



PDF SHARE FORMS

Online, Offline, OnDemand

PDF forms and SharePoint are better together

Populating data to fields using BCS

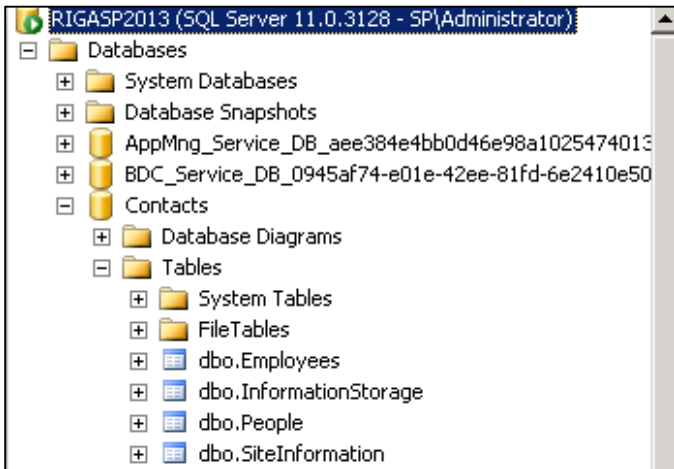
Contents

Prepare SQL tables.....	2
Create External Content Type.....	3
Populating data to fields using BCS data source with Scan action	8
Populating data to fields using BCS data source without scanning.....	17
Form Submit script for case with Scan.....	26
On Form Load script for case with Scan.....	27
On Form Submit script for case without Scan	29

This guide describe how to populate form by using Business Connectivity Services.

Prepare SQL tables

This structure is being used
 (“People” table is not being used):



Employees table (this table store information about Employees and it will be populated in EmployeePIN dropdown and fields connected with this dropdown):

```
CREATE TABLE [dbo].[Employees](
  [ID] [int] IDENTITY(1,1) NOT NULL,
  [EmployeePIN] [nvarchar](15) NULL,
  [QueryReferenceNumber] [nvarchar](15) NULL,
  [FirstName] [nchar](20) NULL,
  [LastName] [nchar](20) NULL
) ON [PRIMARY]
```

InformationStorage table (there will be stored information after form submit):

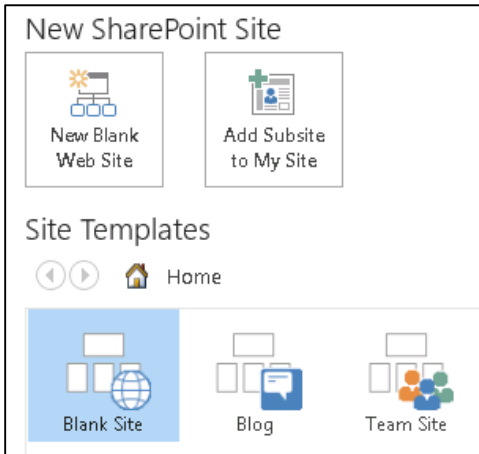
```
CREATE TABLE [dbo].[InformationStorage](
  [ID] [int] IDENTITY(1,1) NOT NULL,
  [EmployeePIN] [nvarchar](15) NULL,
  [QueryReferenceNumber] [nvarchar](15) NULL,
  [FirstName] [nchar](20) NULL,
  [LastName] [nchar](20) NULL,
  [SITECode] [nvarchar](15) NULL,
  [SITEName] [nvarchar](20) NULL,
  [PayrollWeekQueried] [int] NULL,
  [ActionTaken] [text] NULL,
  [Results] [text] NULL,
  [Approvalment] [nchar](15) NULL
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

SiteInformation table (this table store information which will be populated in SITE Code dropdown and fields connected with this dropdown)

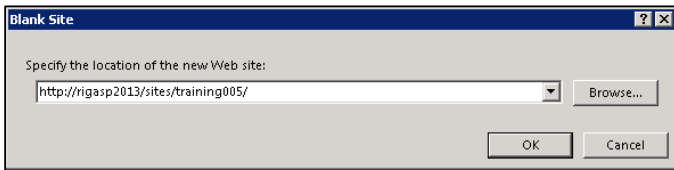
```
CREATE TABLE [dbo].[SiteInformation](
  [ID] [int] IDENTITY(1,1) NOT NULL,
  [SITECode] [nvarchar](15) NOT NULL,
  [SITEName] [nvarchar](20) NOT NULL
) ON [PRIMARY]
```

Create External Content Type

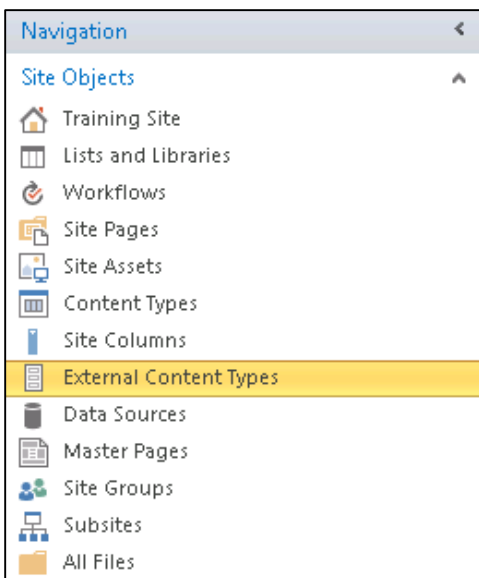
Step 1. Go to SharePoint Designer and create New SharePoint Site



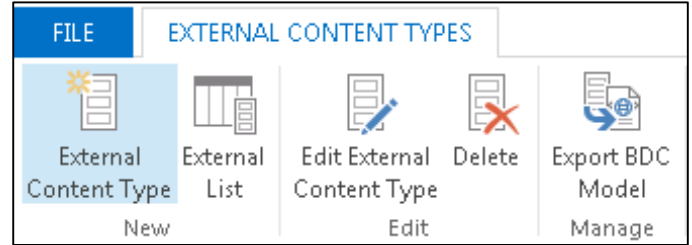
For example (rigasp2013 is server name):



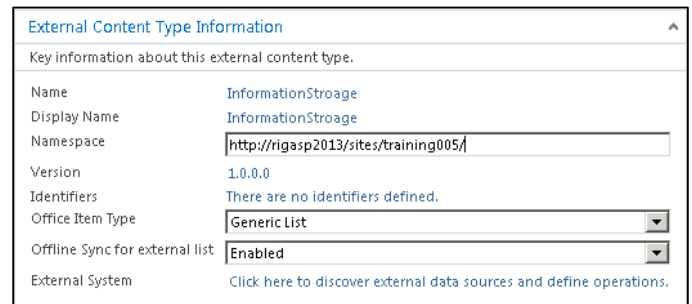
Step 2. Go to **Navigation** → **External Content Types** to create new **External Content Type**



Click on **External Content Type**:

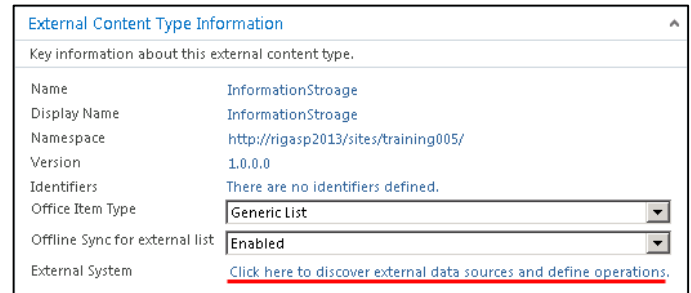


Specify **Name** and **Display Name**:

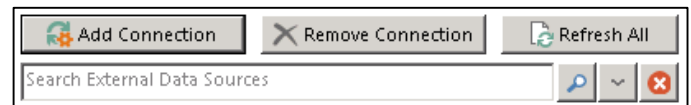


Specify **External System**:

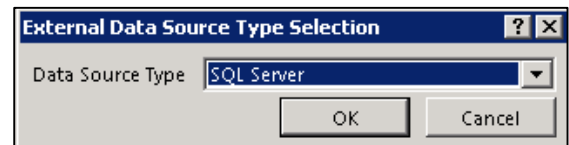
- Define operations



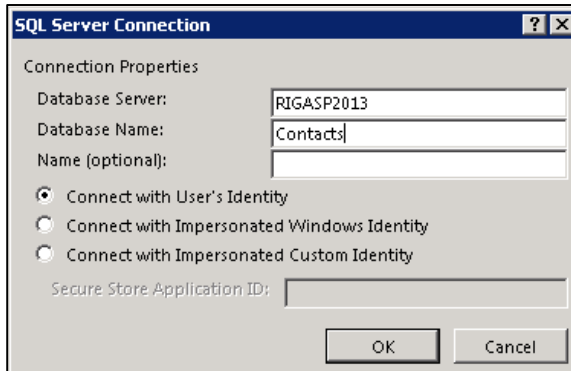
- Click **Add Connection**



- Choose **SQL Server**

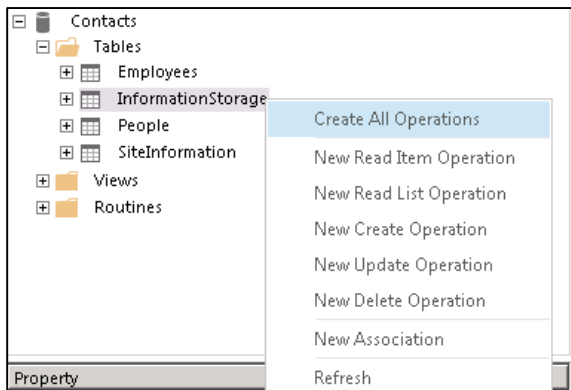


- Write DB Server and DB Name

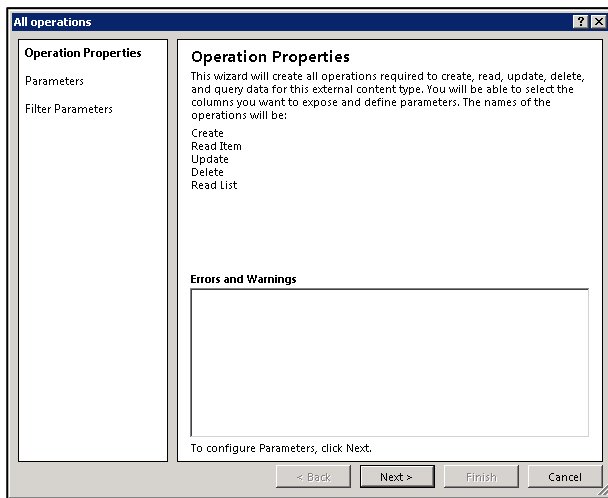


Step 3. Specify actions available with ECT:

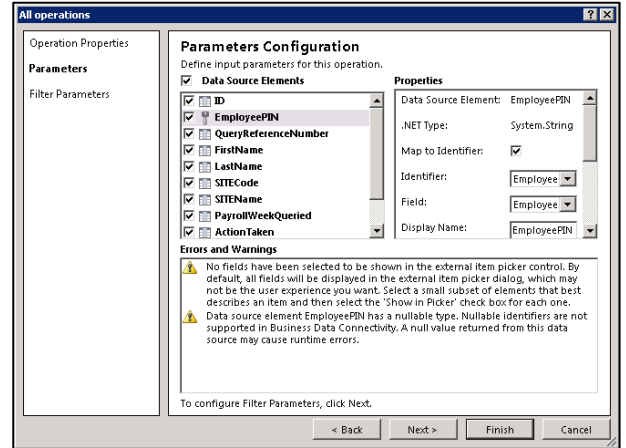
- Choose proper SQL table and create all operations:



- Click Next



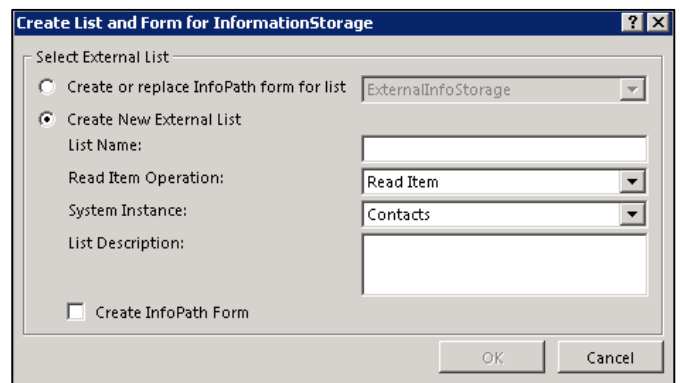
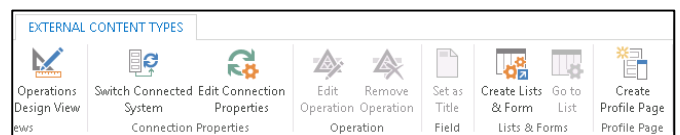
- Configure Identifier and choose fields that will be used with earlier mentioned actions.



In this example identifier is Employee PIN so all other information will stick to it. So when EmployeePIN will be called in the form, all required information will be taken from list depending on selected EmployeePIN.

Then click Finish.

Step 4. Create External List. Navigate to External Content Types → Create Lists & Forms



Step 5. Add permissions to this list:

Navigate to **Central Administrator → Manage Service Applications → BDCS:**

BDCS	Business Data Connectivity Service Application
BDCS	Business Data Connectivity Service Application Proxy
Excel Services Application	Excel Services Application Web Service Application
Excel Services Application	Excel Services Application Web Service Application Proxy

Choose proper List:

<input checked="" type="checkbox"/>	InformationStorage	InformationStorage	http://rigasp2013/sites/training005	1.0.0.0	Contacts
<input type="checkbox"/>	InformationStorage	InformationStorage	http://rigasp2013/sites/training007	1.0.0.0	Contacts
<input type="checkbox"/>	SiteInfo	SiteInfo	http://rigasp2013/sites/training005	1.0.0.0	Contacts
<input type="checkbox"/>	SiteInfo	SiteInfo	http://rigasp2013/sites/training007	1.0.0.0	Contacts

Navigate to **Edit → Set Object Permissions:**

Set Object Permissions ✕

To add an account, or group, type or select it below and click 'Add'.

i:\0#w\sp\administrator

Training User 005

To remove an account, or group, select it above and click 'Remove'.

Permissions for Training User 005:

Edit	<input checked="" type="checkbox"/>
Execute	<input checked="" type="checkbox"/>
Selectable In Clients	<input checked="" type="checkbox"/>
Set Permissions	<input checked="" type="checkbox"/>

Propagate permissions to all methods of this external content type. Doing so will overwrite existing permissions.

Allow user to Edit, Execute, Selectable and Clients and Set Permissions.

Repeat all the same steps with all other tables.



Results

- for **InformationStorage** External Content Type:

External Content Type Information Key information about this external content type.			Permissions Permissions for this external content type.																										
Name	InformationStorage		Name	Permissions																									
Display Name	InformationStorage		sp\administrator	Edit, Execute, Set Permissions, Selectable In Clients																									
Namespace	http://rigasp2013/sites/training005		sp/training005	Edit, Execute, Set Permissions, Selectable In Clients																									
Version	1.0.0.0		External Lists																										
Identifiers	ID[Int32]		View and navigate to external lists bound to this external content type.																										
Office Item Type	Generic List		Name	URL																									
Offline Sync for external list	Enabled		ExternalInfoStorage	/sites/training005/Lists/ExternalInfoStorage/Read List.aspx																									
External System	Contacts		Abra	/sites/training005/Lists/Abra/Read List.aspx																									
External Content Type Operations Use this part to manage the operations of this external content type.			Fields A list of the fields that were defined in the operations of this external content type.																										
✓ This external content type has read, write, and search capabilities. You may associate it with other external content types by creating an Association operation from the Operations Design View.			<table border="1"> <thead> <tr> <th>Field Name</th> <th>Type</th> </tr> </thead> <tbody> <tr><td>EmployeePIN</td><td>System.String</td></tr> <tr><td>QueryReferenceNumber</td><td>System.String</td></tr> <tr><td>FirstName</td><td>System.String</td></tr> <tr><td>LastName</td><td>System.String</td></tr> <tr><td>SITECode</td><td>System.String</td></tr> <tr><td>SITENAME</td><td>System.String</td></tr> <tr><td>PayrollWeekQueried</td><td>System.Nullable<System.Int32></td></tr> <tr><td>ActionTaken</td><td>System.String</td></tr> <tr><td>Results</td><td>System.String</td></tr> <tr><td>Approvalment</td><td>System.String</td></tr> <tr><td>ID</td><td>System.Int32</td></tr> </tbody> </table>			Field Name	Type	EmployeePIN	System.String	QueryReferenceNumber	System.String	FirstName	System.String	LastName	System.String	SITECode	System.String	SITENAME	System.String	PayrollWeekQueried	System.Nullable<System.Int32>	ActionTaken	System.String	Results	System.String	Approvalment	System.String	ID	System.Int32
Field Name	Type																												
EmployeePIN	System.String																												
QueryReferenceNumber	System.String																												
FirstName	System.String																												
LastName	System.String																												
SITECode	System.String																												
SITENAME	System.String																												
PayrollWeekQueried	System.Nullable<System.Int32>																												
ActionTaken	System.String																												
Results	System.String																												
Approvalment	System.String																												
ID	System.Int32																												
Name	Type	Data Source Object																											
Create	Create	InformationStorage																											
Read Item	Read Item	InformationStorage																											
Update	Update	InformationStorage																											
Delete	Delete	InformationStorage																											
Read List	Read List	InformationStorage																											

- for **SiteInfo** External Content Type:

External Content Type Information Key information about this external content type.			Permissions Permissions for this external content type.										
Name	SiteInfo		Name	Permissions									
Display Name	SiteInfo		sp\administrator	Edit, Execute, Set Permissions, Selectable In Clients									
Namespace	http://rigasp2013/sites/training005		sp/training005	Edit, Execute, Set Permissions, Selectable In Clients									
Version	1.0.0.0		External Lists										
Identifiers	SITECode(String)		View and navigate to external lists bound to this external content type.										
Office Item Type	Generic List		Name	URL									
Offline Sync for external list	Enabled		ExternalSiteInfo	/sites/training005/Lists/ExternalSiteInfo/Read List.aspx									
External System	Contacts		Fields										
External Content Type Operations Use this part to manage the operations of this external content type.			A list of the fields that were defined in the operations of this external content type.										
✓ This external content type has read, write, and search capabilities. You may associate it with other external content types by creating an Association operation from the Operations Design View.			<table border="1"> <thead> <tr> <th>Field Name</th> <th>Type</th> </tr> </thead> <tbody> <tr><td>SITECode</td><td>System.String</td></tr> <tr><td>SITENAME</td><td>System.String</td></tr> <tr><td>ID</td><td>System.Int32</td></tr> </tbody> </table>			Field Name	Type	SITECode	System.String	SITENAME	System.String	ID	System.Int32
Field Name	Type												
SITECode	System.String												
SITENAME	System.String												
ID	System.Int32												
Name	Type	Data Source Object											
Create	Create	SiteInformation											
Read Item	Read Item	SiteInformation											
Update	Update	SiteInformation											
Delete	Delete	SiteInformation											
Read List	Read List	SiteInformation											

- for **WagesQuery** External Content Type:

External Content Type Information		
Key information about this external content type.		
Name	WagesQuery	
Display Name	WagesQuery	
Namespace	http://rigasp2013/sites/training007	
Version	1.0.0.0	
Identifiers	EmployeePIN(String)	
Office Item Type	Generic List	
Offline Sync for external list	Enabled	
External System	Contacts	

External Content Type Operations		
Use this part to manage the operations of this external content type.		
This external content type has read, write, and search capabilities. You may associate it with other external content types by creating an Association operation from the Operations Design View.		
Name	Type	Data Source Object
Create	Create	Employees
Read Item	Read Item	Employees
Update	Update	Employees
Delete	Delete	Employees
Read List	Read List	Employees

Permissions	
Permissions for this external content type.	
Name	Permissions
sp\administrator	Edit, Execute, Set Permissions, Selectable In Clients
sp\training007	Edit, Execute, Set Permissions, Selectable In Clients

External Lists	
View and navigate to external lists bound to this external content type.	
Name	URL

Fields	
A list of the fields that were defined in the operations of this external content type.	
Field Name	Type
EmployeePIN	System.String
QueryReferenceNumber	System.String
FirstName	System.String
LastName	System.String
ID	System.Int32

Now in the SharePoint will be three lists:

Lists

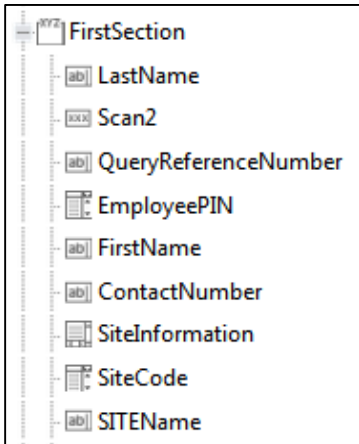
- ExternalEmployee
- ExternalSiteInfo
- ExternalInfoStorage
- Abra
- Home
- Documents
- Recent

NOTE: Names were given in **Step 3**.

Populating data to fields using BCS data source with Scan action

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.

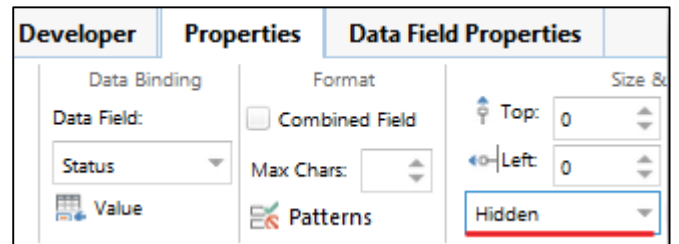
Step 1. Prepare template:



According to the **Employee PIN** dropdown's field value will be filled **Query Reference Number, First Name** and

Last Name fields. According to **SITE Code** dropdown's field value will be filled **SITE Name**. **Scan** button will start actions to fill text fields with text depending on dropdowns value by taking information from External Content Lists. **Employee PIN, Query Reference Number, First Name, Last Name, SITE Code, SITE Name, Payroll Week Queried and Area Manager** will be stored in **ExternalInfoStorage** List.

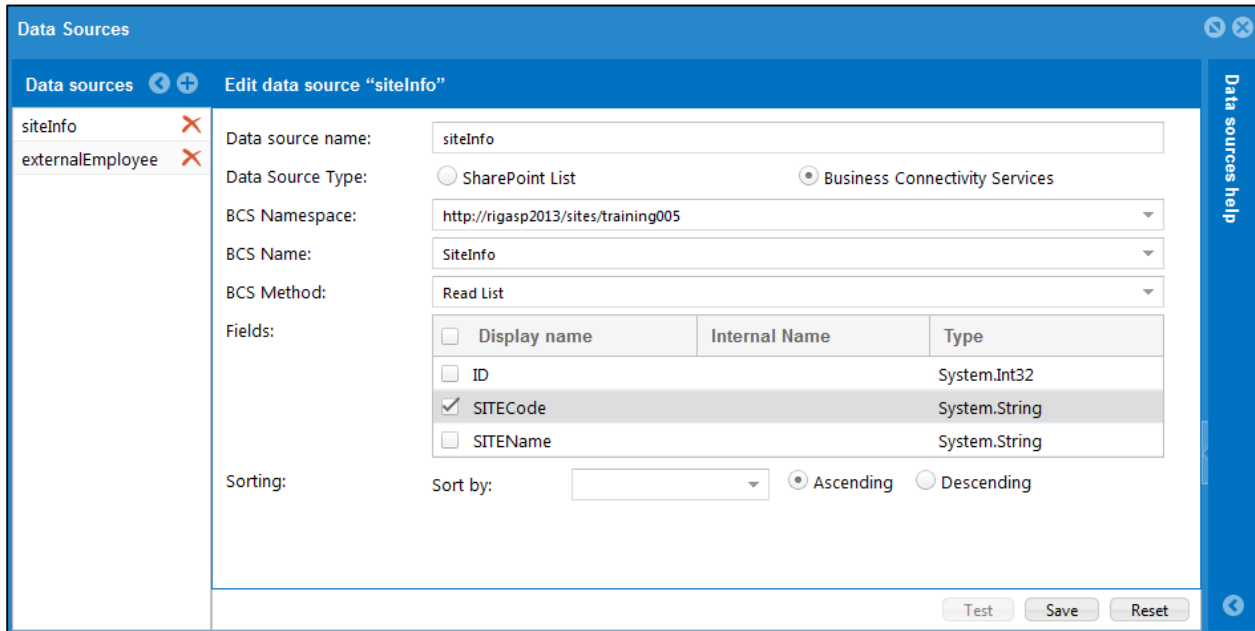
Status and **exStatus** fields need set to hidden: choose field and navigate to **PDF Form Tools** → **Properties**.



Wages Query Form			
EMPLOYEE DETAILS			
Employee PIN	<input type="text"/>	Scan	Query Reference Number <input type="text"/>
Title	<input type="text"/>		
First Name	<input type="text"/>	Last Name	<input type="text"/>
Email Address	<input type="text"/>		
Contact Number	<input type="text"/>	Other Contact Number	<input type="text"/>
SITE INFORMATION & QUERY			
SITE Code	<input type="text"/>	Scan	SITE Name <input type="text"/>
Payroll Week Queried	<input type="text"/>	Area Manager	<input type="text"/>
Query Details	<input type="text"/>		
Action Required By	<input type="text"/>		
			SUBMIT

Step 2. Create additional Data Sources with Business Connectivity Services type
PDF Form Tools → Developer → Data Sources

- **SiteInfo**

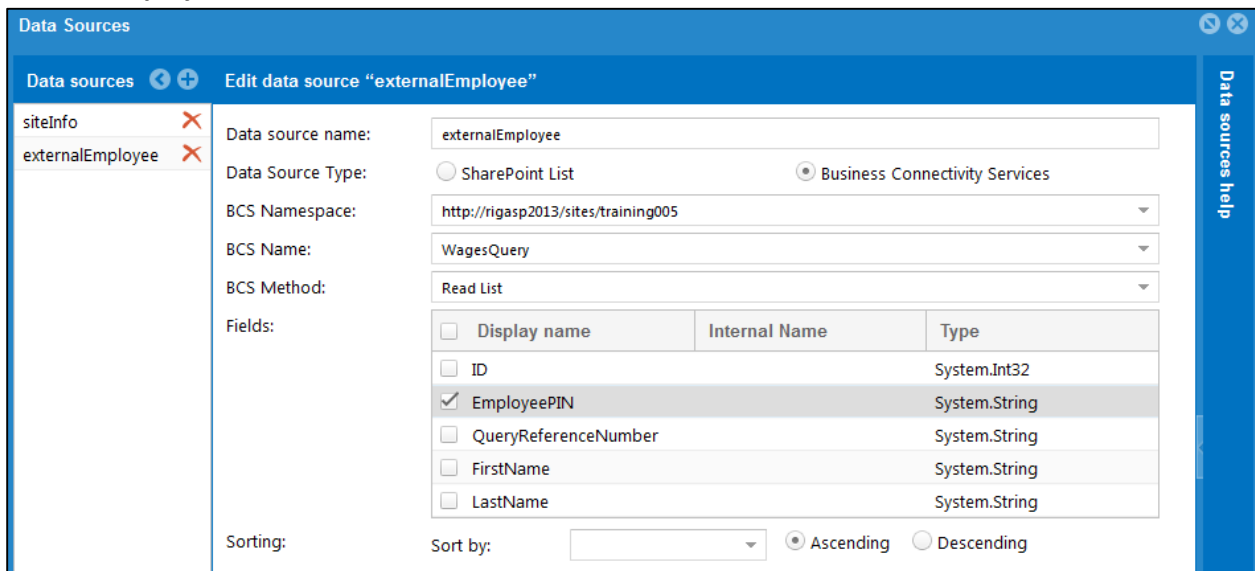


The screenshot shows the 'Data Sources' window with the 'Edit data source "siteInfo"' configuration. The 'Data Source Type' is set to 'Business Connectivity Services'. The 'BCS Namespace' is 'http://rigasp2013/sites/training005', 'BCS Name' is 'SiteInfo', and 'BCS Method' is 'Read List'. The 'Fields' table is as follows:

Display name	Internal Name	Type
<input type="checkbox"/>	ID	System.Int32
<input checked="" type="checkbox"/>	SITECode	System.String
<input type="checkbox"/>	SITENAME	System.String

Sorting is set to 'Ascending'.

- **externalEmployee**



The screenshot shows the 'Data Sources' window with the 'Edit data source "externalEmployee"' configuration. The 'Data Source Type' is set to 'Business Connectivity Services'. The 'BCS Namespace' is 'http://rigasp2013/sites/training005', 'BCS Name' is 'WagesQuery', and 'BCS Method' is 'Read List'. The 'Fields' table is as follows:

Display name	Internal Name	Type
<input type="checkbox"/>	ID	System.Int32
<input checked="" type="checkbox"/>	EmployeePIN	System.String
<input type="checkbox"/>	QueryReferenceNumber	System.String
<input type="checkbox"/>	FirstName	System.String
<input type="checkbox"/>	LastName	System.String

Sorting is set to 'Ascending'.

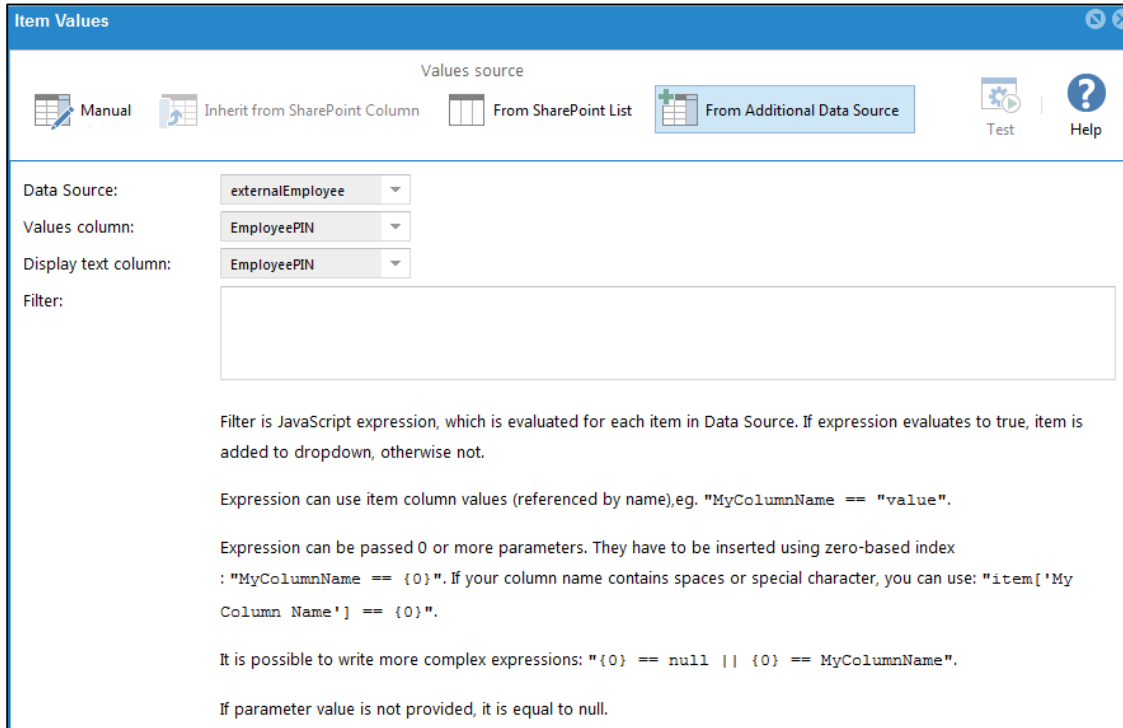
siteInfo is for **SITE Code** dropdown and **externalEmployee** is for **EmployeePIN** dropdown.

There are two data sources because there are two different dropdowns, which will be filled with information by using two different external content types. First one "siteInfo" will read data (Read List method) from SQL table by "SiteInfo" BCS and fill **SITECode** dropdown field.

Step 3. Configure dropdowns to store values:

Select dropdown → PDF Form Tools → Properties → Item Values (do the same with second dropdown)

- First dropdown



Item Values

Values source

Manual Inherit from SharePoint Column From SharePoint List From Additional Data Source Test Help

Data Source: externalEmployee

Values column: EmployeePIN

Display text column: EmployeePIN

Filter:

Filter is JavaScript expression, which is evaluated for each item in Data Source. If expression evaluates to true, item is added to dropdown, otherwise not.

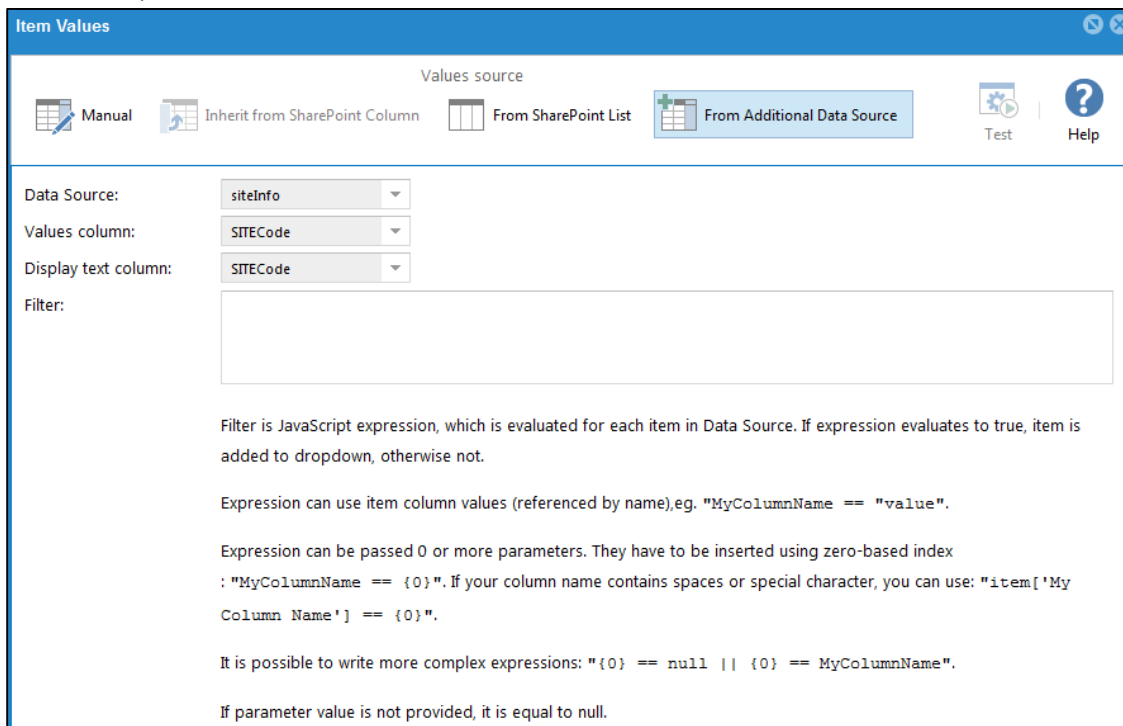
Expression can use item column values (referenced by name), eg. "MyColumnName == "value".

Expression can be passed 0 or more parameters. They have to be inserted using zero-based index : "MyColumnName == {0}". If your column name contains spaces or special character, you can use: "item['My Column Name'] == {0}".

It is possible to write more complex expressions: "{0} == null || {0} == MyColumnName".

If parameter value is not provided, it is equal to null.

- Second dropdown



Item Values

Values source

Manual Inherit from SharePoint Column From SharePoint List From Additional Data Source Test Help

Data Source: siteInfo

Values column: SITECode

Display text column: SITECode

Filter:

Filter is JavaScript expression, which is evaluated for each item in Data Source. If expression evaluates to true, item is added to dropdown, otherwise not.

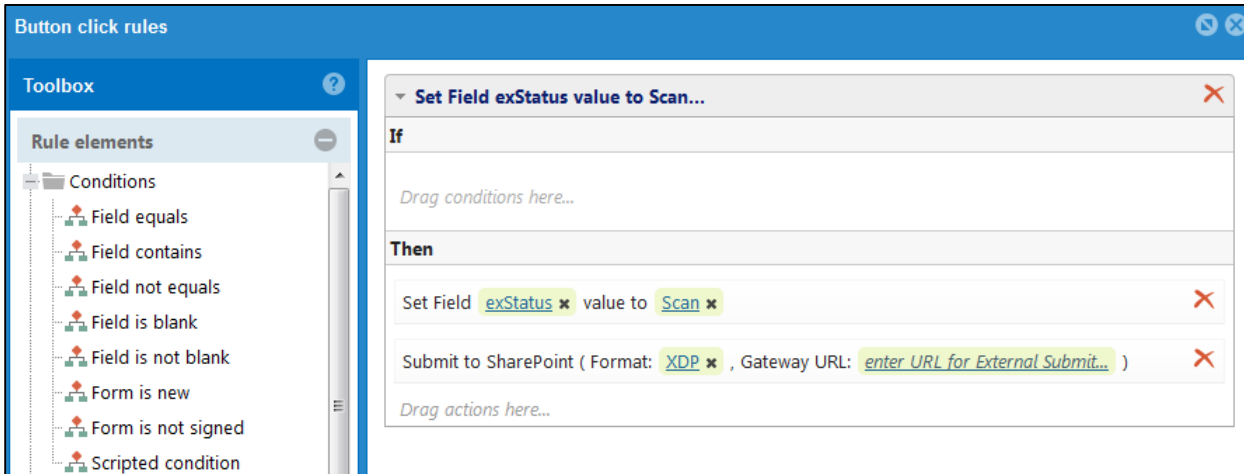
Expression can use item column values (referenced by name), eg. "MyColumnName == "value".

Expression can be passed 0 or more parameters. They have to be inserted using zero-based index : "MyColumnName == {0}". If your column name contains spaces or special character, you can use: "item['My Column Name'] == {0}".

It is possible to write more complex expressions: "{0} == null || {0} == MyColumnName".

If parameter value is not provided, it is equal to null.

Step 4. Add action to **Scan** button:



It will set **exStatus** field's value to **Scan** so form would not be submitted but re-loaded with proper values.

Step 5. Add **On Form Ready Script**:

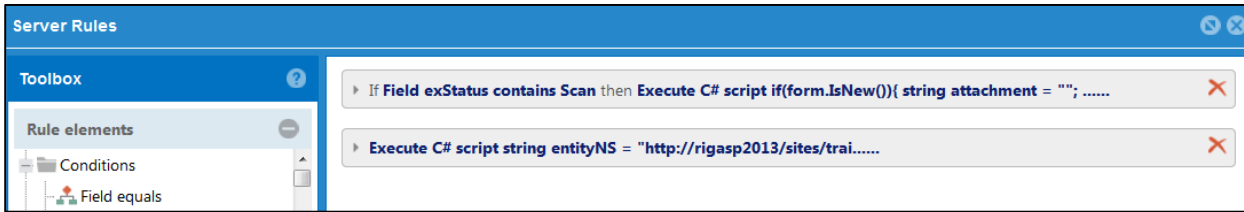
```

if (xfa.resolveNode("$data..Status").value != 'new'){
xfa.resolveNode('$form.Root.Default..FirstSection').access = "readOnly";
}
if(xfa.resolveNode("$data..Status").value == "second" || xfa.resolveNode("$data..Status").value ==
"third" || xfa.resolveNode("$data..Status").value == "final"){
xfa.resolveNode("$form..SecondSection").presence = 'visible';
}
if(xfa.resolveNode("$data..Status").value == "third" || xfa.resolveNode("$data..Status").value ==
"final"){
xfa.resolveNode("$form..ThirdSection").presence = 'visible';
}
if (xfa.resolveNode("$data..Status").value != 'second'){
  xfa.resolveNode('$form.Root.Default..SecondSection').access = "readOnly";
}
if (xfa.resolveNode("$data..Status").value != 'third'){
  xfa.resolveNode('$form.Root.Default..ThirdSection').access = "readOnly";
}

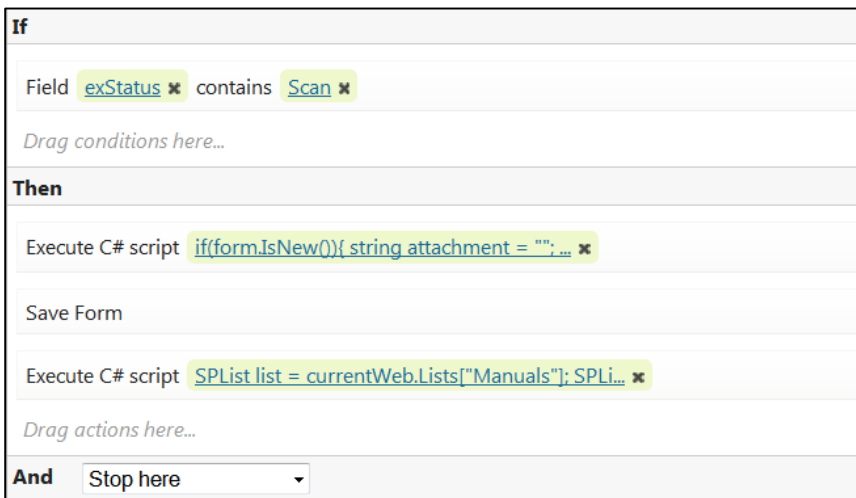
```

This script set presence of sections to “visible” and set sections to “readOnly” depending on **Status** field.

Step 6. Add conditions to **PDF Form Tools → Developer → Form Submit:**



First condition:



If field **exStatus** contains **Scan** then will be executed first script:

```

if(form.IsNew()){
    string attachment = "";
    if (!string.IsNullOrEmpty(data.resolveNode("EmployeePIN").Value)){
        attachment += "&ParamValue=" + data.resolveNode("EmployeePIN").Value;}
    if (!string.IsNullOrEmpty(data.resolveNode("SiteCode").Value)){
        attachment += "&SCode=" + data.resolveNode("SiteCode").Value;}
    if (!string.IsNullOrEmpty(data.resolveNode("EmailAddress").Value)){
        attachment += "&Email=" + data.resolveNode("EmailAddress").Value;}
    if (!string.IsNullOrEmpty(data.resolveNode("ContactNumber").Value)){
        attachment += "&contactNumber=" + data.resolveNode("ContactNumber").Value;}
    if (!string.IsNullOrEmpty(data.resolveNode("PayrollWeek").Value)){
        attachment += "&payrollW=" + data.resolveNode("PayrollWeek").Value;}
    if (!string.IsNullOrEmpty(data.resolveNode("AreaManager").Value)){
        attachment += "&AreaM=" + data.resolveNode("AreaManager").Value;}
    form.RedirectURL =
    "http://training.pdfsf.com/sites/training005/_layouts/15/PDFShareForms/PDFFormRender.aspx?List={D5
    B8C55C-F9BA-44CB-85BA-867D08C19A49}&RootFolder=%2Fsites%2Ftraining005%2FManuals
    &ContentTypeId=0x0101005E18512A6DD2784A85C521627FB7876200D4ABB58F0280A74AB57B17D765CE7BC7&Source=%
    2Fsites%2Ftraining005%2FManuals%2FForms%2FAllItems.aspx" + attachment; }

```

This script will redirect user to the new form which will be filled with information from Lists depending on dropdowns and all information which user added before clicking on **Scan** button.

NOTE: URL that is highlighted with red must be copied from new form's URL. It must be specified for proper case.

Then place “Save” action and **Execute script**, which will delete saved empty form:

```
SPList list = currentWeb.Lists["Manuals"];
SPListItem item = list.GetItemByUniqueId(form.DocumentID);
item.Delete();
```

Then stop actions by choosing **Stop here** in **And** tab.

Second condition:

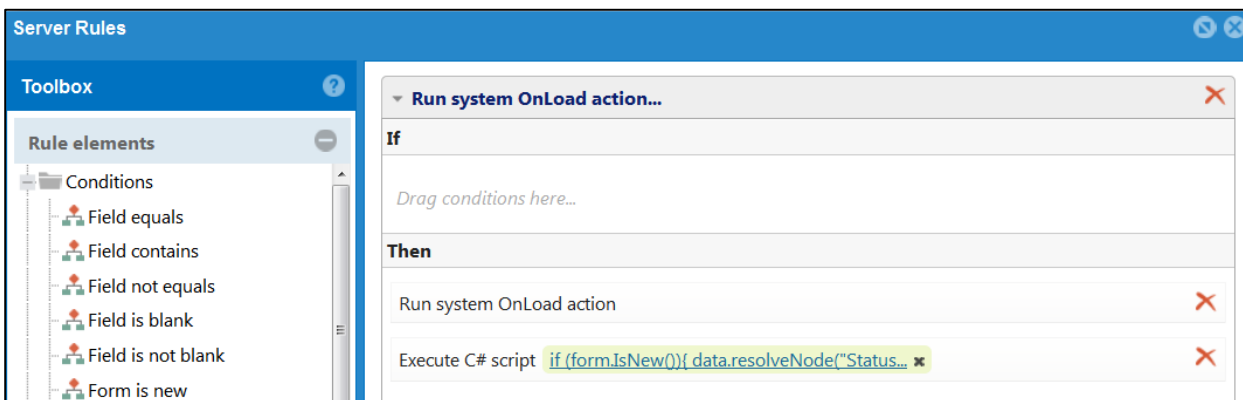


Script

This script connect to External Content Type **InformationStorage** and create new Item or upgrade existing with information from the form.

Step 7. Add conditions to PDF Form Tools → Developer → Form Load

Add **Execute C# script** under **Run system On Load** action:



Script



Step 8. Result:

First open (filled):

EMPLOYEE DETAILS	
Employee PIN	API12KO <input type="button" value="Scan"/>
Query Reference Number	<input type="text"/>
Title	
First Name	<input type="text"/>
Last Name	<input type="text"/>
Email Address	<input type="text" value="test@address.com"/>
Contact Number	<input type="text" value="22334455"/>
Other Contact Number	<input type="text"/>
SITE INFORMATION & QUERY	
SITE Code	BLINK21v1 <input type="button" value="Scan"/>
SITE Name	<input type="text"/>
Payroll Week Queried	1 <input type="button" value="Scan"/>
Area Manager	<input type="text" value="administrator"/>
Query Details	<input type="text"/>
Action Required By	<input type="text"/>
<input type="button" value="SUBMIT"/>	

After clicking on **Scan** button:

EMPLOYEE DETAILS	
Employee PIN	API12KO <input type="button" value="Scan"/>
Query Reference Number	<input type="text" value="12903"/>
Title	
First Name	<input type="text" value="Alex"/>
Last Name	<input type="text" value="Bond"/>
Email Address	<input type="text" value="test@address.com"/>
Contact Number	<input type="text" value="22334455"/>
Other Contact Number	<input type="text"/>
SITE INFORMATION & QUERY	
SITE Code	BLINK21v1 <input type="button" value="Scan"/>
SITE Name	<input type="text" value="Teleport"/>
Payroll Week Queried	1 <input type="button" value="Scan"/>
Area Manager	<input type="text" value="administrator"/>
Query Details	<input type="text"/>
Action Required By	<input type="text"/>
<input type="button" value="SUBMIT"/>	



First Submit:

ID	EmployeePIN	QueryReferenceNumber	FirstName	LastName	SITECode	SITENAME	PayrollWeekQueried	ActionTaken	Results	Approvalment
1017	API12KO	12903	Alex	Bond	BLINK21v1	Teleport	1			

+ Add new item

Second open:

Employee PIN	API12KO	Scan	Query Reference Number	12903
Title				
First Name	Alex	Last Name	Bond	
Email Address	test@address.com			
Contact Number	22334455	Other Contact Number		
SITE INFORMATION & QUERY				
SITE Code	BLINK21v1	Scan	SITE Name	Teleport
Payroll Week Queried	1	Area Manager	administrator	
Query Details				
Action Required By				
FOR PAYROLL DEPARTMENT ONLY				
Action Taken (Investigation)	Testing001			
SUBMIT				

Second Submit:

ID	EmployeePIN	QueryReferenceNumber	FirstName	LastName	SITECode	SITENAME	PayrollWeekQueried	ActionTaken	Results	Approvalment
1017	API12KO	12903	Alex	Bond	BLINK21v1	Teleport	1	Testing001		

+ Add new item



Third open:

Employee PIN	<input type="text" value="API12KO"/>	Scan	Query Reference Number	<input type="text" value="12903"/>
Title				
First Name	<input type="text" value="Alex"/>	Last Name	<input type="text" value="Bond"/>	
Email Address	<input type="text" value="test@address.com"/>			
Contact Number	<input type="text" value="22334455"/>	Other Contact Number	<input type="text"/>	
SITE INFORMATION & QUERY				
SITE Code	<input type="text" value="BLINK21v1"/>	Scan	SITE Name	<input type="text" value="Teleport"/>
Payroll Week Queried	<input type="text" value="1"/>	Area Manager	<input type="text" value="administrator"/>	
Query Details	<input type="text"/>			
Action Required By	<input type="text"/>			
FOR PAYROLL DEPARTMENT ONLY				
Action Taken (Investigation)	<input type="text" value="Testing001"/>			
RESULTS OF INVESTIGATION				
RESULTS	<input type="text" value="Third Section"/>			
<input checked="" type="radio"/> Approved <input type="radio"/> Rejected				
<input type="button" value="SUBMIT"/>				

Third Submit:

ID	EmployeePIN	QueryReferenceNumber	FirstName	LastName	SITECode	SITENAME	PayrollWeekQueried	ActionTaken	Results	Approval
1017	API12KO	12903	Alex	Bond	BLINK21v1	Teleport	1	Testing001	Third Section	Approved

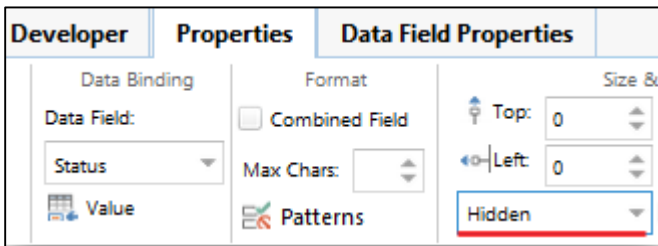
[Add new item](#)

Populating data to fields using BCS data source without scanning

Step 1. Prepare template

According to the **Employee PIN** dropdown's field value, will be filled **Query Reference Number, First Name** and **Last Name** fields. According to **SITE Code** dropdown's value will be filled **SITE Name**. **Employee PIN, Query Reference Number, First Name, Last Name, SITE Code, SITE Name** and **Payroll Week Queried** will be stored in **ExternalInfoStorage** List.

Status field need set to hidden: choose field and navigate to **PDF Form Tools → Properties**.

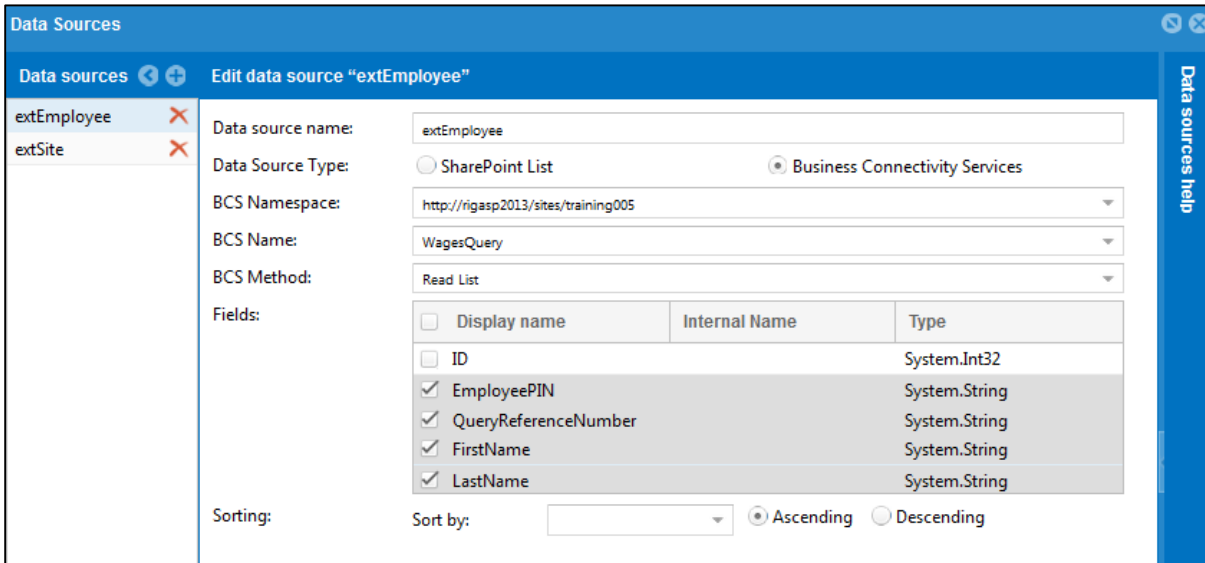


Prepared template:

EMPLOYEE DETAILS	
Employee PIN	Query Reference Number
Title	
First Name	Last Name
Email Address	
Contact Number	Other Contact Number
SITE INFORMATION & QUERY	
SITE Code	SITE Name
Payroll Week Queried	Area Manager
Query Details	
Action Required By	

Step 2. Create additional Data Sources with Business Connectivity Services type **PDF Form Tools → Developer → Data Sources**

First Data Source:



Data Sources | Edit data source "extEmployee"

Data source name: extEmployee

Data Source Type: SharePoint List Business Connectivity Services

BCS Namespace: http://rigasp2013/sites/training005

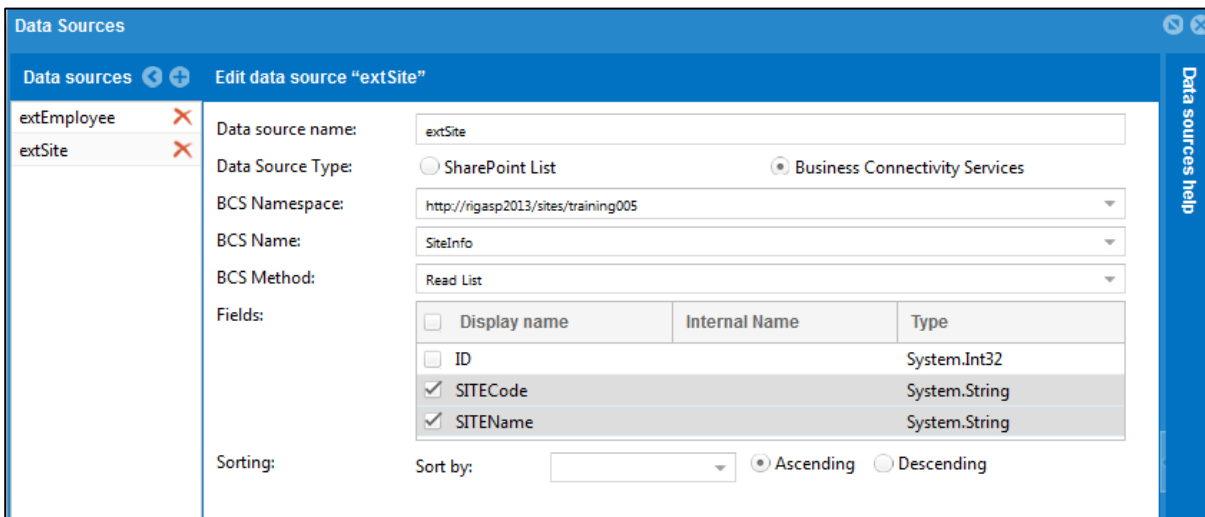
BCS Name: WagesQuery

BCS Method: Read List

Display name	Internal Name	Type
<input type="checkbox"/>	ID	System.Int32
<input checked="" type="checkbox"/>	EmployeePIN	System.String
<input checked="" type="checkbox"/>	QueryReferenceNumber	System.String
<input checked="" type="checkbox"/>	FirstName	System.String
<input checked="" type="checkbox"/>	LastName	System.String

Sorting: Sort by: [] Ascending Descending

Second Data Source:



Data Sources | Edit data source "extSite"

Data source name: extSite

Data Source Type: SharePoint List Business Connectivity Services

BCS Namespace: http://rigasp2013/sites/training005

BCS Name: SiteInfo

BCS Method: Read List

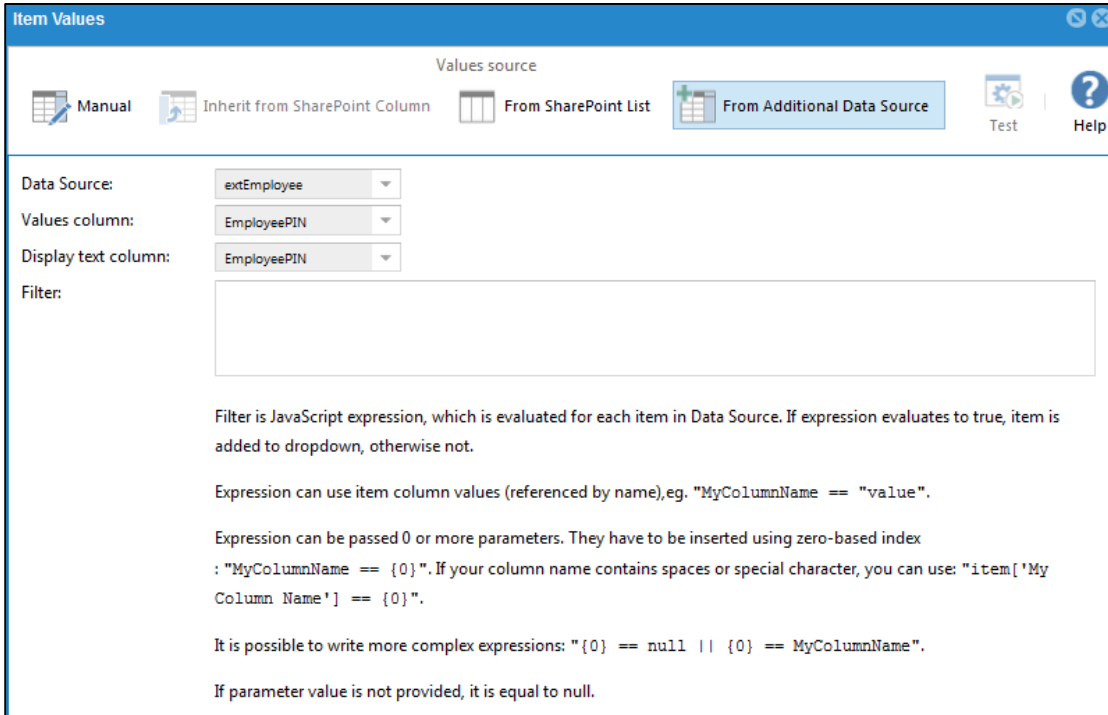
Display name	Internal Name	Type
<input type="checkbox"/>	ID	System.Int32
<input checked="" type="checkbox"/>	SITECode	System.String
<input checked="" type="checkbox"/>	SITENAME	System.String

Sorting: Sort by: [] Ascending Descending

Step 3. Configure dropdowns to store values.

Select dropdown → PDF Form Tools → Properties → Item Values (do the same with second dropdown)

- First dropdown:



Item Values

Values source

Manual Inherit from SharePoint Column From SharePoint List From Additional Data Source Test Help

Data Source: extEmployee

Values column: EmployeePIN

Display text column: EmployeePIN

Filter:

Filter is JavaScript expression, which is evaluated for each item in Data Source. If expression evaluates to true, item is added to dropdown, otherwise not.

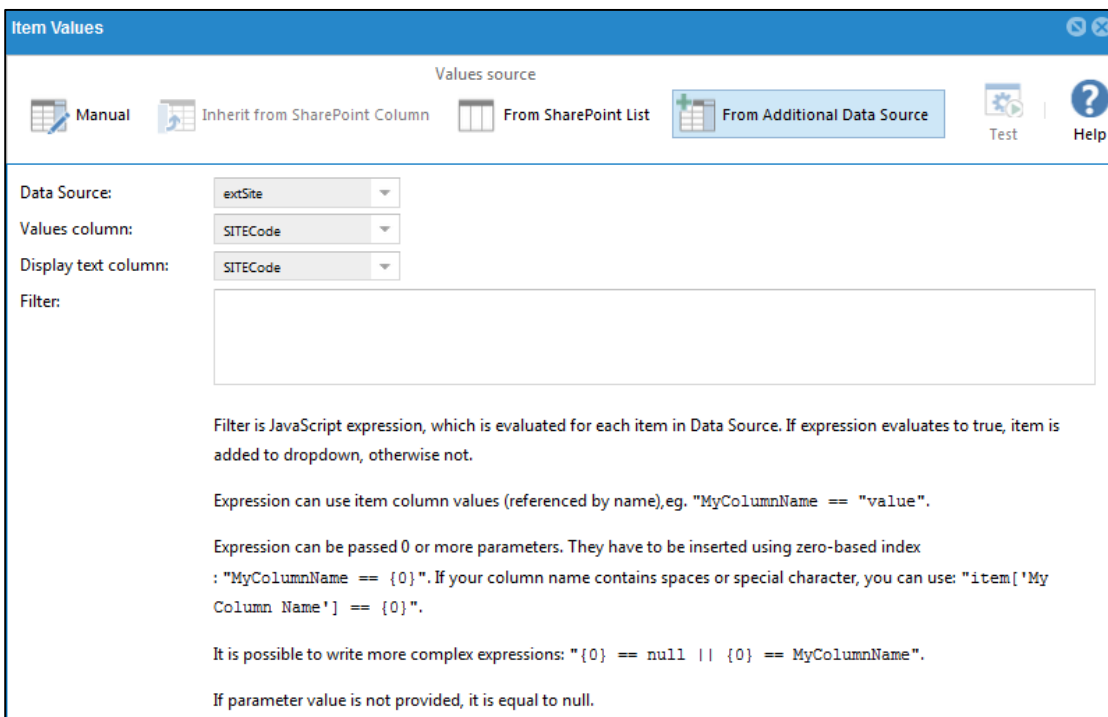
Expression can use item column values (referenced by name), eg. "MyColumnName == "value".

Expression can be passed 0 or more parameters. They have to be inserted using zero-based index : "MyColumnName == {0}". If your column name contains spaces or special character, you can use: "item['My Column Name'] == {0}".

It is possible to write more complex expressions: "{0} == null || {0} == MyColumnName".

If parameter value is not provided, it is equal to null.

- Second dropdown:



Item Values

Values source

Manual Inherit from SharePoint Column From SharePoint List From Additional Data Source Test Help

Data Source: extSite

Values column: SITECode

Display text column: SITECode

Filter:

Filter is JavaScript expression, which is evaluated for each item in Data Source. If expression evaluates to true, item is added to dropdown, otherwise not.

Expression can use item column values (referenced by name), eg. "MyColumnName == "value".

Expression can be passed 0 or more parameters. They have to be inserted using zero-based index : "MyColumnName == {0}". If your column name contains spaces or special character, you can use: "item['My Column Name'] == {0}".

It is possible to write more complex expressions: "{0} == null || {0} == MyColumnName".

If parameter value is not provided, it is equal to null.

This will populate **Employee PIN** and **SITE Code** dropdowns.

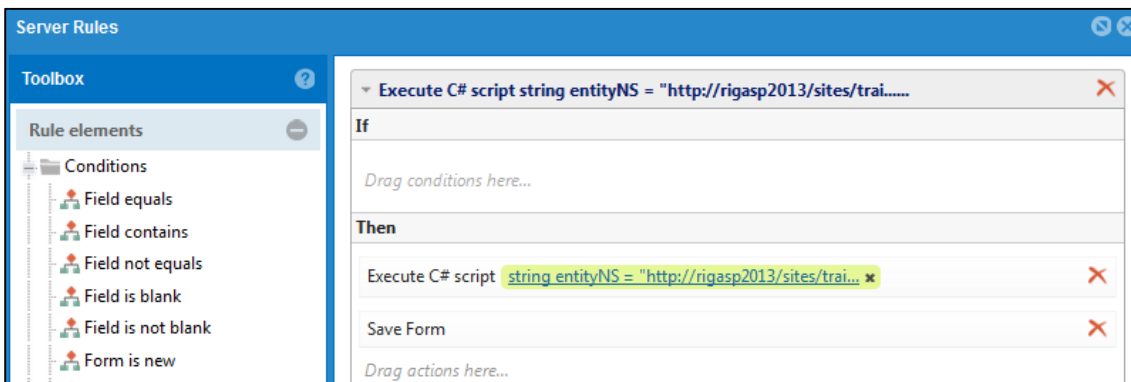
Step 4. Add On Form Ready Script:

```

if (xfa.resolveNode("$data..Status").value != 'new'){
xfa.resolveNode('$form.Root.Default..FirstSection').access = "readOnly";
}
if(xfa.resolveNode("$data..Status").value == "second" || xfa.resolveNode("$data..Status").value ==
"third" || xfa.resolveNode("$data..Status").value == "final"){
xfa.resolveNode("$form..SecondSection").presence = 'visible';
}
if(xfa.resolveNode("$data..Status").value == "third" || xfa.resolveNode("$data..Status").value ==
"final"){
xfa.resolveNode("$form..ThirdSection").presence = 'visible';
}
if (xfa.resolveNode("$data..Status").value != 'second'){
  xfa.resolveNode('$form.Root.Default..SecondSection').access = "readOnly";
}
if (xfa.resolveNode("$data..Status").value != 'third'){
  xfa.resolveNode('$form.Root.Default..ThirdSection').access = "readOnly";
}
  
```

This script set presence of sections to “visible” and set sections to “readOnly” depending on **Status** field.

Step 5. Add Execute C# script before Save Form action (PDF Form Tools → Developer → Form Submit):

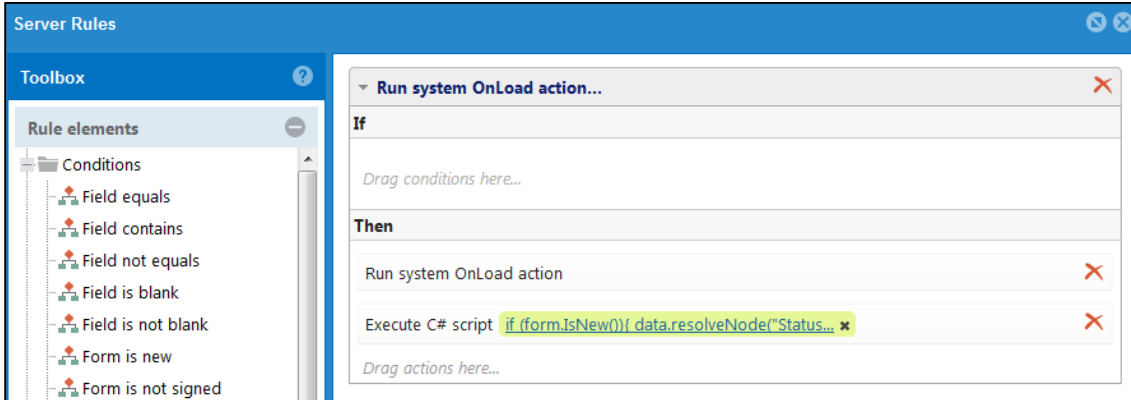


Script

This script connect to External Content Type **InformationStorage** and create new Item or upgrade existing with information from the form. Items will appear in the **ExternalInfoStorage List**.

Step 6. Add conditions to **PDF Form Tools → Developer → Form Load**

Add **Execute C# script** under **Run system On Load** action:



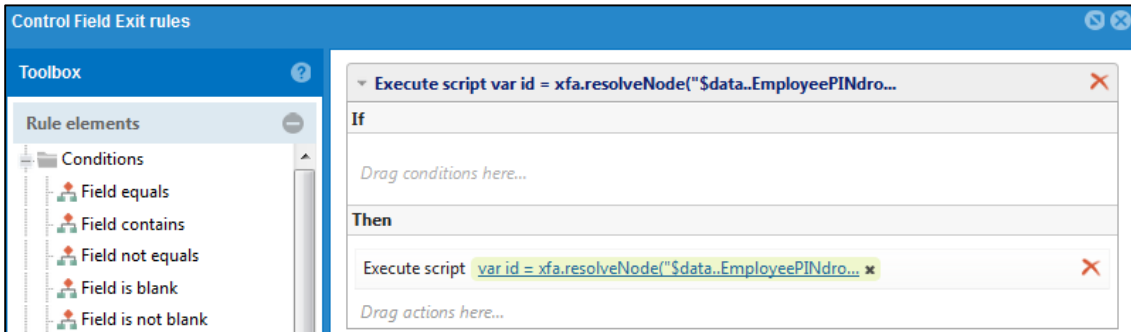
Script:

```

if (form.IsNew()){
    data.resolveNode("Status").Value = "new";
}
else{
    if(data.resolveNode("Status").Value == "third"){
        data.resolveNode("Status").Value = "final";
    }
    if(data.resolveNode("Status").Value == "second"){
        data.resolveNode("Status").Value = "third";
    }
    if (data.resolveNode("Status").Value == "new"){
        data.resolveNode("Status").Value = "second";
    }
}
}

```

Step 7. Add Field Exit action on Employee PIN dropdown:



Script:

```
var id = xfa.resolveNode("$data..EmployeePINdrop").value;
var i = 0;
for(i = 0;i< PdfForms.DataSources.extEmployee.length; i++ ){
    if(PdfForms.DataSources.extEmployee[i]["EmployeePIN"] == id){
        xfa.resolveNode("$data..QueryReferenceNumber").value =
PdfForms.DataSources.extEmployee[i]["QueryReferenceNumber"];
        xfa.resolveNode("$data..FirstName").value =
PdfForms.DataSources.extEmployee[i]["FirstName"];
        xfa.resolveNode("$data..LastName").value =
PdfForms.DataSources.extEmployee[i]["LastName"];
    }
}
```

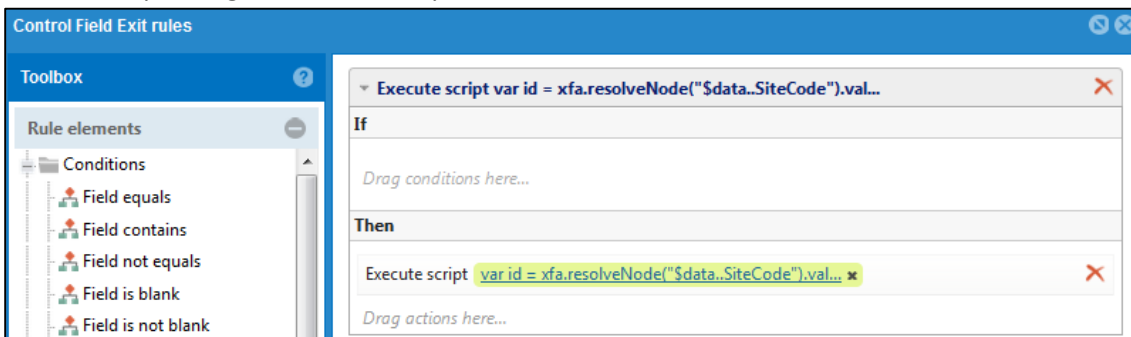
Fill fields depending on selected dropdown value.

Step 7.1. Add Field Exit action on SITE Code dropdown:

Script:

```
var id = xfa.resolveNode("$data..SiteCode").value;
var i = 0;
for(i = 0;i< PdfForms.DataSources.extSite.length; i++){
    if(PdfForms.DataSources.extSite[i]["SITECode"] == id){
        xfa.resolveNode("$data..SITENAME").value =
PdfForms.DataSources.extSite[i]["SITENAME"];
    }
}
```

Fill fields depending on selected dropdown value.





Step 8. Result:

First open and choosing values in dropdowns:

EMPLOYEE DETAILS			
Employee PIN	API12KO	Query Reference Number	12903
Title			
First Name	Alex	Last Name	Bond
Email Address			
Contact Number		Other Contact Number	
SITE INFORMATION & QUERY			
SITE Code	BLINK21v1	SITE Name	Teleport
Payroll Week Queried		Area Manager	
Query Details			
Action Required By			
<input type="button" value="SUBMIT"/>			

First Submit:

ID	EmployeePIN	QueryReferenceNumber	FirstName	LastName	SITECode	SITENAME	PayrollWeekQueried	ActionTaken	Results	Approvalment
1020	API12KO	12903	Alex	Bond	BLINK21v1	Teleport				
<input type="button" value="Add new item"/>										



Second open:

EMPLOYEE DETAILS	
Employee PIN	API12KO
Query Reference Number	12903
Title	
First Name	Alex
Last Name	Bond
Email Address	
Contact Number	
Other Contact Number	
SITE INFORMATION & QUERY	
SITE Code	BLINK21v1
SITE Name	Teleport
Payroll Week Queried	
Area Manager	
Query Details	
Action Required By	
FOR PAYROLL DEPARTMENT ONLY	
Action Taken (Investigation)	Test
SUBMIT	

Second Submit:

ID	EmployeePIN	QueryReferenceNumber	FirstName	LastName	SITECode	SITENAME	PayrollWeekQueried	ActionTaken	Results	Approvalment
1020	API12KO	12903	Alex	Bond	BLINK21v1	Teleport		Test		
+ Add new item										



Third open:

EMPLOYEE DETAILS			
Employee PIN	API12KO	Query Reference Number	12903
Title			
First Name	Alex	Last Name	Bond
Email Address			
Contact Number		Other Contact Number	
SITE INFORMATION & QUERY			
SITE Code	BLINK21v1	SITE Name	Teleport
Payroll Week Queried		Area Manager	
Query Details			
Action Required By			
FOR PAYROLL DEPARTMENT ONLY			
Action Taken (Investigation)	Test		
RESULTS OF INVESTIGATION			
RESULTS	Test005		
Approved <input checked="" type="radio"/>		Rejected <input type="radio"/>	
SUBMIT			

Third Submit:

ID	EmployeePIN	QueryReferenceNumber	FirstName	LastName	SITECode	SITENAME	PayrollWeekQueried	ActionTaken	Results	Approval
1020	API12KO	12903	Alex	Bond	BLINK21v1	Teleport		Test	Test005	Approved
+ Add new item										

Form Submit script for case with Scan

```

string entityNS = "http://rigasp2013/sites/training005"; //there must be user's site
string entityName = "InformationStorage"; //user's ECT
using(BCSAccessor bdcAccessor = new BCSAccessor(currentWeb.Site, entityNS, entityName)) {
    if(form.IsNew()){var fieldValues = new System.Collections.Hashtable();
        fieldValues["EmployeePIN"] = form.GetQueryParam("ParamValue");
        fieldValues["QueryReferenceNumber"] = data.resolveNode("QueryReferenceNumber").Value;
        fieldValues["FirstName"] = data.resolveNode("FirstName").Value;
        fieldValues["LastName"] = data.resolveNode("LastName").Value;
        fieldValues["SITECode"] = data.resolveNode("SiteCode").Value;
        fieldValues["SITENAME"] = data.resolveNode("SITENAME").Value;
        fieldValues["PayrollWeekQueried"] = data.resolveNode("PayrollWeek").Value;
        fieldValues["ActionTaken"] = data.resolveNode("ActionTaken").Value;
        fieldValues["Results"] = data.resolveNode("Results").Value;
        fieldValues["Approvement"] = data.resolveNode("Approvement").Value;
        Identity id = bdcAccessor.CreateItem("Create", fieldValues);
        data.resolveNode("ItemID").Value = id.Serialize();}
    else{Identity id = Identity.Deserialize(data.resolveNode("ItemID").Value);
        IEntityInstance entityInstance = bdcAccessor.GetEntityInstance("Read Item", id);
        var fieldValues2 = new System.Collections.Hashtable();
        fieldValues2["EmployeePIN"] = data.resolveNode("EmployeePIN");
        fieldValues2["QueryReferenceNumber"] =
data.resolveNode("QueryReferenceNumber").Value;
        fieldValues2["FirstName"] = data.resolveNode("FirstName").Value;
        fieldValues2["LastName"] = data.resolveNode("LastName").Value;
        fieldValues2["SITECode"] = data.resolveNode("SiteCode").Value;
        fieldValues2["SITENAME"] = data.resolveNode("SITENAME").Value;
        fieldValues2["PayrollWeekQueried"] = data.resolveNode("PayrollWeek").Value;
        fieldValues2["ActionTaken"] = data.resolveNode("ActionTaken").Value;
        fieldValues2["Results"] = data.resolveNode("Results").Value;
        fieldValues2["Approvement"] = data.resolveNode("Approvement").Value;
        bdcAccessor.UpdateItem("Update", entityInstance, fieldValues2);    } }
  
```

[Back to the guide](#)

On Form Load script for case with Scan

```

/*This part check how many times form was opened. If it is opened second time Status field will be
filled with text "Second" and so on till form is opened fourth time. In this case Status field
will be filled with text "final". This is made for control which sections user can see. With every
step one additional section will appear (but this is done in On Form Ready) */
if (form.IsNew()){data.resolveNode("Status").Value = "new";}
else{
    if(data.resolveNode("Status").Value == "third"){
        data.resolveNode("Status").Value = "final";}
    if(data.resolveNode("Status").Value == "second"){
        data.resolveNode("Status").Value = "third";}
    if (data.resolveNode("Status").Value == "new"){
        data.resolveNode("Status").Value = "second";}}
//All "if" conditions check if URL contain proper parameter
if (form.GetQueryParam("ParamValue")!= null)
    {string entityNS = "http://rigasp2013/sites/training005"; //user's site (can be found in SP
Designer External Content Type tab)
    string entityName = "WagesQuery"; //ECT name from which information is taken
    using(BCSAccessor bdcAccessor = new BCSAccessor(currentWeb.Site, entityNS,
entityName))
    {
        object[] array1 = new object[1];
        array1[0] = form.GetQueryParam("ParamValue");
        DataTable dt2 = bdcAccessor.GetItem("Read Item", array1);
        foreach (DataRow row in dt2.Rows)
            //get information and write it down to the proper fields
            {
                data.resolveNode("EmployeePIN").Value =
form.GetQueryParam("ParamValue");
                data.resolveNode("QueryReferenceNumber").Value =
row["QueryReferenceNumber"].ToString();
                data.resolveNode("FirstName").Value = row["FirstName"].ToString();
                data.resolveNode("LastName").Value = row["LastName"].ToString();
            }
        }
    }
}
if (form.GetQueryParam("SCode")!= null){
    string entityNS = "http://rigasp2013/sites/training005";
    string entityName = "SiteInfo";
    using(BCSAccessor bdcAccessor = new BCSAccessor(currentWeb.Site, entityNS,
entityName))
    {
        object[] arr1 = new object[1];
        arr1[0] = form.GetQueryParam("SCode");
        DataTable dt = bdcAccessor.GetItem("Read Item", arr1);
        foreach (DataRow row in dt.Rows)
            {
                data.resolveNode("SiteCode").Value = form.GetQueryParam("SCode");
                data.resolveNode("SITENAME").Value = row["SITENAME"].ToString();
            }
        }
    }
}

```



```
if (form.GetQueryParam("Email")!= null){  
    data.resolveNode("EmailAddress").Value = form.GetQueryParam("Email");}  
if (form.GetQueryParam("contactNumber")!= null){  
    data.resolveNode("ContactNumber").Value = form.GetQueryParam("contactNumber");}  
if (form.GetQueryParam("payrollW")!= null){  
    data.resolveNode("PayrollWeek").Value = form.GetQueryParam("payrollW");}  
if (form.GetQueryParam("AreaM")!= null){  
    data.resolveNode("AreaManager").Value = form.GetQueryParam("AreaM");}
```

[Back to the guide](#)

On Form Submit script for case without Scan

```
String entityNS = "http://rigasp2013/sites/training005"; //specify proper way
string entityName = "InformationStorage";
using(BCSAccessor bdcAccessor = new BCSAccessor(currentWeb.Site, entityNS, entityName)) {
    if(form.IsNew()){
        var fieldValues = new System.Collections.Hashtable();
        fieldValues["EmployeePIN"] = data.resolveNode("EmployeePINdrop").Value;
        fieldValues["QueryReferenceNumber"] = data.resolveNode("QueryReferenceNumber").Value;
        fieldValues["FirstName"] = data.resolveNode("FirstName").Value;
        fieldValues["LastName"] = data.resolveNode("LastName").Value;
        fieldValues["SITECode"] = data.resolveNode("SiteCode").Value;
        fieldValues["SITENAME"] = data.resolveNode("SITENAME").Value;
        fieldValues["PayrollWeekQueried"] = data.resolveNode("PayrollWeek").Value;
        fieldValues["ActionTaken"] = data.resolveNode("ActionTaken").Value;
        fieldValues["Results"] = data.resolveNode("Results").Value;
        fieldValues["Approvement"] = data.resolveNode("Approvement").Value;
        Identity id = bdcAccessor.CreateItem("Create", fieldValues);
        data.resolveNode("ItemID").Value = id.Serialize();
    }
    else{
        Identity id = Identity.Deserialize(data.resolveNode("ItemID").Value);
        IEntityInstance entityInstance = bdcAccessor.GetEntityInstance("Read Item", id);
        var fieldValues2 = new System.Collections.Hashtable();
        fieldValues2["EmployeePIN"] = data.resolveNode("EmployeePINdrop");
        fieldValues2["QueryReferenceNumber"] =
data.resolveNode("QueryReferenceNumber").Value;
        fieldValues2["FirstName"] = data.resolveNode("FirstName").Value;
        fieldValues2["LastName"] = data.resolveNode("LastName").Value;
        fieldValues2["SITECode"] = data.resolveNode("SiteCode").Value;
        fieldValues2["SITENAME"] = data.resolveNode("SITENAME").Value;
        fieldValues2["PayrollWeekQueried"] = data.resolveNode("PayrollWeek").Value;
        fieldValues2["ActionTaken"] = data.resolveNode("ActionTaken").Value;
        fieldValues2["Results"] = data.resolveNode("Results").Value;
        fieldValues2["Approvement"] = data.resolveNode("Approvement").Value;
        bdcAccessor.UpdateItem("Update", entityInstance, fieldValues2);
    }
}
```

[Back to the guide](#)