

# VBCS Lab

## Master-Detail Pattern

### Introduction

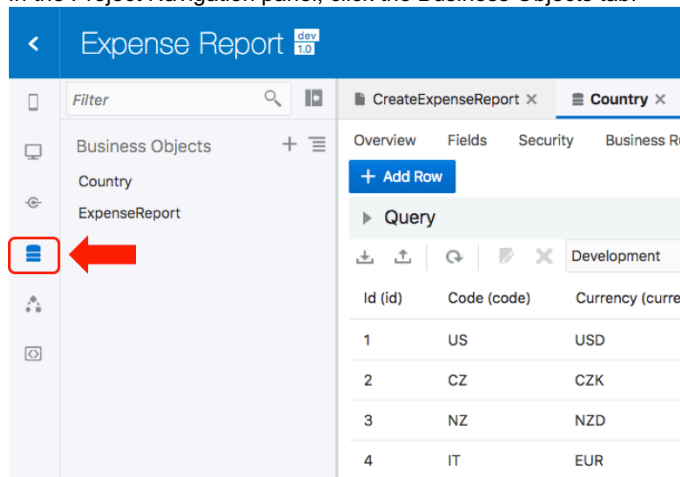
In this lab we will examine the Master-Detail pattern. We will add a list of child Line Items to the Edit Expense Report page.

### Hands on Lab Instructions

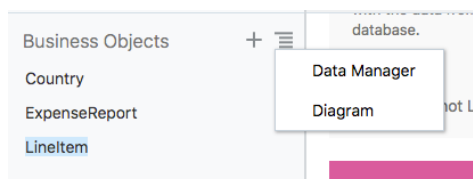
#### Checking Out the Project

First we will check out the project from Git. As Git exports don't contain data we will also have to upload the data for the all the business objects from one ZIP file.

1. Click Import > Import from Git
2. Click Add Credentials
3. Enter <https://developer.em2.oraclecloud.com/developer20509-cloud01/> for the DevCS URL and your username and password. Click Finish.
4. Enter the following information:
  - DevCS URL with Credentials - <https://developer.em2.oraclecloud.com/developer20509-cloud01/>
  - Project Selection - VBCS Training
  - Repository Selection - expenseReport.git
  - Branch Selection - solution
  - Application Name - MasterDetailYourName
  - Application ID - accept default
5. Click Import. The application is opened in the list of apps.
6. Click your app in the list of apps to open it.
7. In the Project Navigation panel, click the Business Objects tab.



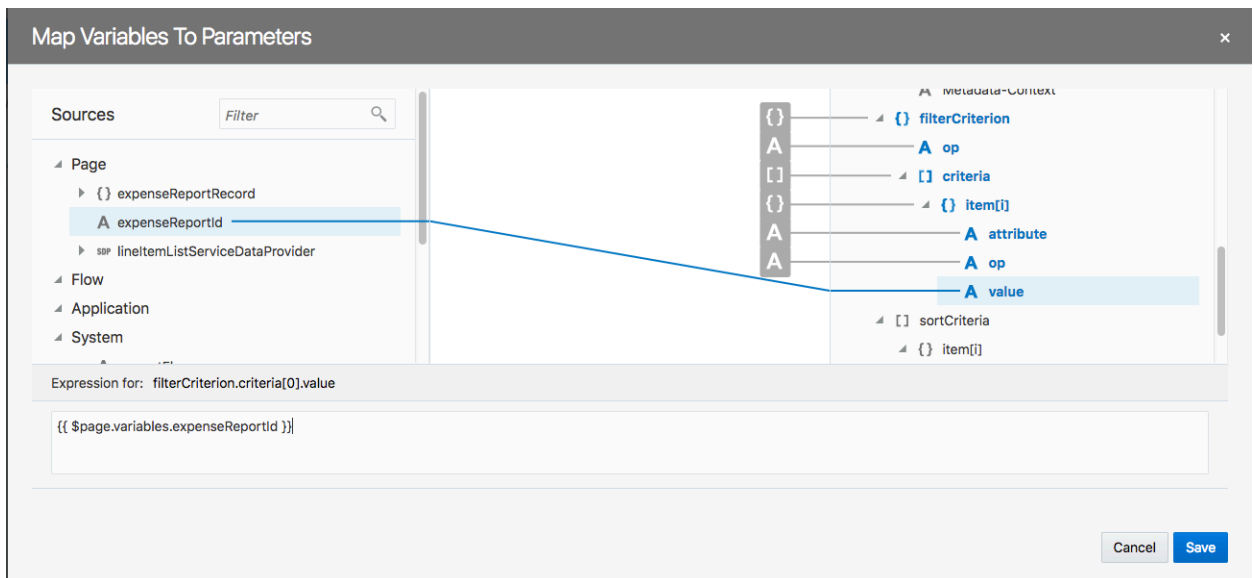
8. Click the Business Objects menu and click Data Manager.



9. Click Import from File.
10. Upload ExpenseReportData.zip

### Adding a Child Table

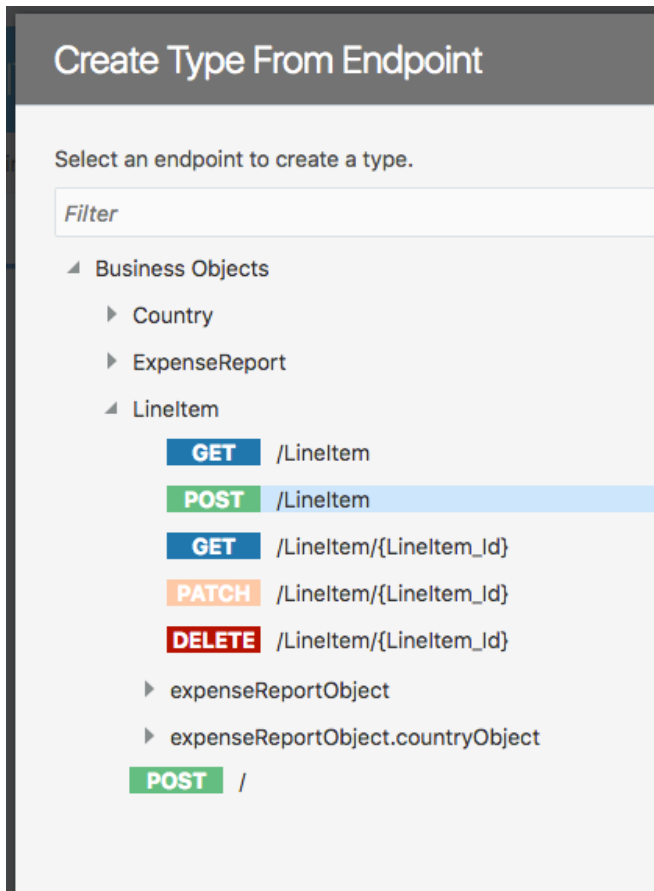
1. Open EditExpenseReport. Use the trick where you use the Live mode from the list of Expense Reports so it loads with some data in it.
2. Add a Heading between the form and the buttons and call it Line Items
3. Add a List View below the heading.
4. In the Property Inspector, click Add Data
5. In the Choose an Endpoint tab, select Business Objects > Lineltem > GET /Lineltem. Click Next.
6. Select Custom Template. Click Next.
7. Choose the **name** and **amount** fields. Click Next. Then click Finish.
8. Open Variables tab. Select lineltemListServiceDataProvider variable and in the Property Inspector click the (->) button for Input Parameters.
9. and set the following in filterCriterion:
  - op: "\$or"
  - attribute: "expenseReport"
  - op: "\$eq"
  - value: \$page.variables.expenseReportId (you can drag expenseReportId from the left panel to bind it)



10. Click Save

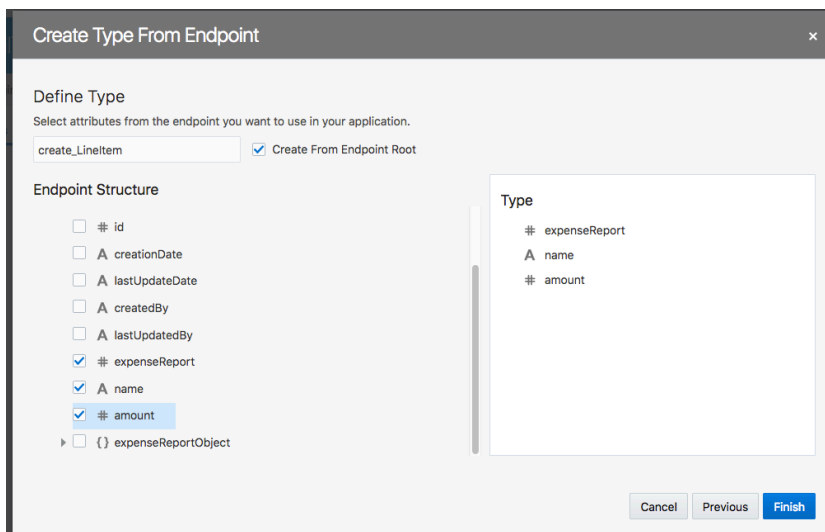
### Adding a New Line Item Form

11. Go to the Variables panel of the EditExpenseReport page.
12. Click the Types tab. Create a New Type from Endpoint.
13. Select Business Objects > Lineltem > POST /Lineltem



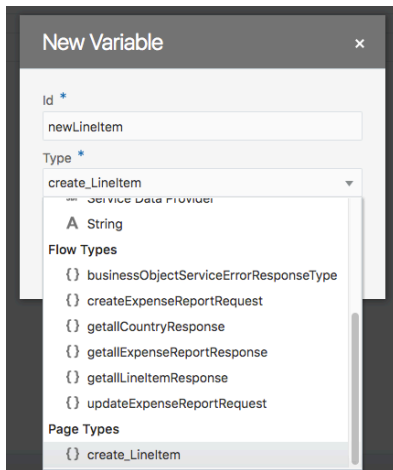
14. Click Next.

15. Add the **expenseReport**, **name**, and **amount** fields.



16. Click Finish.

17. Click the Variable Tab and add a new variable called newLinItem of type createLinItem



18. Expand the newLineItem variable and select the expenseReport property.
19. In the Property Inspector, change Default Value to {{ \$page.variables.expenseReportId }}
20. Go back to the Page Designer.
21. Drop a Form Layout component between the Line Items heading and the list of Line Items

My Application

## Edit Expense Report

Name

### Line Items

Form Layout

name	Hotel
Amount	400

22. Add an Input Text component and a Currency component to the Form Layout. Change the display labels to Name on the Input Text and Amount on the Currency component.

### Line Items

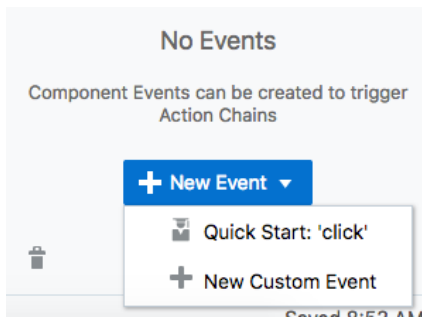
Name

Amount

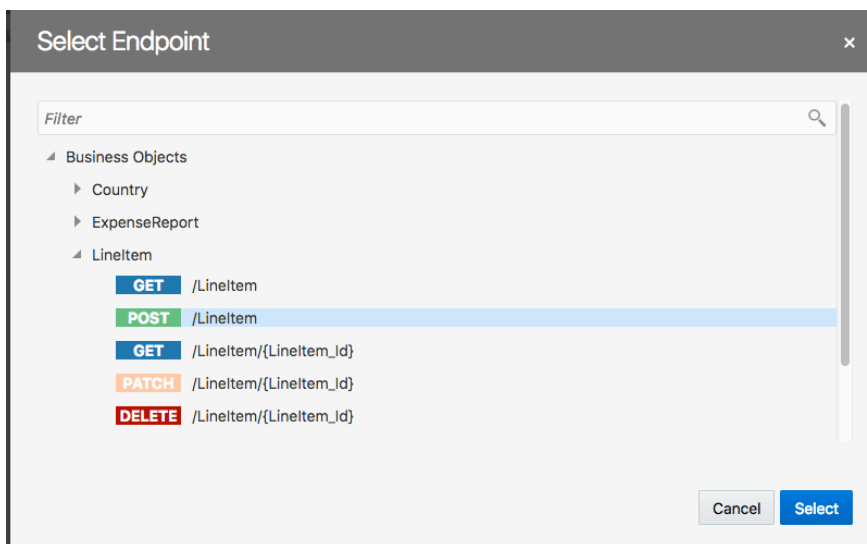
name	Hotel
Amount	400

23. Select the Name field in the Page Designer. In the Data tab of the Property Inspector, bind Value to {{ \$page.variables.newLineItem.name }}.
24. Do the same to bind the Amount field to {{ \$page.variables.newLineItem.amount }}
25. Add a Button below the Form Component. Change the Text to Add.

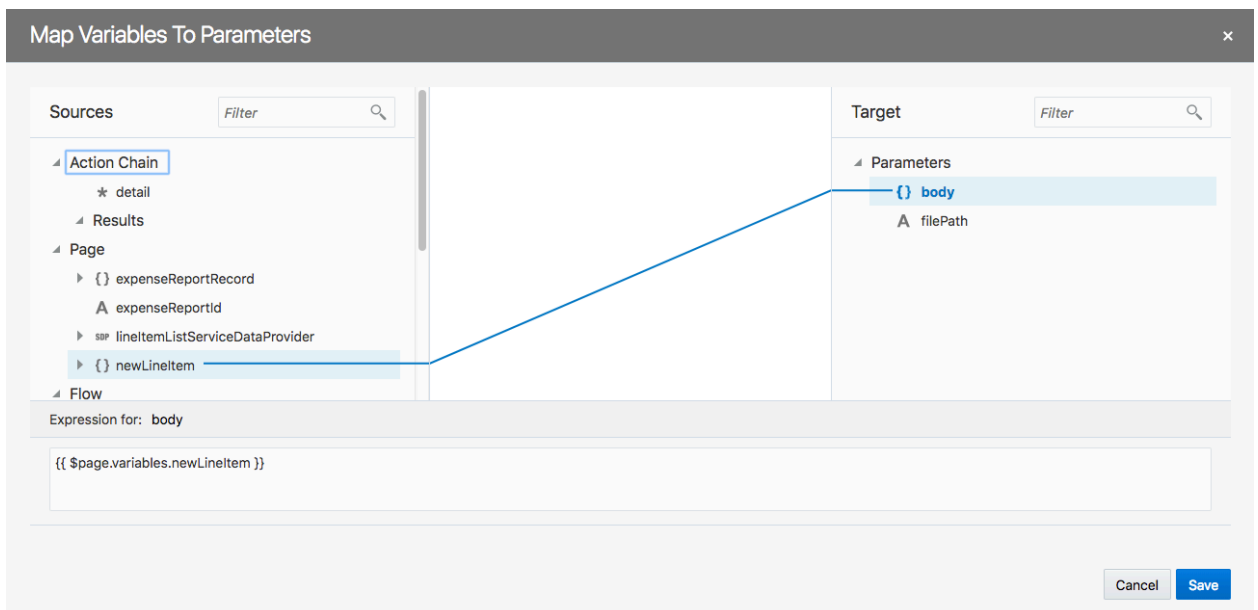
26. Select the Button and click the Events tab of the Property Inspector. Click New Event > Quick Start: 'click'



27. Drop a Call REST Endpoint action under Start. Click Select Endpoint and select Business Objects > Lineltem > POST /Lineltem.



28. Click (->) for Input Parameters. Map {} newLineltem on the left panel to {} body on the right panel. Click Save.



29. Drag a Fire Data Provider Event under the Call REST Endpoint action. Select `{{ $page.variables.lineItemListServiceDataProvider }}` as the Event Target. Set Refresh as the Type.

### Fire Data Provider Event

Id \*

fireDataProviderEvent1

Event Target \* (x)

{{ \$page.variables.lineItemListServiceDataProvider }}

Type

☒ Refresh

☐ Mutate

30. Drag a Reset Variables action to under the Fire Data Provider Event action. Set `$page.variables.newLineItem` as the Variables to Reset.

### Reset Variables

Id \*

resetVariables1

Variables to Reset (x)

\$page.variables.newLineItem

31. Run and test the app.