Sohaib Workflow Tutorial:

**Caution: See if you have credentials for the following module or not. Only work on module for which you have credential**

# Making Table

Let start work on fake requirement. Suppose we are working in HR module and client wants that worker will add their expense and send for Approval. For this requirement, first we create a custom table and create its relation with Worker table. And then develop a custom workflow on it.  
So this post contains tasks

* Table creation and relation with worker
* Create a simple entry form and list page.
* Workflow  
    
  So first we create a new table and generate its relation with Worker ID. If we open HCMworker table we will find that its primary key is “PersonnelNumber”.

Making Enum to use in tables:Make base enum. Make Extended data type (Enum type).  We add new enum type which has expected values, medical, Certification, Food expense and others. Now create a new table set its Name as **Sohaib\_Worker\_table**.(Make Table name with your name at start)

## Make Fields of table:

* + 1. Person ID (forging key)
    2. Expense type (education, medical etc.) (enum type) (specify enum type)
    3. Status (pending, accepted, declined) (enum type) (specify enum type)
    4. Amount

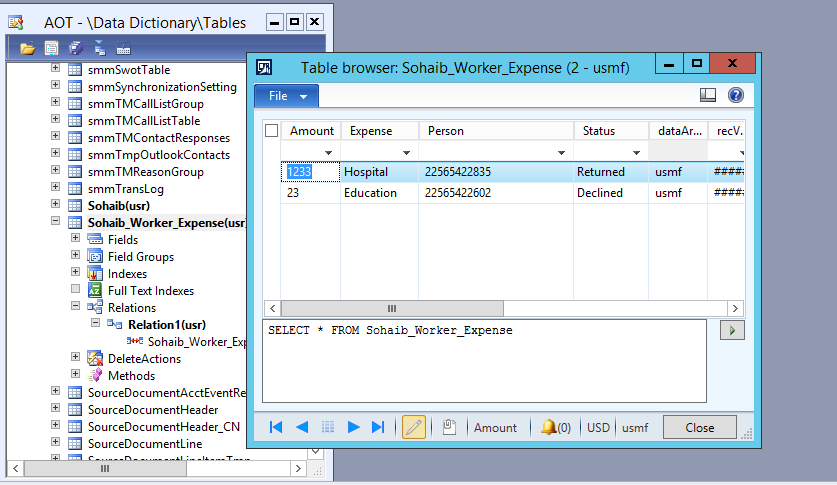
 Make a key in your table (**Sohaib\_Worker\_table**) of int64 (same as personal key type in HCMWorker)

Right Click on relation. Set relation table (HCM Worker). Now make a new relation of type (Normal)

Specify key (person) (Your table)

Related key (person) (HCMWorker)

## Table final structure will be as follow.



## Now click on method of the data source and over-ride this method

**public** **boolean** canSubmitToWorkflow(**str** \_workflowType =' ')

{

**boolean** ret;

**if** (this.Status ==Sohaib\_Woker\_Status\_Enum::Pending)

{

ret = **boolean**::true;

}

**else**

{

ret =**boolean**::false;

}

**return** **ret**;

}

## New add a new static method with Name “updateworkflow” and update it as follow

**public** **static** **void** updateWorkFlowState(RefRecId \_id, Sohaib\_Woker\_Status\_Enum \_status)

{

Sohaib\_Worker\_Expense \_Expense;

**select** **forUpdate** \_Expense **where** \_Expense.RecId == \_id;

**if** (\_Expense !=**null**)

{

**ttsBegin**;

\_Expense.Status= \_status;

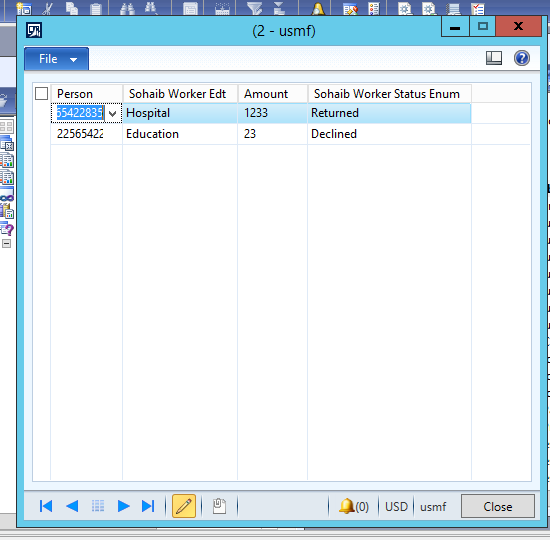
\_Expense.update();

**ttsCommit**;

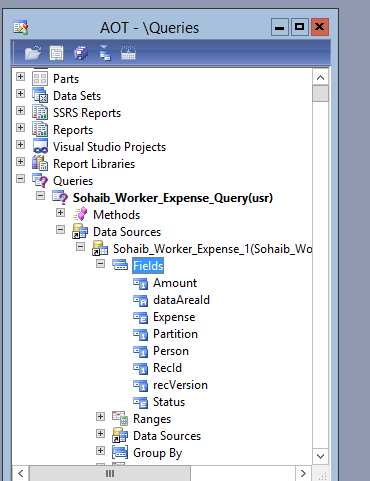
}

}

# Now we are creating Form:

Form create simple form with Name **Sohaib\_worker\_Form** (specify your name) and set its data source **Sohaib\_worker\_table** which you make. Add grid on form and drag and drop fields from data source to grid. Save it.  Form structure will be look like. Go to the grid properties and set data source their too.   
  
****

# Query

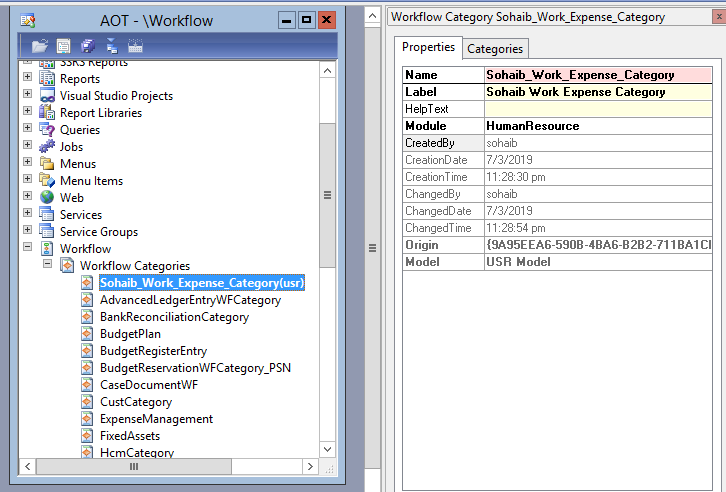
Now create a Query and This query will later use when we build Workflow type on it. Add **Sohaib\_Worker\_table** in it and set Data Source field dynamic to yes.  
  


# Creating Workflow Category:

Go to AOT Search for option workflow than expand the workflow. Go to workflow Category make new workflow Category.

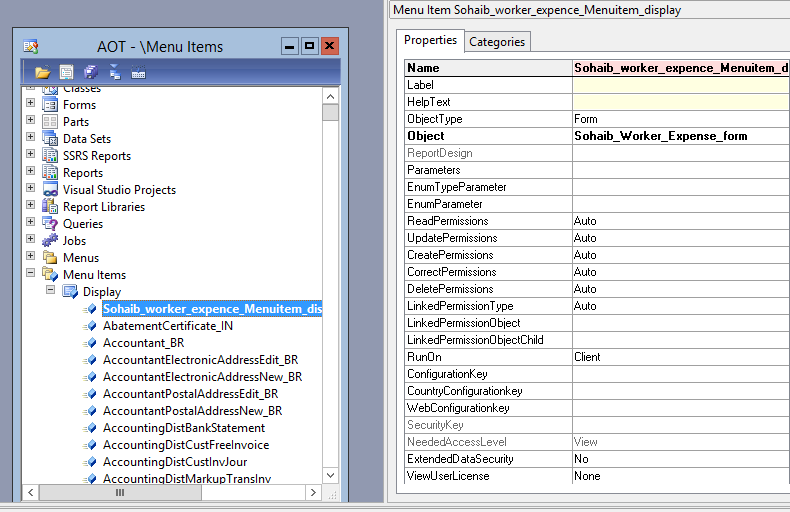
Go to the properties:

Set Module property as **Human resources.**



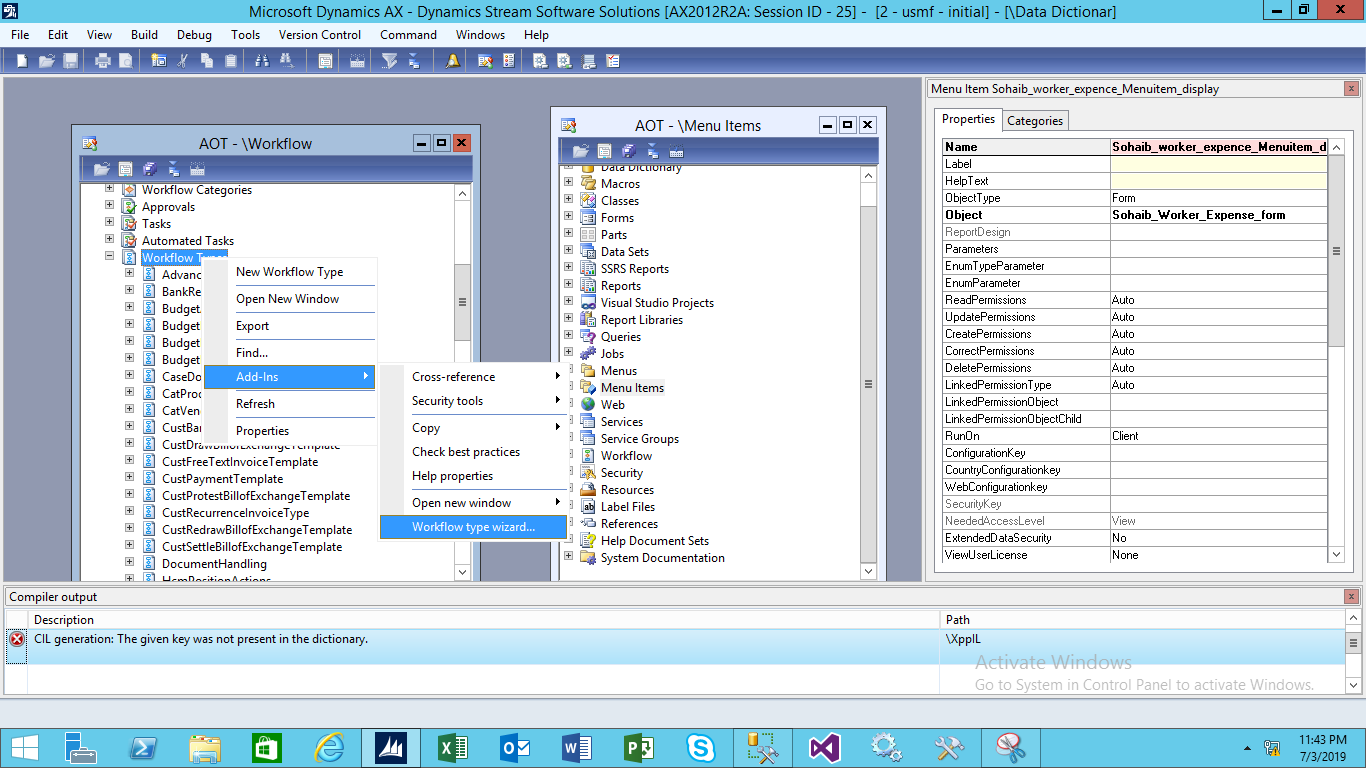
# Creating Menu Item Button:

Now go to AOT expand Menu item than add a new Menu Item button here. By right clicking on the display. Just add your form into the object property.

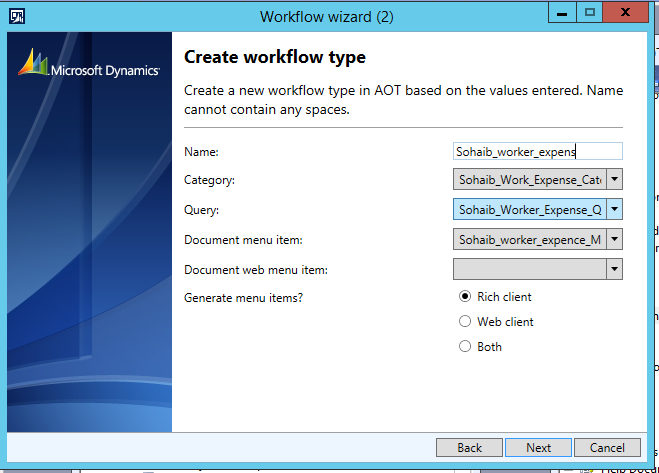


# Now add Workflow Type:

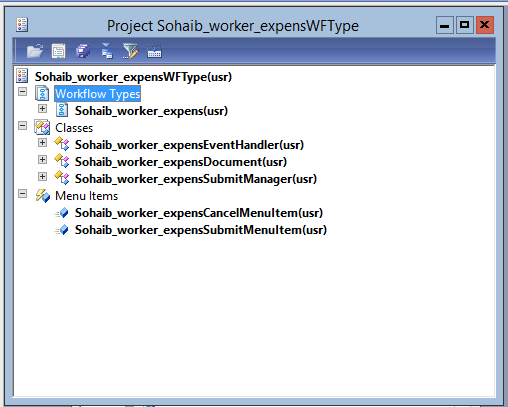
Go to AOT expand workflow property than go to workflow type and add new so the picture to do so.



Now just press next



After success you got the class like this.



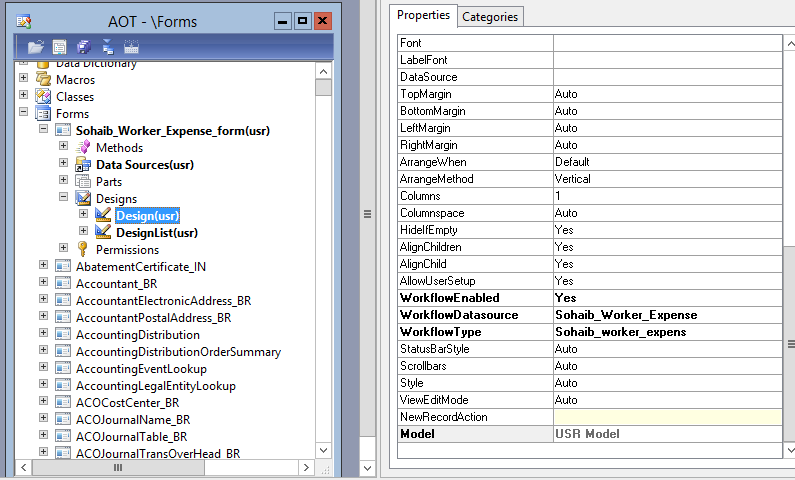
# Change properties setting:

Now expand the form (We created this in one above step) and expand its design, and set following properties  to it will workflow enable

WorkflowEnabled =yes

WorkflowDataSorce =Sohaib\_worker\_Expense (Your Table)

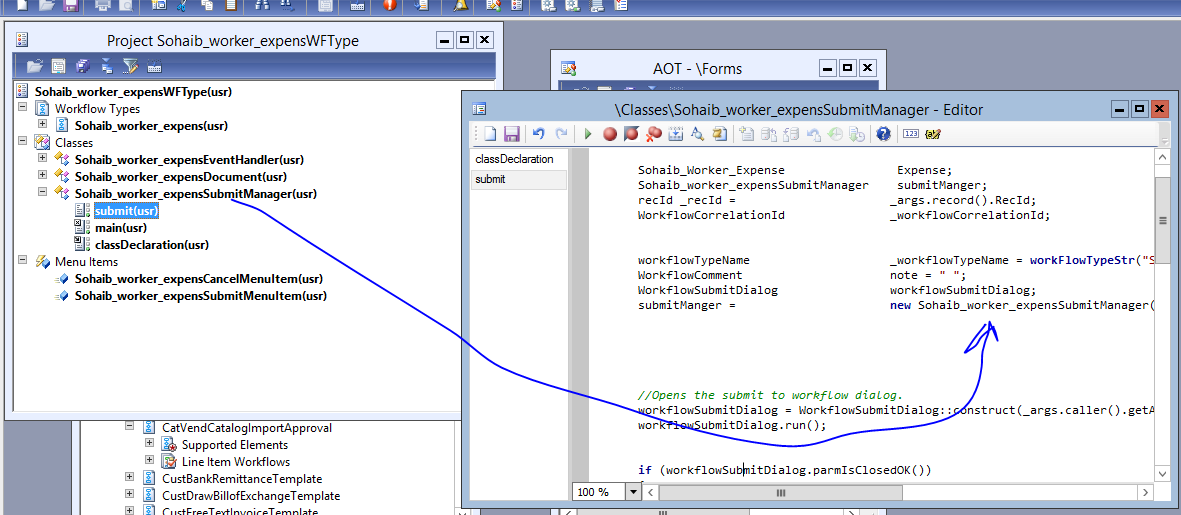
WorkflowType = Sohaib\_Work\_expens (Your Workflow Type)



# Now go to the new project which is created after workflow Type Execution:

Now expand submit manager class in workflow type project and update logic as follow.

And Create a new method with following logic.



## Code :

**public** **void** submit(Args \_args)

{

Sohaib\_Worker\_Expense Expense;

Sohaib\_worker\_expensSubmitManager submitManger;

recId \_recId = \_args.record().RecId;

WorkflowCorrelationId \_workflowCorrelationId;

workflowTypeName \_workflowTypeName = **workFlowTypeStr**("Sohaib\_worker\_expens");

WorkflowComment note = " ";

WorkflowSubmitDialog workflowSubmitDialog;

submitManger = **new** Sohaib\_worker\_expensSubmitManager();

*//Opens the submit to workflow dialog.*

workflowSubmitDialog = WorkflowSubmitDialog::construct(\_args.caller().getActiveWorkflowConfiguration());

workflowSubmitDialog.run();

**if** (workflowSubmitDialog.parmIsClosedOK())

{

Expense = \_args.record();

*// Get comments from the submit to workflow dialog.*

note = workflowSubmitDialog.parmWorkflowComment();

**try**

{

**ttsbegin**;

*// Activate the workflow.*

\_workflowCorrelationId = Workflow::activateFromWorkflowType(\_workflowTypeName, Expense.RecId, note, NoYes::No);

Expense.Status = Sohaib\_Woker\_Status\_Enum::Pending;

Expense.update();

**ttscommit**;

*// Send an Infolog message.*

info("WorkFlow Submitted");

}

**catch** (Exception::Error)

{

error("Error in workflow");

}

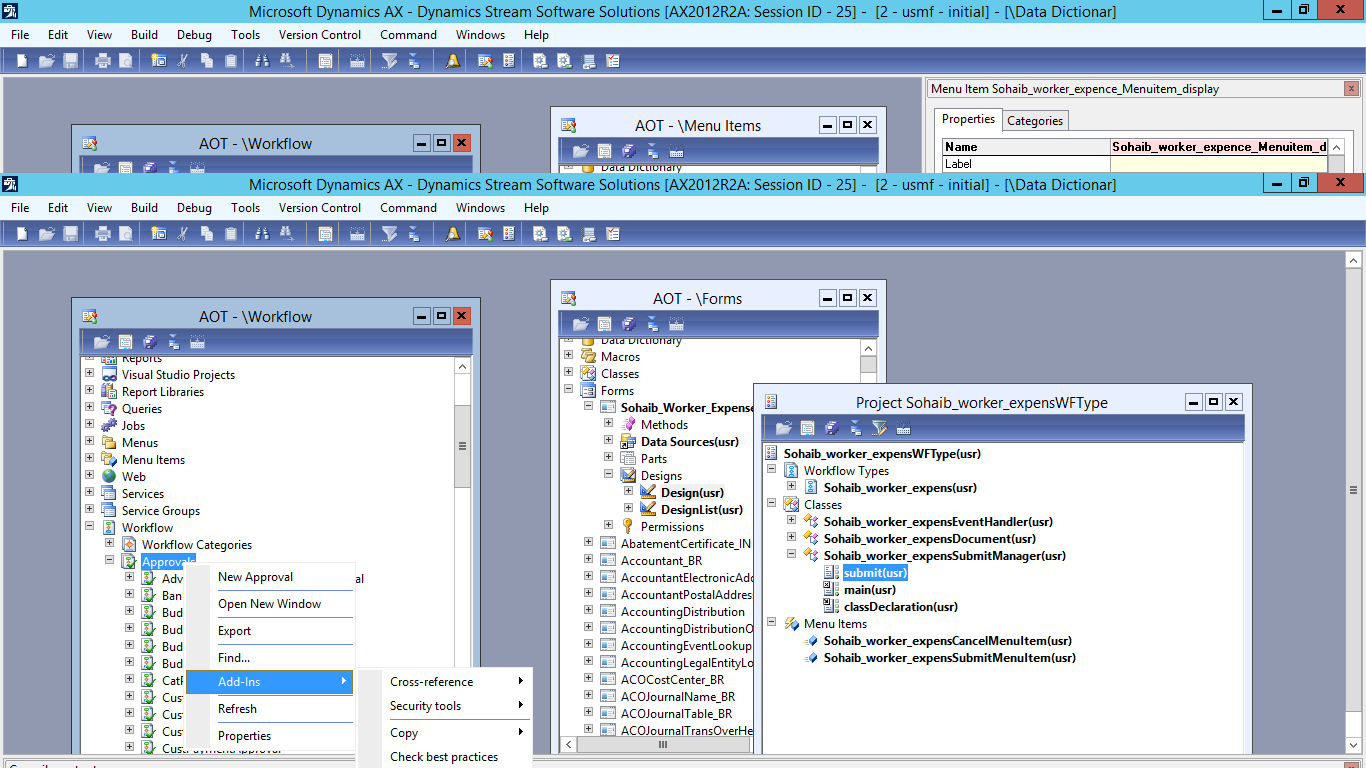
}

\_args.caller().updateWorkFlowControls();

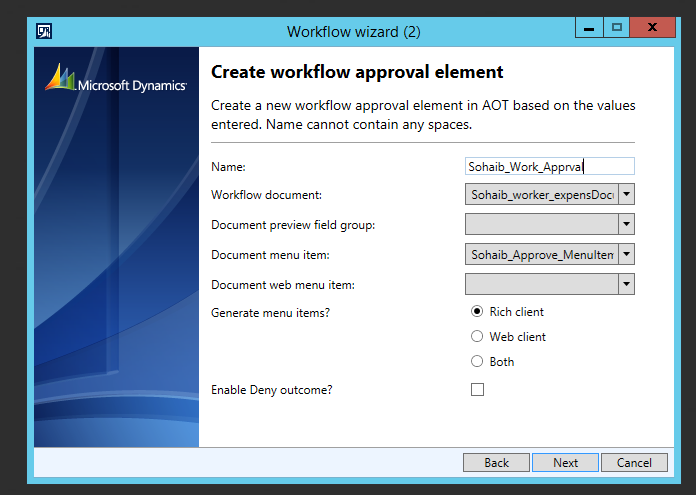
}

# Now we Create a workflow Approval:

Expand Workflow node in AOT and then again expand Workflow approval.

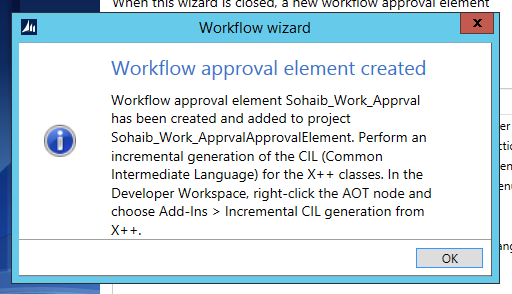


After doing setup same things as done in the upper workflow type



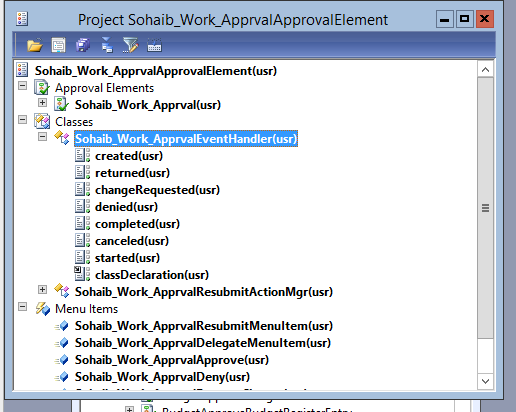
Document and menu item of approve will automatically appear as name of your workflow type

After clicking this setting just press next and finish.

****

After success you will get this screen.

# Now we have a new Project again (Approval Project)



Now after expanding the following class you have the created, returned, denied etc. functions.

## Override the following methods:

**public** **void** returned(WorkflowElementEventArgs \_workflowElementEventArgs)

{

*// TODO: Write code to execute once the workflow is returned.*

Sohaib\_Worker\_Expense::updateWorkFlowState(\_workflowElementEventArgs.parmWorkflowContext().parmRecId(), Sohaib\_Woker\_Status\_Enum::Returned);

}

**public** **void** changeRequested(WorkflowElementEventArgs \_workflowElementEventArgs)

{

*// TODO: Write code to execute once change is requested for the workflow.*

Sohaib\_Worker\_Expense::updateWorkFlowState(\_workflowElementEventArgs.parmWorkflowContext().parmRecId(), Sohaib\_Woker\_Status\_Enum::Returned);

}

**public** **void** denied(WorkflowElementEventArgs \_workflowElementEventArgs)

{

*// TODO: Write code to execute once the workflow is denied.*

Sohaib\_Worker\_Expense::updateWorkFlowState(\_workflowElementEventArgs.parmWorkflowContext().parmRecId(), Sohaib\_Woker\_Status\_Enum::Declined);

}

**public** **void** completed(WorkflowElementEventArgs \_workflowElementEventArgs)

{

*// TODO: Write code to execute once the workflow is completed.*

Sohaib\_Worker\_Expense::updateWorkFlowState(\_workflowElementEventArgs.parmWorkflowContext().parmRecId(), Sohaib\_Woker\_Status\_Enum::Accepted);

}

**public** **void** canceled(WorkflowElementEventArgs \_workflowElementEventArgs)

{

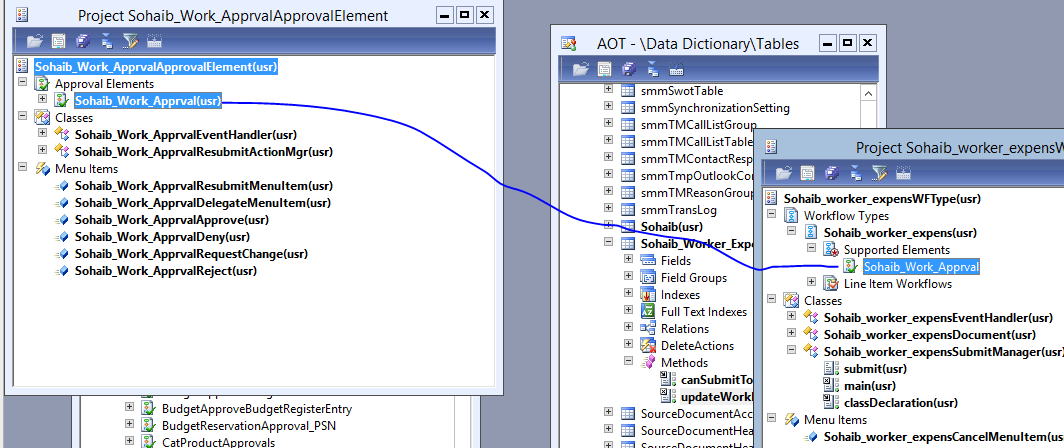
*// TODO: Write code to execute once the workflow is canceled.*

Sohaib\_Worker\_Expense::updateWorkFlowState(\_workflowElementEventArgs.parmWorkflowContext().parmRecId(), Sohaib\_Woker\_Status\_Enum::Declined);

}

## Drag Approval Artifacts to Workflow type Supported elements:

Now expand the workflow and in sub node Supported Elements, drag and drop approval artifact.



# Create New Menu Button:

Now create a new menu item give name (Sohaib \_Cust\_Approval) and set its form and set its following properties.

* Form **WorkflowTableListPage**
* EnumTypeParameter to ModuleAxapta
* EnumParameter to Human Resources.(For which module you are making)

## Open Menu item

 Now generate increment CIL. It is necessary step.

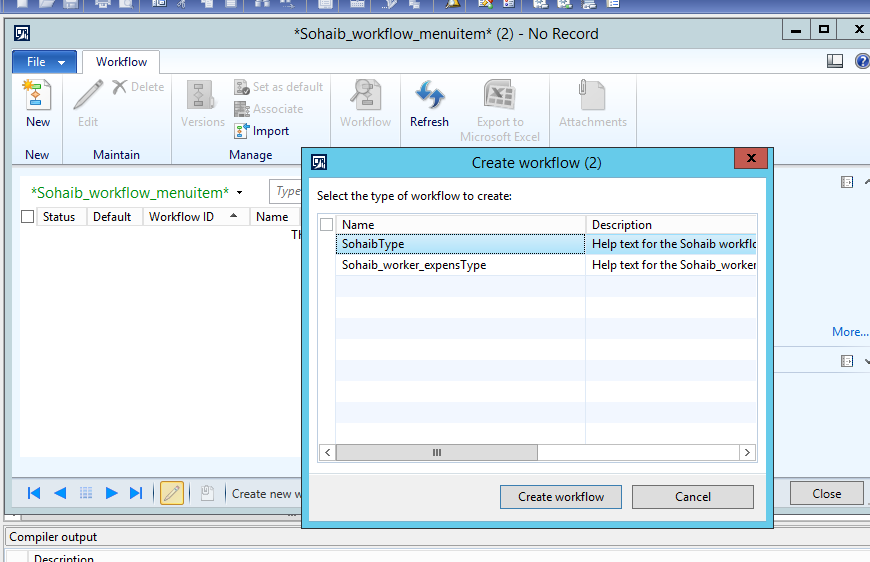
Right clicking on the menu item and open

Click on this and open the workflow design.

 Navigate to the menu item to open **WorkflowTableListPage** to design the workflow.

Click on new button.

Your Workflows will appear:

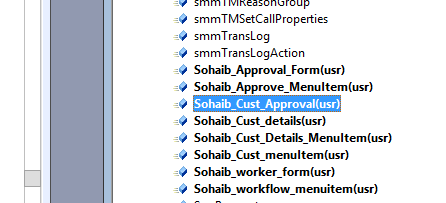


# Now we have to set up Confrigation of the Workflow

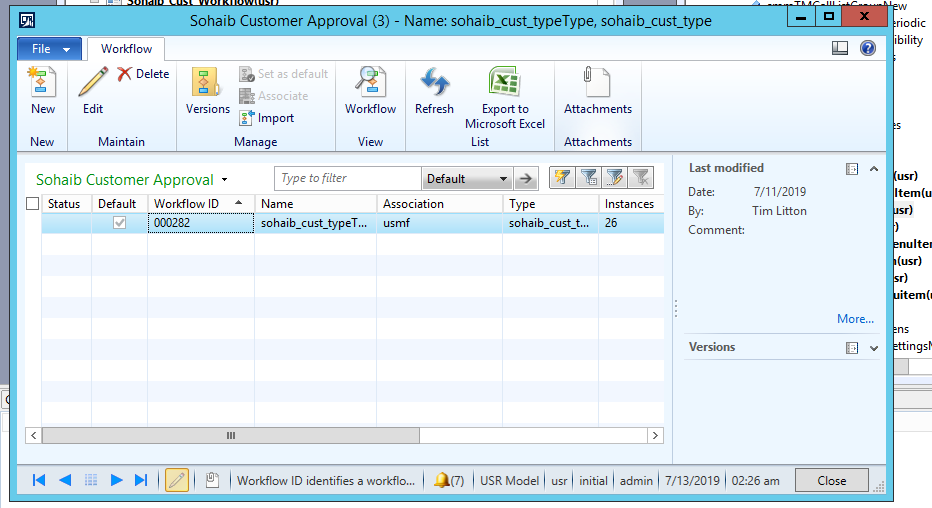
Go to the menu item

Go to Display

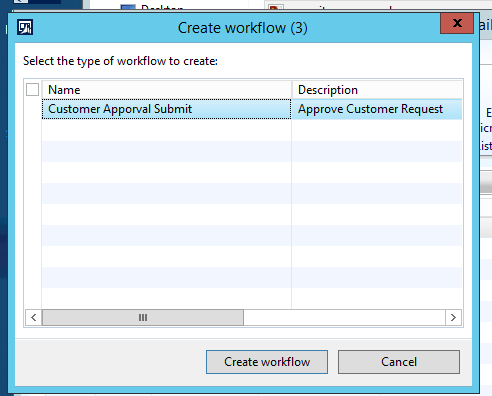
Open the form in which you specify the workflow List page in my case its Name is (Sohaib \_Cust\_Approval)



After opening the screen, you will get this type of screen

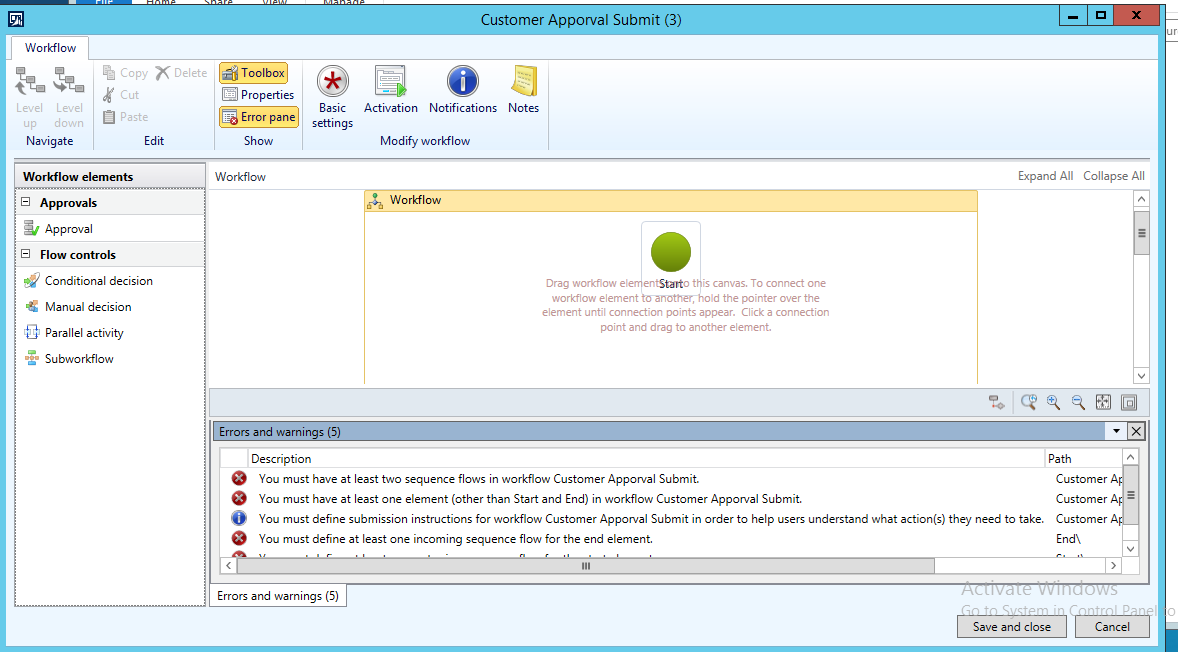


Now click on new. After clicking new you will get following screen



Here is your workflow which you have created. (If it’s not appearing here kindly check the module you specify in menu-button and workflow category it should be same ) (if still you workflow is not here than make confirm that you are login with a user that has permission for the module on which you are working for or simple log in through administrator account.)

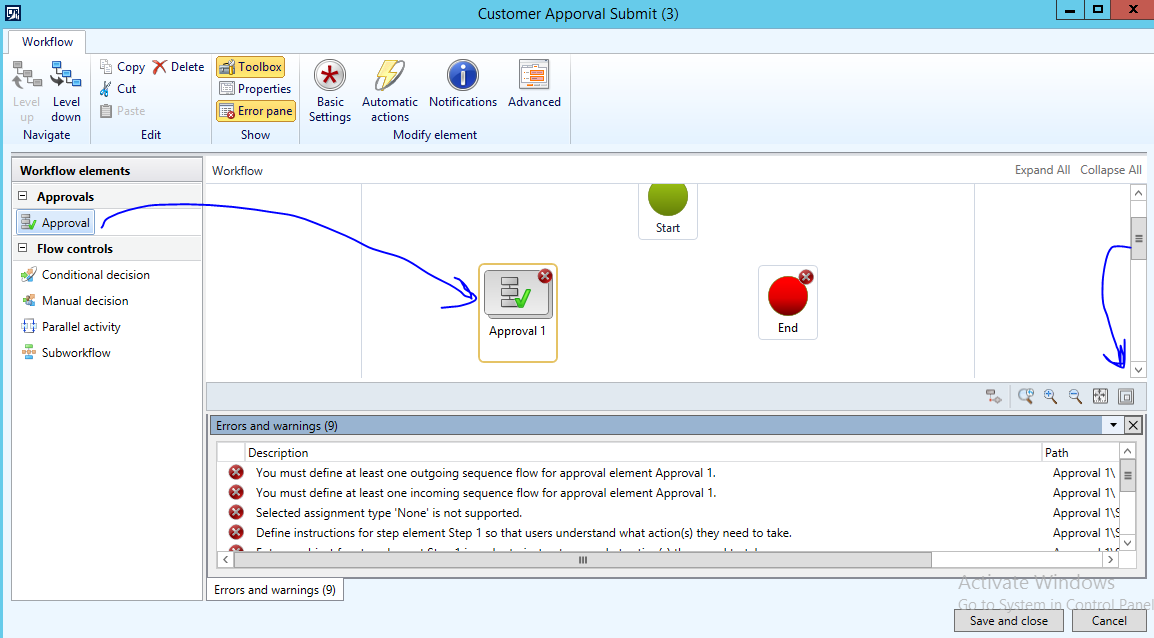
Now Click on Create Workflow:



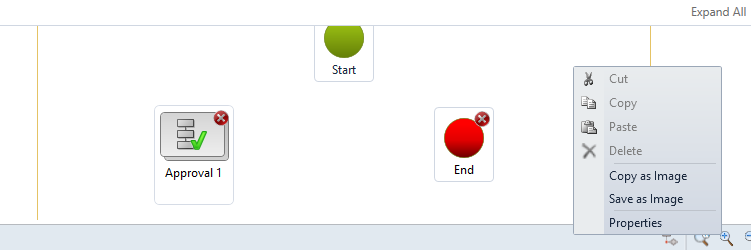
Now we have to resolve all of the errors below don’t worry its easy to resolve these errors.

Scroll down to the end you will find the end point of the workflow in red.

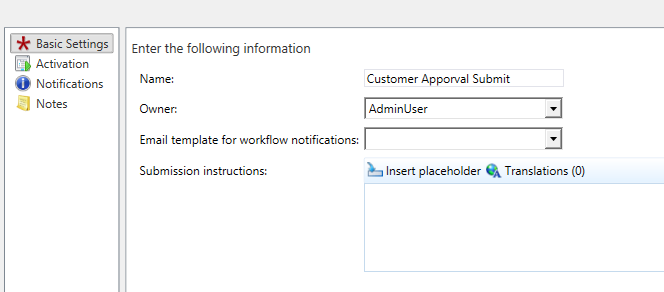
Now drag the approval to the screen to for creating a workflow



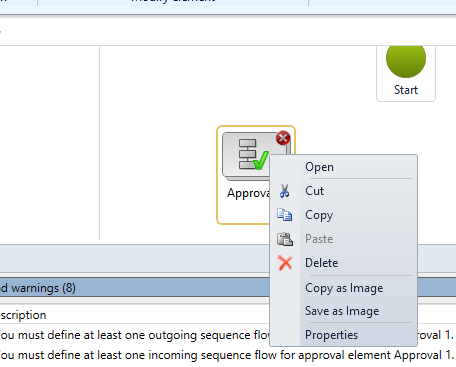
Now you have to go into the properties.



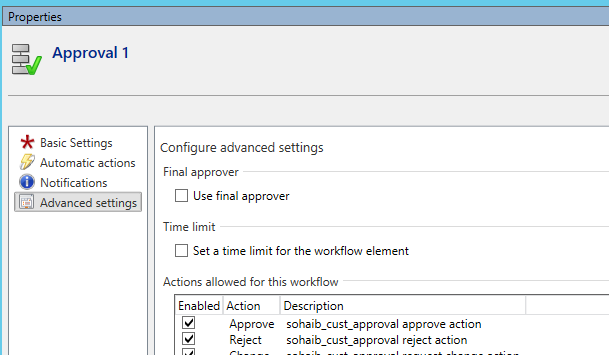
By default, you are the owner of the workflow if you want to set someone else owner of the workflow than set the owner.



Now go into the approval properties.

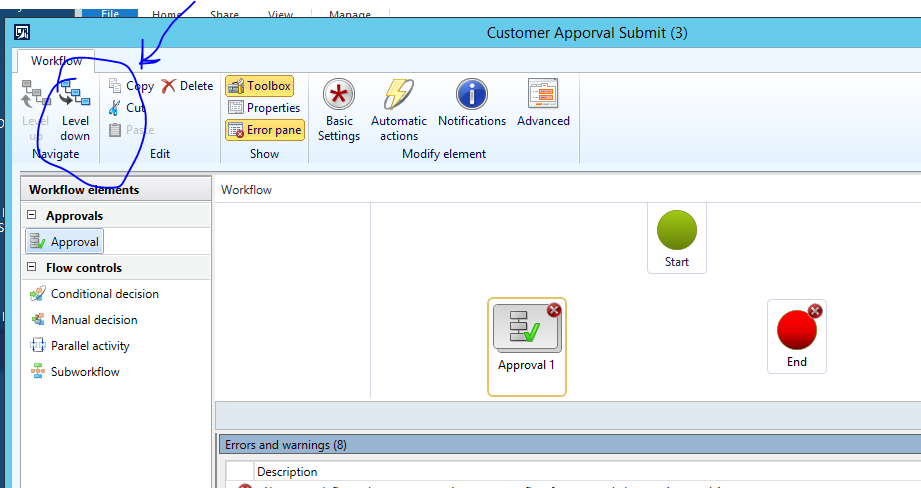


1. Now go into the advance settings
2. Tick mark the (Use final Approval) and select the User who will final Approver of the workflow.

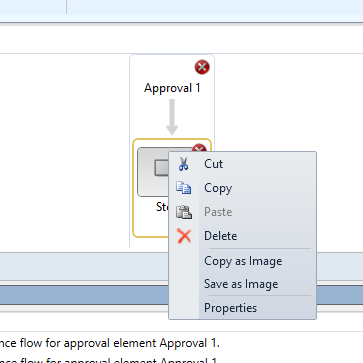


Now Close the form after selecting the final approver

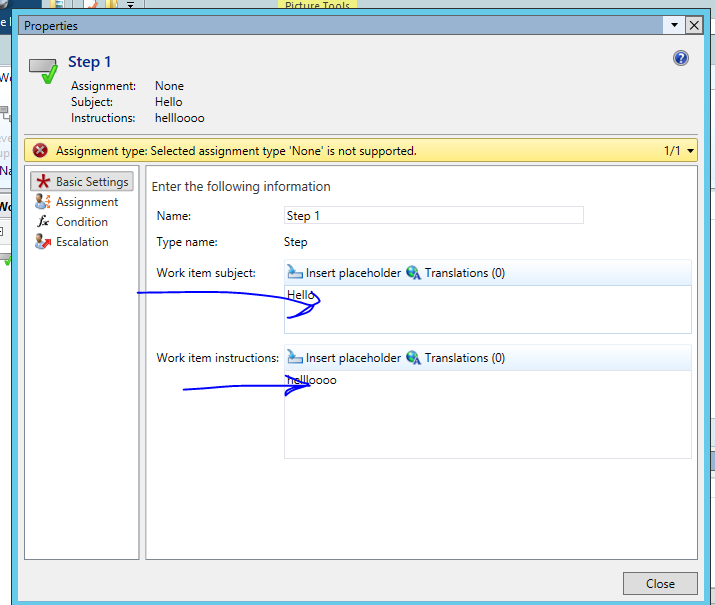
Now click on the Approver and click the level down Option



Go into the properties again

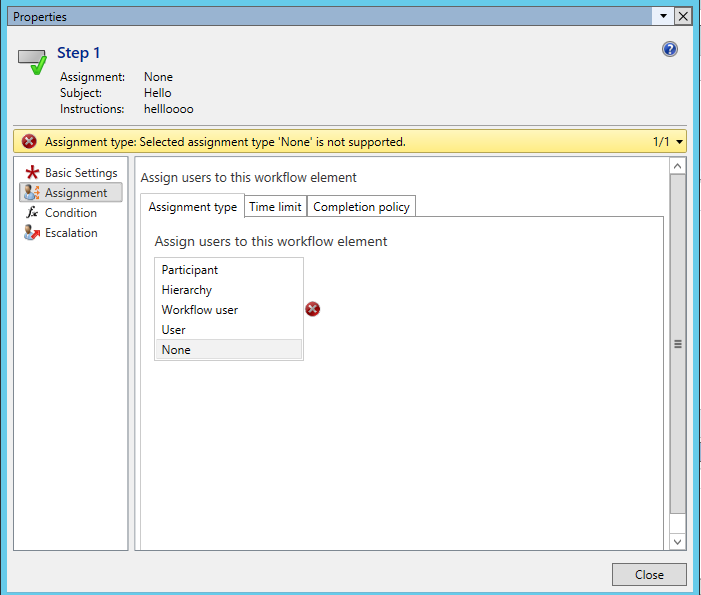


Fill the Instruction here

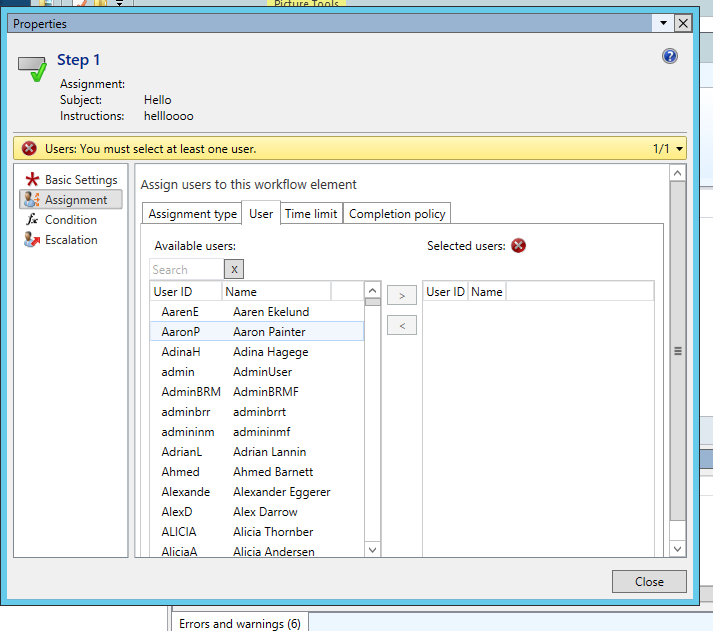


Now go to the assignment option

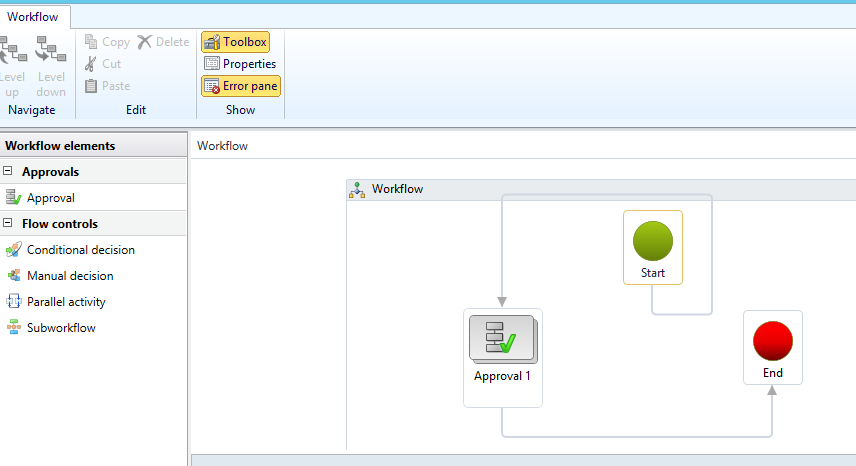
Select user here



Now a user tab will appear. Go to the User tab and select the user who will approve the workflow as much as you can.

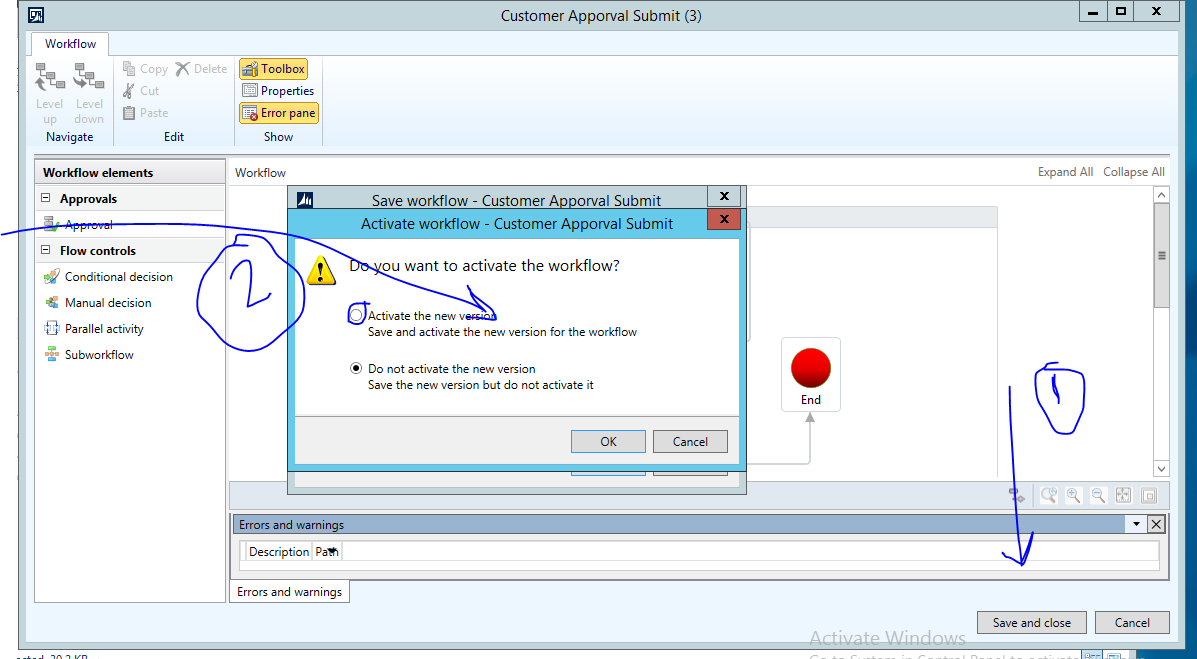


Now Close this and Connect the arrows after this you will see that all errors will disappear.

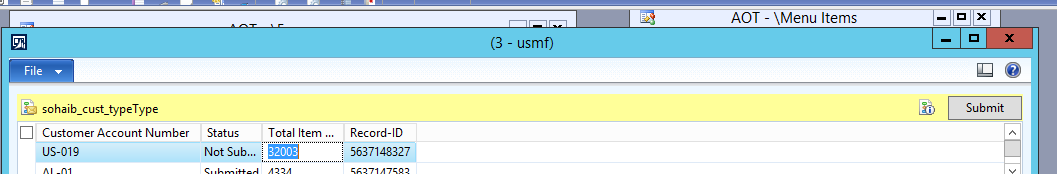


Now click on save and close.

And then activate the workflow like this.

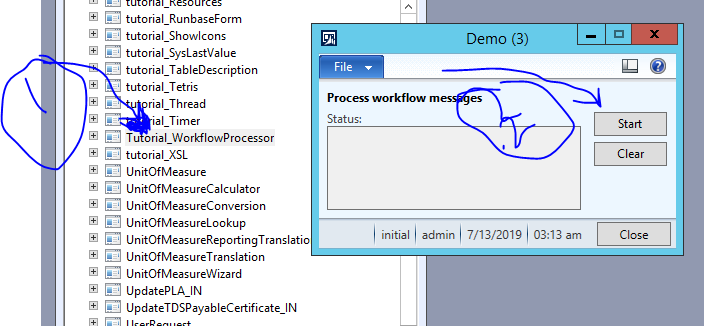


Now go to your form and add new records to it save it. You will see that the yellow bar will appear on the top. After submitting run he tutorial (Workflow Processor) to process the workflow request.

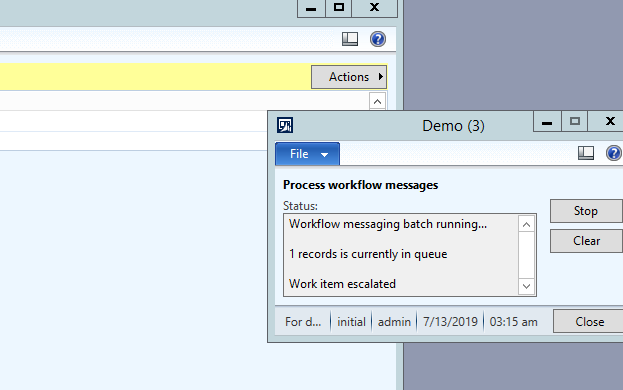


# Now run the Batch Job for the tutorial.

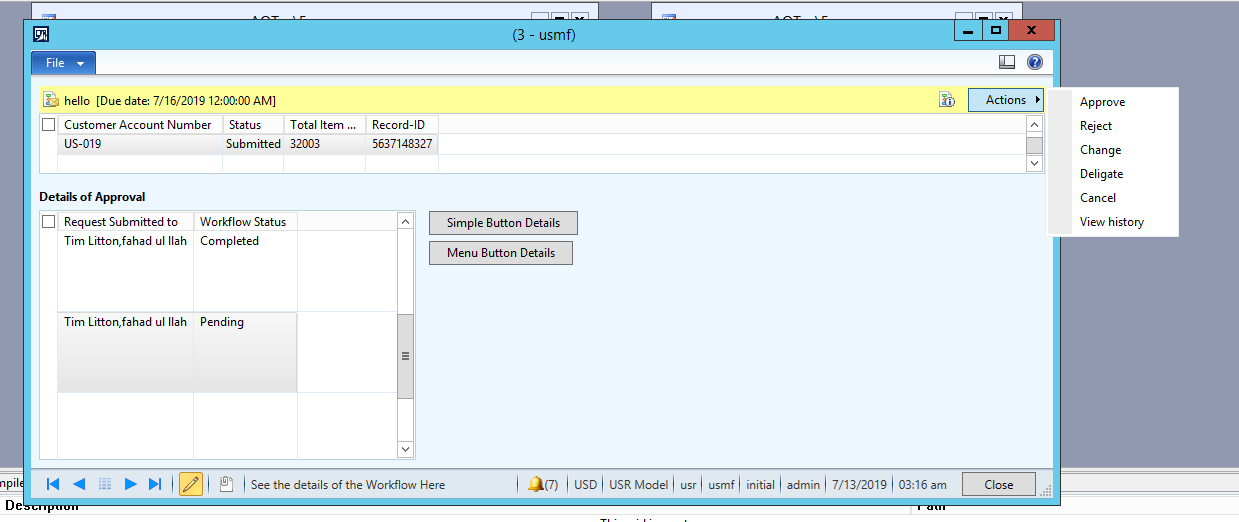
Go into the forms and search for Tutorial workflow processor. Start the services.



You can see that 1 record is processing



Now when your job is processed this will be your final results.



We will add below grid in the latter tutorial see (13 July Tutorial)

Below grid is showing all the information of the Workflow. (Approvals, owner of the workflow, Comments)