

Guide: Event Handler for Dropdown Selection in D365 FinOps Forms Using X++

This guide provides step-by-step instructions on setting up event handlers for dropdown selection in D365 FinOps forms using X++. The objective is to enable the sub-department dropdown to display values based on the selection made in the department dropdown.

Step 1: Create Event Handler for Department

First, create an event handler for the Department field by copying the `OnModified` method from the specific form's field and pasting it into a class. Then, paste the following code under the event handler method:

Code:

```
[FormControlEventHandler(formControlStr(DIN_ItemPurchaseForm,
DIN_ItemInventoryTable_NTI_ItemDepartment), FormControlEventType::Lookup)]

public static void DIN_ItemInventoryTable_NTI_ItemDepartment_OnLookup(FormControl sender,
FormControlEventArgs e)
{
    Query          query;
    QueryBuildDataSource qbds;
    SysTableLookup  sysTableLookup;

    // Initialize the SysTableLookup object
    sysTableLookup = SysTableLookup::newParameters(tableNum(Departments), sender);

    // Create a new query
    query = new Query();
    qbds = query.addDataSource(tableNum(Departments));

    // (Optional) Add ranges to filter the data if needed
    // qbds.addRange(fieldNum(Department, FieldName)).value(queryValue("SomeValue"));
```

```

// Set the fields to be displayed in the lookup
sysTableLookup.addLookupfield(fieldNum(Departments, Dep_Code));
sysTableLookup.addLookupfield(fieldNum(Departments, Dep_Name));

// Perform the lookup
sysTableLookup.parmQuery(query);
sysTableLookup.performFormLookup();
}

```

Step 2: Modify Method for Department Field

Next, create a `modified` method under the department form's field:

```

[Control("String")]
class DIN_ItemInventoryTable_NTI_ItemDepartment
{
    /// <summary>
    /// Method triggered when the Department field is modified
    /// </summary>
    /// <returns></returns>
    public boolean modified()
    {
        boolean ret;

        ret = super();

        //super();
    }
}

```

```

// Refresh the sub-department field lookup

// Ensure the sub-department field is re-initialized

DIN_ItemInventoryTable.NTI_ItemSubDepartment = "";

DIN_ItemInventoryTable_ds.refreshEx(FieldNum(DIN_ItemInventoryTable,
NTI_ItemSubDepartment));


return ret;

}

}

```

Step 3: Create Event Handler for Sub-Department

Create an event handler for the Sub-Department field by following the same method as for the department. Then, create a class and paste the following code under that event handler:

Code:

```

[FormControlEventHandler(formControlStr(DIN_ItemPurchaseForm,
DIN_ItemInventoryTable_NTI_ItemSubDepartment), FormControlEventType::Lookup)]

```

```

public static void DIN_ItemInventoryTable_NTI_ItemSubDepartment_OnLookup(FormControl
sender, FormControlEventArgs e)

```

```

{

    Query          query;

    QueryBuildDataSource  qbds;

    SysTableLookup      sysTableLookup;

    FormRun              formRun;

    FormDataSource        formDataSource;

    DIN_ItemInventoryTable  dinItemInventoryTable;

```

```
formRun = sender.formRun();

formDataSource = formRun.dataSource(tableStr(DIN_ItemInventoryTable));

dinItemInventoryTable = formDataSource.cursor();


// Initialize the SysTableLookup object

sysTableLookup = SysTableLookup::newParameters(tableNum(SubDepartment), sender);


// Create a new query

query = new Query();

qbds = query.addDataSource(tableNum(SubDepartment));


// Add range based on selected department

qbds.addRange(fieldNum(SubDepartment,
Dept)).value(queryValue(dinItemInventoryTable.NTI_ItemDepartment));


// Set the fields to be displayed in the lookup

// sysTableLookup.addLookupfield(fieldNum(SubDepartment, Dept));

sysTableLookup.addLookupfield(fieldNum(SubDepartment, SubDept));


// Perform the lookup

sysTableLookup.parmQuery(query);

sysTableLookup.performFormLookup();

}
```

Follow these steps to ensure that the sub-department dropdown displays values associated with the selected department in the D365 FinOps forms using X++.