SPListItemCollection items = list.GetItems("Title", "Company"); //column should be specified
foreach(SPListItem item in items){
data.Root.Add(new XElement("RepeatableSubform1", //field names must be specified
    new XElement("Name", item["Title"]),
    new XElement("Company", item["Company"]))); }
```

This code will add columns from the “TestRepeatable” list to form repeatable section’s fields.

NOTE: here also is used “Name” column with internal name “Title”.



Step 7. Save template and deploy

Step 8. Open “TestRepeatable” list

List already has two items: John PDF Share Forms and Kate Green Inc, which were added on step 5. Add one more item using “**Add new item**” (in this example Anna Contoso)

✓	Name	Company
	John	PDF Share Forms
	Kate	Green Inc
	Anna	Contoso

Step 9. Runtime

Submit

Name: John
Company: PDF Share Forms

Name: Kate
Company: Green Inc

Name: Anna
Company: Contoso

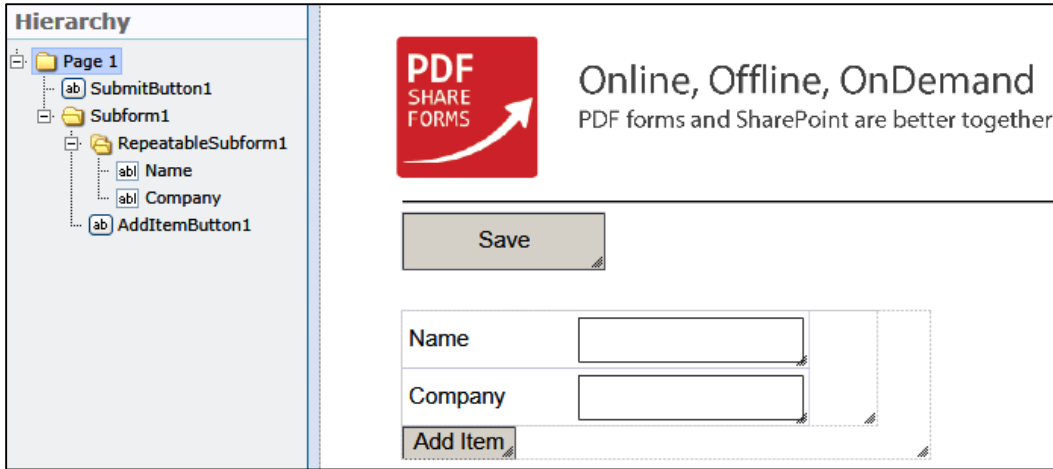
Add Item

New form was populated with data from “TestRepeatable” list.

PDF Share Forms Enterprise for SharePoint 2010

This guide will show how to save data from Repeatable section of PDF form to SharePoint List and load it back from the list to Repeatable section.

Step 1. Template

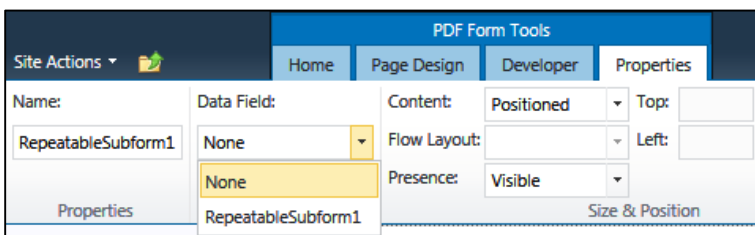


This template has repeatable sub form and two text fields in it. Those two text fields will be repeated using “Add Item” button.

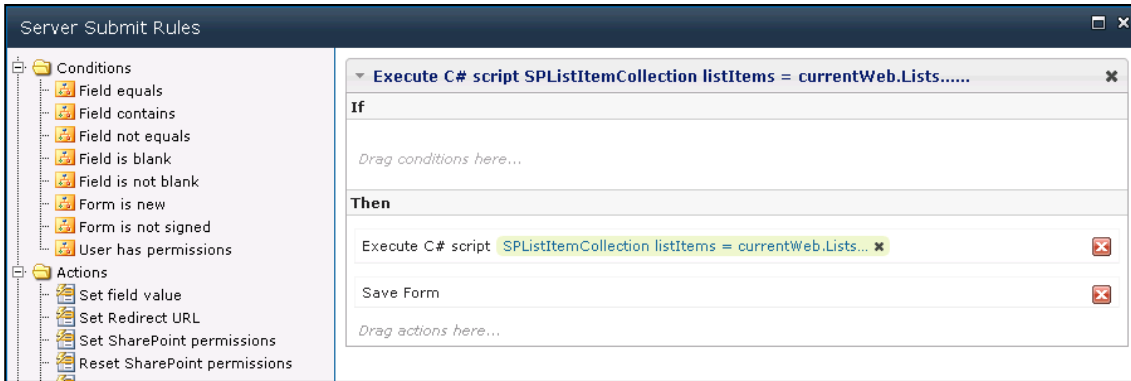
NOTE: Repeatable Subform must be placed in flawed Subform field.

Saving information from Repeatable section to SharePoint list

Step 2. Remove **RepeatablSubform1** Data Binding. Choose **RepeatablSubform1** → **Properties** → **Data Binding**



Step 3. Add action to Form Submit event. Navigate to **PDF Form Tools → Developer → Form Submit**



Add **“Execute script”** action above **“Save Form”** action and place inside it following code:

```
SPListItemCollection listItems=currentWeb.Lists["TestRepeatable"].Items; //name must be specified
foreach (XElement node in data.Root.Elements("RepeatableSubform1"))
{
    SPListItem item = listItems.Add();
    item["Title"] = node.Element("Name").Value; //field name and column must be specified
    item["Company"] = node.Element("Company").Value;
    item.Update();
}
```

This code will take data from all appeared text fields to specified list fields of **“TestRepeatable”** list.

NOTE:

1. List will not be created automatically, in this guide was used custom list **“TestRepeatable”** with following columns:
 - Name – **“single line of text”** column, made from **“Title”** column, will be used to store data from **“Name”** text field;
 - Company – **“single line of text”** column, will be used to store data from **“Company”** text field.
2. **“Name”** column has internal name **“Title”**, that’s why in script we are placing it instead of **“Name”**:


```
item["Title"] = node.Element("Name").Value;
```

To check internal name of the list’s column proceed to **List → List Settings → Columns → Edit column**:

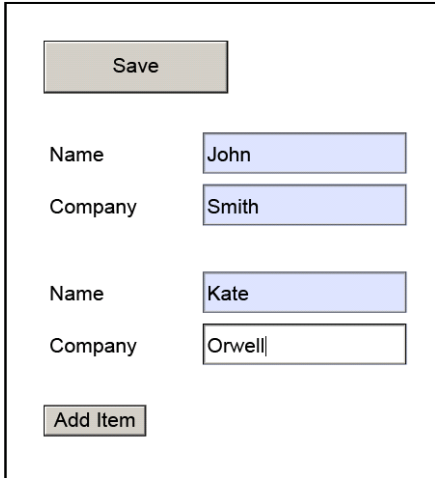
Internal name will be at the end of the URL of edit column page. In this guide URL for **“Name”** column is following:

http://yourSite/_layouts/FldEdit.aspx?List={4A1BDE6C-5640-41F9-BDAC-F88BA89212D6}&Field=Title

Which means that **“Title”** is its internal name.

Step 4. Save template and deploy

Step 5.1. Open PDF form inside your library:



Step 5.2. Click on save button to submit form.

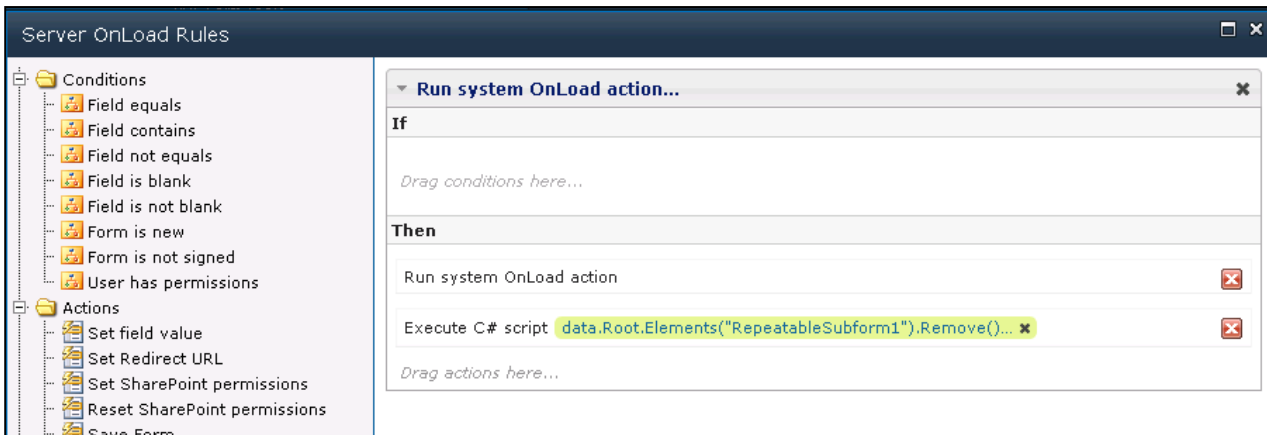
Step 5.3. Open TestRepeatable list:

	Name	Company
	John <small>NEW</small>	Smith
	Kate <small>NEW</small>	Orwell
+ Add new item		

List contains the information that was added in PDF form.

Loading information from SharePoint list to Repeatable section

Step 6. Add a server-side script to Form Load event. Navigate to **PDF Form Tools → Developer → Form Load**



Place **“Execute script”** under **“Run system OnLoad action”**, add this code in C# script action:

```
data.Root.Elements("RepeatableSubform1").Remove(); //column and field names must be specified
SPList list = currentWeb.Lists["TestRepeatable"];
SPListItemCollection items = list.GetItems("Title", "Company");
foreach(SPListItem item in items){
    data.Root.Add(new XElement("RepeatableSubform1",
        new XElement("Name", item["Title"]),
        new XElement("Company", item["Company"]))); }
```

This code will add two selected columns from the list **“TestRepeatable”** to form repeatable section’s fields.

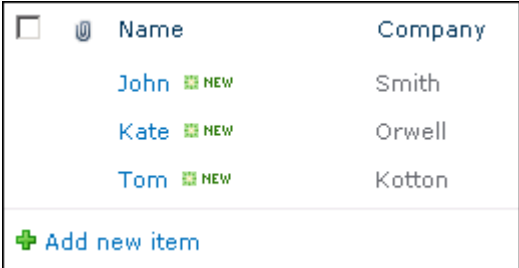
NOTE: here also was used **“Name”** column with internal name **“Title”**.

Step 7. Save template and deploy

Step 8. Open “TestRepeatable” list

List already has two items: John Smith and Kate Orwell, which were added on step 5.

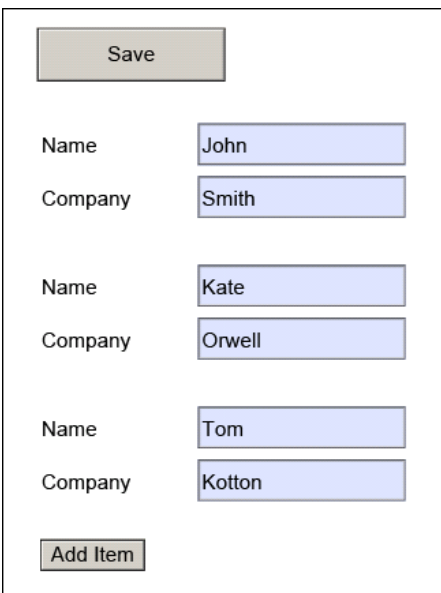
Add one more item using “Add new item” (in this example Tom Kotton)



	Name	Company
	John <small>NEW</small>	Smith
	Kate <small>NEW</small>	Orwell
	Tom <small>NEW</small>	Kotton

[+ Add new item](#)

Step 9. Create new PDF form from template you deployed on step 6



Save

Name: John
Company: Smith

Name: Kate
Company: Orwell

Name: Tom
Company: Kotton

Add Item

New form was populated with data from “TestRepeatable” list.