

Create Conditional Subviews

Last Modified on 04/11/2023 9:53 am EDT

You can create a workflow that calls and manages subviews from other workflows.

1. Create a new workflow in which you need to store the required subviews.
2. In the workflow, create a Form that will have all the required subviews. You need not connect this form with a Start and End activity.
3. Create a SQL table to store the view paths and the information required for calling the views.

For example:

```
insert into DemoTable_51718 values ('India','../../51718_Subflows/{active}/SubFlow_Details/DefaultView.aspx','test description1')
insert into DemoTable_51718 values ('Israel','../../51718_Subflows/{active}/SubFlow_Details/Israel.aspx','test description2')
insert into DemoTable_51718 values ('London','../../51718_Subflows/{active}/SubFlow_Details/London.aspx','test description3')
```

4. Create a stored procedure to search the SQL table.

For example:

```
CREATE proc [dbo].[USP_GetDataforDemo] as
select
fldId as ID,
Name
from DemoTable_51718
GO
```

5. Create another workflow.
6. In this workflow, do the following:
 - a. Create two Data Source objects, one that will be connected to the stored procedure, and other for receiving the path and transferring it to the subview variable.
 - b. Create an object that will be the trigger.
 - c. Connect the object to the first Data Source.
7. Create another stored procedure to receive the selection of combo box and transfer the path of .aspx file from SQL table to the subview in the workflow.

For example:

```
(
@Id int
)
as
select
fldId as ID,
Name,
Path
from DemoTable_51718 where fldId=@Id
```

8. Create a SubView activity and assign the variable path according to the previous stored procedure.

For example:

```
VirtualPath='<%# {ds.item.Path} %>'
```