ORACLE® NETSUITE

SuiteFlow: Workflow Fundamentals

Student Workbook

This printing: May 2022

Oracle NetSuite
500 Oracle Parkway
Redwood Shores, CA 94065

TABLE OF CONTENTS

COURSE INTRODUCTION

Module Exercises	
01	Enabling SuiteFlow in your Training Account
02	Adjusting NetSuite Preferences
Optional Exercises	
03	Review the Course Case Study

01: Enabling SuiteFlow in your Training Account

Scenario: To create workflows in NetSuite, you must first verify that the SuiteFlow feature has been enabled in your SuiteDreams training account.

Enabling SuiteFlow in your Training Account

1 Navigate to **Setup > Company > Enable Features**.

2 Select the **SuiteCloud** subtab, locate the **SuiteFlow** section, and verify that SUITEFLOW is checked.

3 Click **Save** (if necessary).

02: Adjusting NetSuite Preferences

Scenario: In this optional exercise, you will adjust some basic user preferences that will be beneficial as you work through this course.

1 Navigate to **Home > Set Preferences**.

2 On the **General** subtab, adjust the following preferences:

Localization	
LANGUAGE	English (U.S.) or English (International)
TIME ZONE	Adjust to your local time zone. This will be particularly important when we discuss the workflow scheduler.
Formatting	
DATE FORMAT	Adjust to your desired date format. This is useful when creating conditions referencing date/time criteria.
Defaults	
SHOW INTERNAL IDS	Checked This allows you to view the internal IDs for fields and records in NetSuite. We will discuss the importance of this later in the course.

3 On the **Analytics** subtab, check the SHOW LIST WHEN ONLY ONE RESULT checkbox (if not already checked). This will display a list when there is only a single record rather than opening the record in View mode.

Review and set other preferences, as necessary.

4 Click **Save** when finished.

03: Review the Course Case Study (Optional)

Scenario: In this optional exercise, you will download and review the Course Case Study document located in the File Cabinet in your NetSuite training account.

1. Navigate to **Documents > Files > File Cabinet**.
2. Select the *Workflow Fundamentals* folder and download the "Understanding the Case Study.pdf" file.

   This document providers an overview of the business requirements, participants, and workflows you will be building throughout this course.

SuiteFlow Overview

Module Exercises	
01	Inspect the Custom Furniture Request Record
02	Create a New Furniture Request Record
03	Create a Simple Workflow

01: Inspect the Custom Furniture Request Record

Scenario: In this hands-on exercise, you will inspect the custom SuiteDreams Furniture Request record type which will be used as the foundation for developing workflows in this course.

1 Navigate to **Customization > List, Records, & Fields > Record Types**.

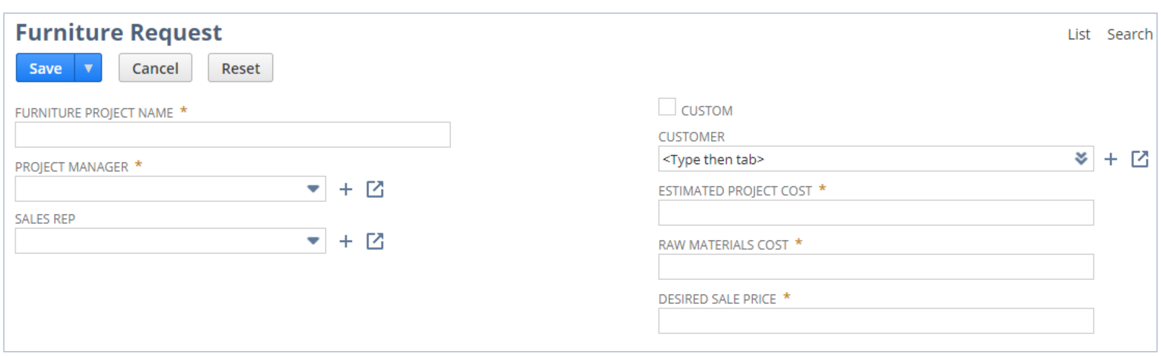
2 Click on the **Furniture Request** link to open the record type definition.

Note the record fields in the **Fields** subtab. You will see that there is a "join" out to the Customer record, as indicated by the Customer List/Record. You will be adding additional fields to this Furniture Request record in future exercises.

3 Hover over the **More** link in the upper-right corner of the page and click the **View Types** link to return to the list of custom record types.

4 Click the **New Record** link in the Furniture Request row. A new entry form will be displayed.

5 Inspect the fields on the record entry form:



Notice that there are currently no defaulted values or field validations.

6 Click the **List** link in the upper-right corner of the page to return to the list of Furniture Request records.

You can also view a list of all Furniture Request records by navigating to **Lists > Custom > Furniture Request**.

02: Create a New Furniture Request Record

Scenario: In this hands-on exercise, you will create a new Furniture Request record.

1. If you are not currently viewing a list of Furniture Request records, use one of the navigation methods above and click the **New Furniture Request** button.
2. Enter a name for the furniture being requested in the FURNITURE PROJECT NAME field.
3. Select a Product Manager from the PROJECT MANAGER dropdown list.
4. (**Optional**) Select a sales rep from the SALES REP dropdown list.
5. (**Optional**) There is no reason to check the CUSTOM checkbox at this time. You will manipulate other fields on the form based on the selection status of this checkbox in future exercises.
6. Select a customer from the CUSTOMER sublist, such as "ABC Marketing Inc".
7. Enter a value in the ESTIMATED PROJECT COST field.
8. Enter a value in the RAW MATERIALS COST field.
9. Enter a value in the DESIRED SALE PRICE field.
10. Click **Save**. The Furniture Request record will be saved and re-displayed in View mode.
11. Click the **List** link in the upper-right corner of the page to view the list of Furniture Request records.

    You can also hover your cursor over the **Recent Records** icon in the NetSuite navigation bar for a list of recently created/accessed records.

03: Create a Simple Workflow

Scenario: In this exercise, you will create a simple single-state workflow that will introduce you to the basic functionality of SuiteFlow and working with various workflow components, such as states and actions.

Create a New Workflow

1. Navigate to **Customization > Workflow > Workflows** and click **New Workflow**. The New Workflow form will be displayed.
2. Fill out the **Basic Information** and **Event Definition** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
NAME	Furniture Request Entry
SCRIPT ID	_sdr_furn_req_entry
RECORD TYPE	Furniture Request
DESCRIPTION	Enter a description of the workflow.
RELEASE STATUS	Released **Note**: During workflow development, RELEASE STATUS should be set to "Testing". This restricts workflow execution to the workflow owner. For most workflows created during this course, you will be setting it to "Released" to make it publicly available. This will allow users (regardless of user role) to execute the workflow.
KEEP INSTANCE AND HISTORY	Always
ENABLE LOGGING	Checked
Event Definition	
ON CREATE	Checked
TRIGGER TYPE	Before Record Load

3. Click **Save**.

The Workflow Manager will be displayed in Edit mode showing a single default state called "State 1". You will know you are in Edit mode if you see a View button above the Workspace.

Modify the Default State

Whenever you create a new workflow, a default state named "State 1" will be created automatically. To make the new state more meaningful, let's modify the name of the default state.

4 Edit "State 1" by selecting the state in the Workflow Manager and clicking the edit link represented by the pencil icon ( ) under the **State** tab, as shown below. You can also double-click on the state to open the Workflow State form.

5 Change the name of the state to "State 1: Data Entry". You can also enter a description of the state, if desired.

6 Click **Save** when finished.

Add a Show Message Action

Now let's add a simple Show Message action that automatically displays a message to the user when a new Furniture Request record is created.

7 While still in edit mode, select State 1: Data Entry in the Workflow Manager, and click **+ New Action** at the bottom of the **State** tab.

8 From the list of available actions, click the **Show Message** link.

9 Fill out the **Parameters** section (leaving all other fields at their defaults unless otherwise noted):

Parameters	
TEXT	You must select a customer when creating a custom Furniture Request.

10 Click **Save**.

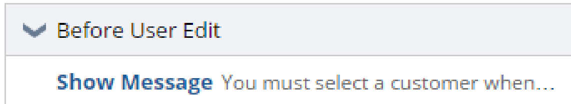
11 Double-click on "State 1: Data Entry" and you will see your new Show Message action listed under the **Actions** subtab.

> Rather than clicking **+ New Action** from the **State** tab to create a new action, you can click the **New Action** button under the **Actions** subtab when editing a state.

12 Click **Cancel** on the Workflow State form.

13 In the Workflow Manager, select (but do not edit) State 1: Data Entry once again.

You should see your new Show Message action listed in the **State** tab under the Before User Edit client trigger:

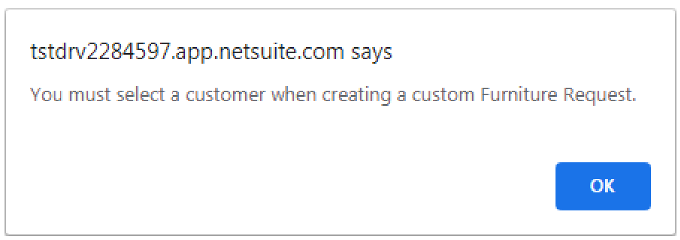


Execute the Workflow and Test

To see the fruits of your labor (displaying the Show Message action), let's execute your new workflow by creating a new Furniture Request record.

14 Navigate to **Lists > Custom > Furniture Request > New**. Try opening the entry form in a new browser tab.

Your workflow executes and immediately displays the configured Show Message action:



Notice that this message is displayed before you have a chance to interact with the form itself.

15 Click **OK** to acknowledge the message.

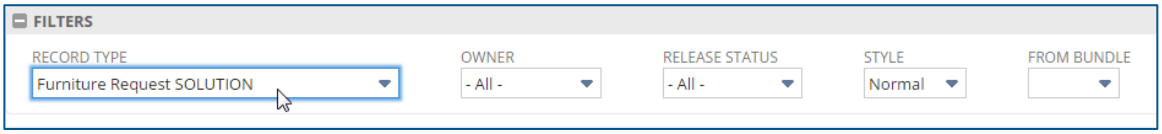
16 (**Optional**) Complete the entry of a new Furniture Request record.

EXERCISE SOLUTIONS

Having trouble with this module's workflow?

1 Navigate to **Customization > Workflow > Workflows**.

2 Change the RECORD TYPE filter to "Furniture Request SOLUTION".



3 Click **Edit** next to the workflow you would like to analyze:

- SOL Furniture Request Entry 02 (*Exercise 03*)

Configuring Actions on Data Entry

Module Exercises	
01	Modify the Furniture Request Entry Workflow
02	Implement Actions and Conditions using Client Triggers
Optional Exercises	
03	Client Triggers and User Roles
04	Notify User of Missing Email Address

01: Modify the Furniture Request Entry Workflow

Scenario: Let's return our focus back to the case study and SuiteDreams' business requirements. SuiteDreams would like to enhance the user experience and enforce business rules when creating and updating Furniture Request records.

In the following exercises, you will create a single-state workflow and implement a series of actions and conditions using only **client triggers**.

When testing actions configured on client triggers, it can be beneficial to clear your browser's cache when testing client-side actions. This is done by pressing **CTRL+F5** (Windows) or **CMD+R** (Mac OS).

Remove the Show Message Action

To avoid having the Show Message action execute every time we create or update a new Furniture Request record, let's remove the action from the Furniture Request Entry workflow.

1. If you are not already viewing the list of workflows, navigate to **Customization > Workflow > Workflows**.
2. Click on the **Edit** link next to the "Furniture Request Entry" workflow.
3. Select (but do not edit) State 1: Data Entry.
4. Hover your mouse over the Show Message action and click on the trash can icon ( ) to delete the action.

Note: You could have also simply inactivated the action by editing the Show Message action and checking the INACTIVE checkbox.

Modify "State 1: Data Entry"

5 Select State 1: Data Entry and click on the edit link represented by the pencil icon ( ) located in the **State** tab.

6 On the Workflow State form, change the name of the state to "State 1: Furniture Request Entry".

7 Enable the DO NOT EXIT WORKFLOW checkbox.

This ensures the workflow remains running on all records on which it is initiated. We will discuss this option in more detail later in this course.

8 Click **Save**. Your workflow should now resemble the following:

02: Implement Actions and Conditions using Client Triggers

Scenario: In this exercise, you will practice configuring actions and conditions using client triggers. SuiteDreams has provided specific requirements they would like to see implemented to facilitate the entry of new Furniture Requests. Evaluate each of the business rules below and identify the condition(s) that specify when each action should be executed.

Business Rule	Action/Condition Configuration
A	Default ESTIMATED PROJECT COST to $1,000. **Action**: Set Field Value **Client Trigger**: Before User Edit
B	Make the CUSTOMER field mandatory when CUSTOM is checked, otherwise CUSTOMER should be optional. **Action**: Set Field Mandatory **Client Trigger**: After Field Edit
C	Display an error message to the user upon record save when the ESTIMATED PROJECT COST is greater than $10,000. **Action**: Return User Error **Client Trigger**: Before User Submit

Test each action as you configure them to verify that each action executes correctly. This is a good habit to get into when developing workflows of increasing complexity.

Implement Business Rule A

Default ESTIMATED PROJECT COST on new Furniture Request records to $1,000.

Remember to use the **Make Copy** option and change the necessary settings for the new Set Field Value action.

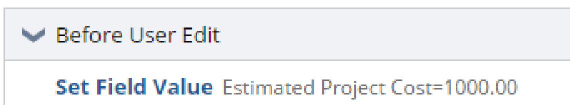
1. Make sure you are in Edit mode for the "Furniture Request Entry" workflow.
2. With State 1: Furniture Request Entry selected in the Workflow Manager, click **+ New Action**.
3. Select the **Set Field Value** action by clicking on its link.

4 Fill out the **Basic Information**, **Parameters**, and **Value** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Before User Edit
Parameters	
FIELD	Estimated Project Cost
Value (Static Value)	
TEXT	1000

5 Click **Save** on the Workflow Action configuration form.

In the list of actions under the **State** tab, you should see the following:



Test Business Rule A

6 Test the action by creating a new Furniture Request record by navigating to **Lists > Custom > Furniture Request > New**.

ESTIMATED PROJECT COST should default to 1000.00.

Implement Business Rule B

Make the CUSTOMER field mandatory when CUSTOM is checked, otherwise CUSTOMER should be optional.

7 Re-edit the Furniture Request Entry workflow.

8 With State 1: Furniture Request Entry still selected, click **+ New Action**.

9 Select the **Set Field Mandatory** action by clicking on its link.

10 Fill out the **Basic Information** and **Parameters** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	After Field Edit
TRIGGERING CLIENT FIELDS	Custom
Parameters	
FIELD	Customer
MANDATORY	Checked

Click on the icon to open the Visual Builder. Configure the following condition:

Condition	
FIELD	COMPARE TYPE
Custom	checked

11 Click **Add** and then **Save** on the Workflow Condition form.

12 Click **Save** on the Workflow Action configuration form.

Now let's repeat the previous steps to make the CUSTOMER field optional when CUSTOM is unchecked.

13 With State 1: Furniture Request Entry still selected, click **+ New Action**.

14 Select the **Set Field Mandatory** action by clicking on its link.

15 Fill out the **Basic Information** and **Parameters** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	After Field Edit
TRIGGERING CLIENT FIELDS	Custom
Parameters	
FIELD	Customer
MANDATORY	Unchecked

16 Click on the icon to open the Visual Builder. Configure the following condition:

Condition	
FIELD	COMPARE TYPE
Custom	not checked

17 Click **OK** then **Save** on the Workflow Condition form.

18 Click **Save** on the Workflow Action configuration form. In the list of actions under the **State** tab, you should see the following two Set Field Display Type actions:

After Field Edit
Set Field Mandatory Customer = True
Set Field Mandatory Customer = False

Test Business Rule B

19 Use the previous, unsaved record form (remembering to refresh your browser cache using **CTRL+F5**/**CMD+R** or create a new Furniture Request record (**Lists > Custom > Furniture Request > New**).

20 Check CUSTOM. The CUSTOMER field should be made mandatory.

21 Uncheck CUSTOM. The CUSTOMER field should be made optional.

Implement Business Rule C

Display an error message to the user upon record save when the ESTIMATED PROJECT COST is greater than $10,000.

22 Once again, edit the Furniture Request Entry workflow.

23 With State 1: Furniture Request Entry still selected, click **+ New Action**.

24 Select the **Return User Error** action by clicking on its link.

25 Fill out the **Basic Information** and **Parameters** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Before User Submit
Parameters	
TEXT	Furniture Requests cannot be more than $10,000.

26 Click on the icon to open the Visual Builder. Configure the following condition:

Condition		
FIELD	COMPARE TYPE	VALUE
Estimated Project Cost	greater than	10000

27 Click **Add** and then **Save** on the Workflow Condition form.

28 Click **Save** on the Workflow Action configuration form. In the list of actions under the **State** tab, you should see the following:

Test Business Rule C

29 Use the previous, unsaved record form (remembering to refresh your browser cache) or create a new Furniture Request record.

30 Complete the Furniture Request entry form and change the ESTIMATED PROJECT COST to a value greater than 10,000.

31 Click **Save**.

You should receive an error that "Furniture Requests cannot be more than $10,000".

32 (**Optional**) Correct the error and re-save the record.

03: Client Triggers and User Roles (Optional)

Scenario: Users running your workflows must sometimes have access to data in other related records when executing actions on client triggers. To do so, it requires access to the Customer record. All user roles interacting with Furniture Request records must also have access to the Customer record for this reason.

By default, all user roles interacting with Furniture Requests do not have access to the Employee record, but there are specific user roles that should have access to the Employee record. This exercise implements a business rule granting access to the Employee record by configuring a *join on a related record*—in this case, to the Employee record.

Business rules to implement:

Business Rule	Action/Condition Configuration
D	Restrict Sales Reps to employees only. Employees flagged as "Contractors" should not have access. To determine whether a Sales Rep is a contractor, navigate to the selected Sales Rep's employee record and evaluate the TYPE field located on the **Human Resources** > **Job Information** section. If the selected employee is a contractor, display the following message: *"Sales reps cannot be contractors. Please select a different rep."* **Action**: Return User Error **Client Trigger**: Before User Submit

> You may need to set TYPE = "Contractor" on the Employee record(s) that you plan to select as a Sales Rep during testing. Also, make sure to select an employee in the "US – West Subsidiary", as these are the only employees that can be selected when switching to the role identified in the steps below.

Edit the Furniture Request Entry Workflow

1 Navigate to **Customization > Workflow > Workflows** and click on the **Edit** link next to the Furniture Request Entry workflow.

Implement Business Rule D

2 Select (but do not edit) State 1: Furniture Request Entry and click **+ New Action**.

3 Select the **Return User Error** action by clicking on its link.

4 Fill out the **Basic Information** and **Parameters** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Before User Submit
Parameters	
TEXT	Sales reps cannot be contractors. Please select a different sales rep.

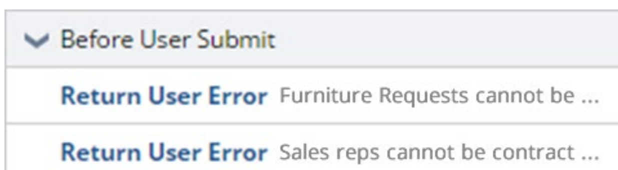
5 Click on the icon to open the Visual Builder. Configure the following condition:

Condition			
RECORD	FIELD	COMPARE TYPE	SELECTION
Sales Rep	Employee Type	any of	Contractor

6 Click **Add** and then **Save** on the Workflow Condition form.

7 Click **Save** on the Workflow Action configuration form.

In the list of actions under the **State** tab, you should see the following Return User Error actions:



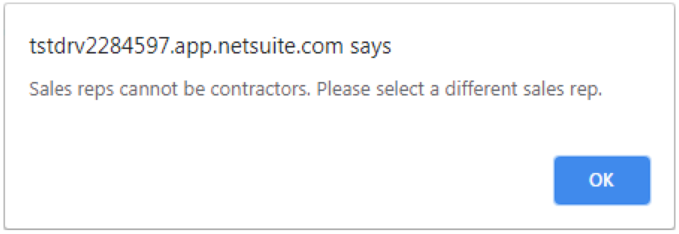
Test Business Rule D

8 Using your **Administrator** role (Larry Nelson), create a new Furniture Request record.

9 Select "Dale Sullivan" from the SALES REP dropdown list. Dale already has an Employee Type of "Contractor" established on his Employee record.

10 Complete the rest of the form fields and click **Save**.

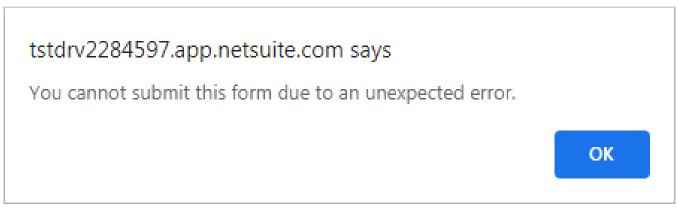
The following error message should be displayed:



11. Switch to your **Sales Rep – US West** role and create a new Furniture Request record by navigating to **Customers > Other > Furniture Request > New**.

12. Once again, select "Dale Sullivan" from the SALES REP dropdown list, complete the rest of the form fields, and click **Save**.

    The following error message is returned:



    While the error message text does not provide much detail, it is displayed because of a permission issue to the Employee record for our non-Administrator user role, "Sales Rep – US West".

Implement Workaround

The workaround is to create a custom field that sources from the EMPLOYEE TYPE field in the Sales Rep record. You can then use the sourced-in field in the condition of the Return User Error action. Unlike record joins, sourced-in fields do not require special permissions.

13. Switch back to your **Administrator** role.

Let's create a custom field on the Furniture Request record that sources from the EMPLOYEE TYPE field of the Sales Rep record.

14. Navigate to **Customization > Lists, Records, & Fields > Record Types** and select the "Furniture Request" record type.

15 In the **Fields** subtab, click **New Field** to add a field representing Employee Type:

LABEL	Employee Type
ID	_sdr_furn_req_part_emp_type
TYPE	List/Record
LISTS/RECORD	Employee Type
STORE VALUE	Unchecked

DISPLAY TYPE (Display subtab)	Hidden **Note**: You could also specify DISPLAY TYPE = "Inline Text". It depends if this field is only to support the workaround or if it should actually display on the Furniture Request record.
SOURCE LIST (Sourcing & Filtering subtab)	Sales Rep
SOURCE FROM (Sourcing & Filtering subtab)	Employee Type

16 Click **Save**.

Now modify the condition by evaluating directly against the value of the EMPLOYEE TYPE field.

17 Navigate to **Customization > Workflow > Workflows** and edit the Furniture Request Entry workflow.

18 Select State 1: Furniture Request Entry and edit the "Sales reps cannot be contractors…" Return User Error action under the Before User Submit trigger.

19 Modify the condition by clicking the icon to open the Visual Builder.

20 You can remove the existing condition by selecting it in the Visual Builder and clicking **Remove** and then redefine the condition as follows:

Condition		
FIELD	COMPARE TYPE	SELECTION
Employee Type	any of	Contractor

21 Click **OK** and then **Save** on the Workflow Condition form.

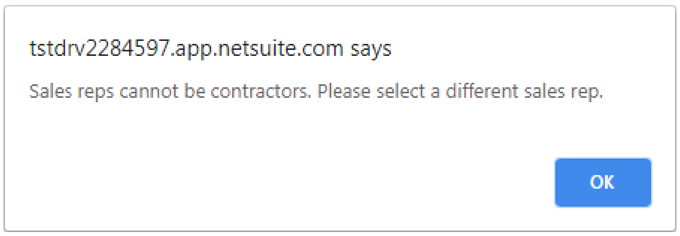
22 Click **Save** on the Workflow Action configuration form.

Re-Test Business Rule D

23 Using your **Administrator** role, create a new Furniture Request record (**Lists > Custom > Furniture Request > New**).

24 In the SALES REP dropdown list, select "Dale Sullivan".

You should receive the following error message:



25 Switch back to the **Sales Rep – US West** role and create a new Furniture Request (**Customers > Other > Furniture Request > New**).

26 Select "Dale Sullivan" once again.

You should now receive the same error message above.

04: Notify User of Missing Email Address (Optional)

Scenario: In this optional exercise, you will create a workflow running on a Support Case record. Support cases are accessed at **Lists > Support > Cases**. The Support Case record contains an email address field that is sourced-in from the related Customer record.

Business rules to implement:

Business Rule	Action/Condition Configuration
E	Display an informational message to the user about the missing email when the selected customer's email address is empty. **Action**: Show Message **Client Trigger**: After Field Sourcing

Create a New Workflow

1. Navigate to **Customization > Workflow > Workflows > New**.
2. Fill out the **Basic Information** and **Event Definition** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
NAME	Support Case Entry
SCRIPT ID	_sdr_support_case_entry
RECORD TYPE	Case
DESCRIPTION	Enter a description of the workflow.
RELEASE STATUS	Testing
KEEP INSTANCE AND HISTORY	Only When Testing
ENABLE LOGGING	Checked
Event Definition	
ON CREATE	Checked
ON VIEW OR UPDATE	Checked
TRIGGER TYPE	Before Record Load

3. Click **Save**.

Modify the Default State

4 Edit the default "State 1" by selecting the state then clicking the edit link represented by the pencil icon ( ).

5 Rename the state "State 1: Data Entry " and enable the DO NOT EXIT WORKFLOW checkbox.

6 Click **Save**.

Implement Business Rule E

7 In the Workflow Manager, select State 1: Data Entry and click **+ New Action**.

8 Select the **Show Message** action by clicking on its link.

9 Fill out the **Basic Information** and **Parameters** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	After Field Sourcing
TRIGGERING CLIENT FIELDS	Company
Parameters	
TEXT	Please obtain the company's email address.

10 Click on the icon to open the Visual Builder. Configure the following condition:

Condition	
FIELD	COMPARE TYPE
E-mail	empty

11 Click **Add** and then **Save** on the Workflow Condition form.

12 Click **Save** on the Workflow Action configuration form.

Test Business Rule E

13 Create a new Support Case by navigating to **Lists > Support > Cases > New**.

14 Enter a subject for the new case in the SUBJECT field.

15 Select a company with an email address ("ABC Marketing Inc") from the COMPANY dropdown list.

The Show Message action does not execute.

16 Now change the company to one without an email address ("Apeture").

The Show Message action executes.

17 Go back to your workflow and change the trigger on the Show Message action to "After Field Edit".

18 Repeat the steps above.

Regardless of which company you select, the Show Message action will be triggered before the value has even been sourced-in from the related Customer record.

EXERCISE SOLUTIONS

Having trouble with this module's workflow?

1. Navigate to **Customization > Workflow > Workflows**.
2. Change the RECORD TYPE filter to "Furniture Request SOLUTION".
3. Click **Edit** next to the workflow you would like to analyze:
   - SOL Furniture Request Entry 04 (*Exercises 01-02*)
   - SOL Furniture Request Entry 04-3 (*Optional Exercise 03*)
   - SOL Support Case Entry 04-4 (*Optional Exercise 04*)
     Note: You will need to change the RECORD TYPE filter to "Case" to see this solution workflow.

Configuring Actions on Record Load

Module Exercises	
01	Create a Workflow to Support Actions on Record Load
02	Implement Actions and Conditions using Server Triggers

01: Create a Workflow to Support Actions on Record Load

Scenario: SuiteDreams would like to enhance the user experience and enforce business rules when loading Furniture Request records. In this exercise, you will create a new workflow to support actions on record load.

Create a New Workflow

1 Navigate to **Customization > Workflow > Workflows > New**.

2 Fill out the **Basic Information** and **Event Definition** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
NAME	Furniture Request Load
SCRIPT ID	_sdr_furn_req_load
RECORD TYPE	Furniture Request
DESCRIPTION	Enter a description of the workflow.
RELEASE STATUS	Released **Note**: Allows non-Administrator users to execute the workflow.
KEEP INSTANCE AND HISTORY	Always
ENABLE LOGGING	Checked
Event Definition	
ON CREATE	Checked
ON VIEW OR UPDATE	Checked
TRIGGER TYPE	Before Record Load

3 Click **Save**.

Modify the Default State

4 Rename State 1 to "State 1: Furniture Request Load" and enable the DO NOT EXIT WORKFLOW checkbox.

5 Click **Save**.

You should have a single-state workflow that resembles the following:

02: Implement Actions and Conditions using Server Triggers

Scenario: In this exercise, you will practice configuring actions and conditions using server triggers. SuiteDreams has provided specific requirements they would like to see implemented to change the display characteristics of some of the fields on the Furniture Request entry form. Evaluate each of the business rules below and identify the condition(s) required to execute each action.

Specifying an event type is enough of a restriction for some of the business rules below, but additional conditions may be required. SuiteDreams business requirements to be implemented in this exercise:

Business Rule	Action/Condition Configuration
A	To establish display persistence of this action when editing a record, let's add the same Set Field Mandatory action to the Before Record Load server trigger. Make the CUSTOMER field mandatory when CUSTOM is checked, otherwise CUSTOMER should be optional. **Action**: Set Field Mandatory **Client Trigger**: Before Record Load
B	Default the current user as the PROJECT MANAGER for all new Furniture Requests. **Action**: Set Field Value **Server Trigger**: Entry
C	Change the display label of the CUSTOMER field to "Furniture Client" <u>only</u> when printing a Furniture Request record. **Action**: Set Field Display Label **Server Trigger**: Before Record Load
D	Users with an "Administrator" role should be able to create Furniture Request records. The only other users who should be allowed to create Furniture Request records are users in the "Installation/Fitout" department. Users who are not members of the Installation/Fitout department should receive the following message when attempting to create new Furniture Request records: *"You are not authorized to initiate furniture projects."* **Action**: Return User Error **Server Trigger**: Before Record Load

When choosing between the Entry or Before Record Load triggers, the question to ask yourself is, "Does the action need to occur every time a record is loaded or just the first time the workflow enters the state?"

When creating a new Furniture Request, the CUSTOMER field is made mandatory based on the value of the CUSTOM checkbox. However, this is only done during data entry using client triggers. To establish display persistence of this action when *editing* a record, let's add the same Set Field Mandatory action to the Before Record Load server trigger.

1 Create a new custom Furniture Request record.

  Notice that during record entry, checking CUSTOM makes the CUSTOMER field mandatory and unchecking CUSTOM makes CUSTOMER optional.

2 Check the CUSTOM checkbox, select a Customer from the dropdown list, and complete the remaining required fields and click **Save**.

3 Click **Edit** to open the record in Edit mode.

  When the record is loaded into the browser, you will see that the CUSTOMER field is not displayed as mandatory even though the CUSTOM checkbox is enabled.

Implement Business Rule A

To ensure proper display persistence of the Set Field Mandatory action, let's make the CUSTOMER field mandatory when CUSTOM is checked when editing an existing record, otherwise CUSTOMER should be optional.

4 Edit the Furniture Request Load workflow, select State 1: Furniture Request Load, and click **+ New Action**.

5 Select the **Set Field Mandatory** action by clicking on its link.

6 Fill out the **Basic Information** and **Parameters** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Before Record Load
Parameters	
FIELD	Customer
MANDATORY	Checked

7 Click on the icon to open the Visual Builder. Configure the following condition:

Condition	
FIELD	COMPARE TYPE
Custom	checked

8 Click **Add** and then **Save** on the Workflow Condition form.

9 Click **Save** on the Workflow Action configuration form.

Test Business Rule A

10 Create and save a new **custom** Furniture Request record and open the record in Edit mode.

The CUSTOMER field should now be displayed as mandatory since CUSTOM is checked.

Implement Business Rule B

Default the current user as the PROJECT MANAGER for all new Furniture Request records.

11 Re-edit the Furniture Request Load workflow and select State 1: Furniture Request Load.

12 Click **+ New Action**.

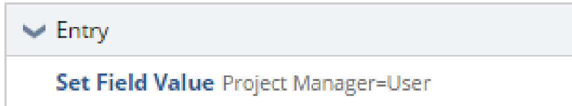
13 Select the **Set Field Value** action by clicking on its link.

14 Fill out the **Basic Information**, **Parameters**, and **Value** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Entry
EVENT TYPES	Create
Parameters	
FIELD	Project Manager
Value (From Field)	
RECORD	Current Record
FIELD	User

15 Click **Save** on the Workflow Action configuration form.

In the list of actions under the **State** tab, you should see the following:



Test Business Rule B

16 In a new browser tab/window, create a new Furniture Request record.

You (Larry Nelson) should automatically default as the PROJECT MANAGER.

Leave the unsaved record open for now and continue with the configuration of Business Rule C.

Implement Business Rule C

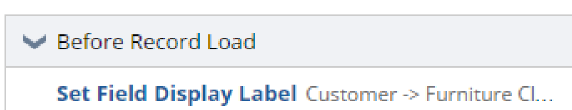
Change the display label of the CUSTOMER field to "Furniture Client" <u>only</u> when printing a Furniture Request record.

17 Return to the Furniture Request Load workflow and add a **Set Field Display Label** action to State 1: Furniture Request Load.

18 Fill out the **Basic Information** and **Parameters** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Before Record Load
EVENT TYPES	Print
Parameters	
FIELD	Customer
LABEL	Furniture Client

19 Click **Save** on the Workflow Action configuration form. In the list of actions under the **State** tab, you should see the following:



Test Business Rule C

20 Return to the open Furniture Request record created above, complete the required fields, and click **Save**.

21 From View mode, click the **Print** icon.

The CUSTOMER field should now be displayed as FURNITURE CLIENT.

Implement Business Rule D

Users with an "Administrator" role should be able to create Furniture Request records. The only other users who should be allowed to create Furniture Request records are users in the "Installation/Fitout" department.

Users who are not members of the Installation/Fitout department should receive the following message when attempting to create/copy Furniture Request records:

"You are not authorized to create furniture requests."

22 Edit the Furniture Request Load workflow and add a **Return User Error** action to State 1.

23 Fill out the **Basic Information** and **Parameters** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Before Record Load
EVENT TYPES	Copy, Create
Parameters	
TEXT	You are not authorized to create Furniture Requests.

24 Click on the icon to open the Visual Builder. Configure the following two conditions:

Condition		
FIELD	COMPARE TYPE	SELECTION
User Role	none of	Administrator

Note: Be sure to click **Add** after each condition.

Condition		
FIELD	COMPARE TYPE	SELECTION
User Department	none of	Installation/Fitout

25 Click **Save** on the Workflow Condition form.

26 Click **Save** on the Workflow Action configuration form.

Test Business Rule D

Before testing this business rule, you will need to grant login access to an employee in the Installation/Fitout department (Inger Brown) and another employee who is not a member of the Installation/Fitout department (Dale Sullivan).

> Remember that this workflow has a RELEASE STATUS of "Released". This allows non-Administrator users to execute the workflow.

27 Navigate to **Setup > Custom > Configure Login Access**. The Setup Login Access page will be displayed.

28 Select "Dale Sullivan" and "Inger Brown" in the EMPLOYEES list on the left side of the page.

29 Select the "Sales Person" role in the ROLES list on the right side of the page.

30 Click the **Generate Employee Access** button. The Setup Login Access page re-displays with links to the IDs containing login access.

ibrown@TSTDRVnnnnnn.com (password: *training1*)
dsullivan@TSTDRVnnnnnn.com (password: *training1*)

31 Make a note of the unique number for your training account that follows *TSTDRV*:

*TSTDRV*___________________

32 Logout as Larry Nelson (Administrator) and login as Inger Brown using the login information above. If prompted to enter security questions, click **Remind Me Later**.

33 Create a new Furniture Request record by navigating to **Customers > Other > Furniture Request > New**.

You should be able to create a new Furniture Request record.

34 Logout as Inger Brown and login as Dale Sullivan using the login information above. If prompted to enter security questions, click **Remind Me Later**.

35 Attempt to create a new Furniture Request record by navigating to **Customers > Other > Furniture Request > New**.

You should receive a server-generated error message indicating that you are not authorized to create Furniture Request records.

36 Logout as Dale Sullivan and log back in as Larry Nelson (Administrator).

EXERCISE SOLUTIONS

Having trouble with this module's workflows?

1. Navigate to **Customization > Workflow > Workflows**.
2. Change the RECORD TYPE filter to "Furniture Request SOLUTION".
3. Click **Edit** next to the workflow you would like to analyze:
   - SOL Furniture Request Load 06 (*Exercises 01-02*)

CONFIGURING ACTIONS ON RECORD SUBMIT

Module Exercises	
01	Create a Workflow to Support Actions on Record Save
02	Identify the Required Fields for the Create Record Action
03	Implement Actions and Conditions using Server Triggers
04	Modify the Estimated Project Cost Validation
Optional Exercises	
05	Customize the Send Email Action

01: Create a Workflow to Support Actions on Record Save

Scenario: Custom Furniture Requests are always specific to an individual customer. SuiteDreams would like to perform some additional processing on Furniture Request records at the time of creation.

SuiteDreams has provided the following business requirements, which you will configure in *Exercise 03*:

- Create a Phone Call record, assign it to the project manager, and associate it with the customer.
- Send an email notification from the project manager to the customer.
- Navigate the user to the selected customer's record.

Create a New Workflow

1 Navigate to **Customization > Workflow > Workflows > New**.

2 Fill out the **Basic Information** and **Event Definition** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
NAME	Furniture Request Save
SCRIPT ID	_sdr_furn_req_save
RECORD TYPE	Furniture Request
DESCRIPTION	Enter a description of the workflow.
EXECUTE AS ADMIN	Checked **Note**: This workflow will access the Employee record type in order to implement the Send Email action. This is in order to check for the existence of an employee's email address. Sales Person roles do not have access to the Employee record by default. By setting the workflow to EXECUTE AS ADMIN, it always runs with Administrator level privileges. You will usually need to check this box for workflows that access data in other records, especially when the workflows are accessed by Sales roles or the Employee Center role.
RELEASE STATUS	Released
KEEP INSTANCE AND HISTORY	Always
ENABLE LOGGING	Checked
Event Definition	
ON CREATE	Checked
TRIGGER TYPE	After Record Submit

3 Click **Save**.

Create the Necessary States

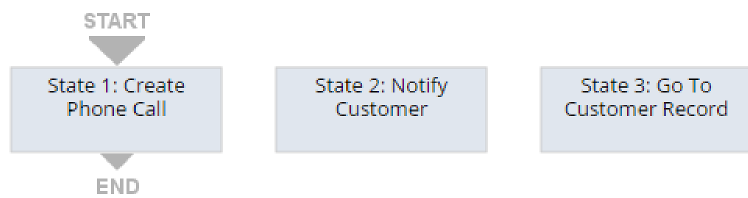
To meet the business requirements above, we need to create three new states, each representing one of the business rules to be implemented.

4 Click the **+ New State** button twice so you end up with three states and rename them as follows:

- State 1 → State 1: Create Phone Call
- State 2 → State 2: Notify Customer
- State 3 → State 3: Go To Customer Record

Note: **DO NOT** enable DO NOT EXIT WORKFLOW for any of the states.

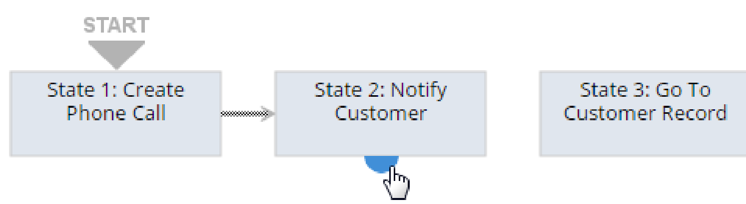
Your workflow should resemble the workflow below:



You can reorder states as needed by clicking and dragging them within the Workflow Manager. You can even draw a marquee around multiple states and drag them as a group!

Create Transitions

5 To create the necessary transition between State 1 and State 2, hover your mouse over State 1 and drag the transition handle (represented by the blue semi-circle) to State 2, as shown below:

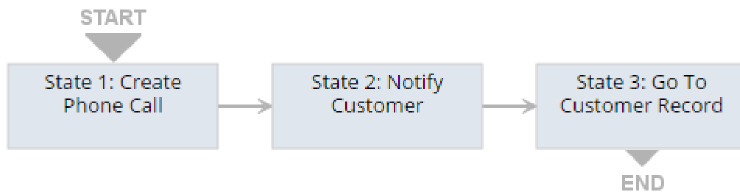


6 Create a transition between State 2 and State 3 in the same way.

One alternative method of creating transitions is to double-click on a state, select the **Transitions** sublist, and click the **New Transition** button.

You can edit an existing transition by clicking on it in the diagram or by editing it from the **Transitions** tab in the Context Panel.

Your workflow should now resemble the following:



Perform Initial Test Prior to Action Configuration

7 Navigate to **Lists > Custom > Furniture Request > New**. You might find it useful to open the new Furniture Request entry form in a new browser tab/window.

8 Create a new Furniture Request record and click **Save**. You will automatically be taken to the new Furniture Request record in View mode.

9 Select the **Workflow History** subtab.

You should clearly see that your new Furniture Request Entry, Load, and Save workflows were initiated on the record and the Save workflow transitioned sequentially from *State 1 > State 2 > State 3*.

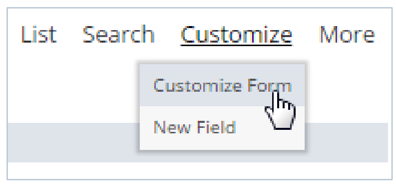
If not, check your configuration in the **Event Definition** section of the workflow definition.

02: Identify the Required Fields for the Create Record Action

Scenario: In order to successfully create the new Phone Call record in State 1, you must provide values in the required fields for that record type. Often times, the names of the fields displayed in dropdown lists are different than the names of fields displayed on forms in the user interface. You can easily determine which fields are which by completing the next series of steps.

1 Create a new Phone Call record by navigating to **Activities > Scheduling > Phone Calls > New**.

2 Select **Customize Form** under the **Customize** link in the upper-right corner of the page, as shown below:



3 Click the **Fields** subtab. The **Main** sublist will be displayed by default listing all the fields in the Phone Call record.

The label you see in the user interface is defined by the LABEL column. The name displayed in various dropdown lists in SuiteFlow is defined by the DESCRIPTION column (which is not editable).

Based on the fields listed, enter the description for each of the following field labels:

Label	Description
SUBJECT	
ORGANIZER	
DATE	
COMPANY (Related Records sublist)	

There is no reason to save the Phone Call record.

03: Implement Actions and Conditions using Server Triggers

Scenario: Custom Furniture Requests are always specific to an individual customer. SuiteDreams would like to perform some additional processing on Furniture Request records at the time of creation.

SuiteDreams has provided the business requirements below:

Business Rule	Action/Condition Configuration
A	Create a Phone Call record, assign it to the Project Manager, and associate it with the customer. The phone call should occur 3 days in the future. **Action**: Create Record **Server Trigger**: Entry
B	Send an email notification from the Project Manager to the customer, welcoming them to SuiteDreams, and notifying them regarding the pending phone call. **Action**: Send Email **Server Trigger**: Entry
C	Navigate the user to the specific Customer record for the inclusion of notations, which are usually entered once a custom Furniture Request has been created. **Action**: Go To Record **Server Trigger**: Entry

Implement Business Rule A

In this step of the exercise, you will create a Phone Call record, assign it to the Project Manager, and associate it with the chosen customer. The phone call should occur 3 days in the future and must take place before the Furniture Request can be approved. We will discuss approval-based workflows later in this course.

1. Edit the Furniture Request Save workflow.
2. Select State 1: Create Phone Call and add a **Create Record** action.

3 Fill out the **Basic Information** and **Parameters** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Entry
Parameters	
RECORD TYPE	Phone Call

4 Still within the **Parameters** section of the Create Record action, complete the following steps to define the Phone Call record fields based on the field information you noted in the previous exercise.

Your **Parameters** should resemble the following when finished:

FIELD *	TEXT	CHECKED	DATE	SELECTION	JOIN	VALUE FIELD	FORMULA
Title	Discuss project with customer.						
Assigned						Project Manager	
Start Date			three days from now				
Company						Customer	

5 Click on the icon to open the Visual Builder. Configure the following condition:

Condition	
FIELD	COMPARE TYPE
Custom	checked

6 Click **Add** and then **Save** on the Workflow Condition form.

7 Click **Save** on the Workflow Action configuration form. In the list of actions under the **State** tab, you should see the following:

Entry
Create Record Phone Call

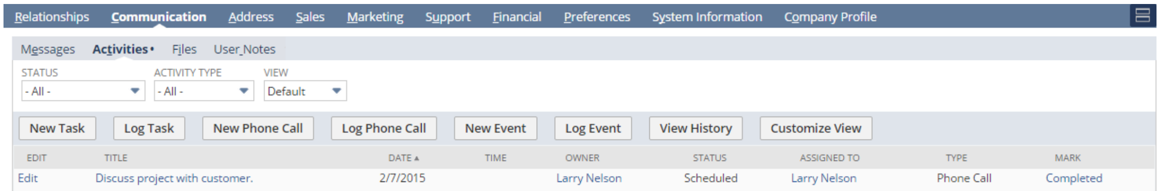
Test Business Rule A

8 Create and save a new custom Furniture Request record.

9 Open the record of the customer you selected in the Furniture Request by clicking on the customer hyperlink, or by navigating to **Lists > Relationships > Customers**.

10 Click the **Communication** subtab, then the **Activities** sublist.

You should see the Phone Call record listed as shown below. Keep in mind the date was set to three days from now.



If you do not see the Phone Call record, you likely did not select a customer when you created the Furniture Request record; and if that was the case, you did not have the CUSTOM checkbox selected.

You can view a full list of Phone Call records by navigating to **Activities > Scheduling > Phone Calls**.

Implement Business Rule B

Now you will configure a Send Email action that sends an automated email notification from the Project Manager to the selected customer, welcoming them to SuiteDreams, and notifying them of the pending phone call.

11 Edit the Furniture Request Save workflow once again.

12 Select State 2: Notify Customer and add a **Send Email** action.

13 Fill out the **Basic Information** and **Parameters** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Entry
Parameters	
Sender (FROM FIELD)	RECORD (JOIN FIELD): Current Record FIELD: Project Manager
Recipient (FROM FIELD)	RECORD (JOIN FIELD): Current Record FIELD: Customer
Content (CUSTOM)	SUBJECT: Welcome to SuiteDreams! BODY: We look forward to working with you to create a piece of furniture to your specifications. Regards, The SuiteDreams Custom Furniture Project Team

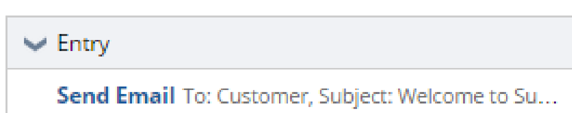
14 Click on the icon to open the Visual Builder. Configure the following three conditions:

Condition	
FIELD	COMPARE TYPE
Custom	checked

Condition		
RECORD	FIELD	COMPARE TYPE
Project Manager	E-mail	not empty
Customer	E-mail	not empty

15 Click **Save** on the Workflow Condition form.

16 Click **Save** on the Workflow Action configuration form. In the list of actions under the **State** tab, you should see the following:



> Make sure Send Email actions are conditionally executed based on the existence of an email address when the email fields are not mandatory. If you try to send an email to a blank email address, it will either give you an "Unexpected Error" message, or the email will fail (noted in the logs) but a copy of the email message will still get attached to the record.
>
> Much like the Email field itself, you may get an error if the Sender or Recipient records are not set on the Furniture Request record (e.g. Project Manager not selected). In this case Project Manager and Customer are required, so there is no concern.

Test Business Rule B

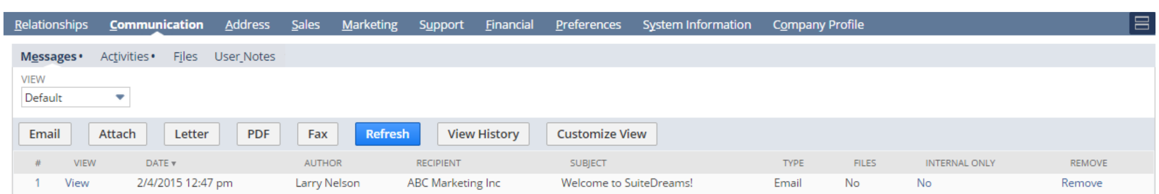
> Before testing, make sure the sender and recipient specified in your Send Email action have email addresses on their respective entity records!

17 Create and save a new custom Furniture Request record (**Lists > Custom > Furniture Request > New**).

18 Open the record of the customer you selected in the Furniture Request by clicking on the customer hyperlink on the new record, or by navigating to **Lists > Relationships > Customers**.

19 Click the **Communication** subtab, then the **Messages** sublist.

You should see the email notification similar to what is shown below:



Clicking on the **View** link opens the email notification.

Implement Business Rule C

Finally, let's configure a Go To Record action which navigates the user to the specific Customer record for the inclusion of notations, which are usually entered once a custom Furniture Request has been created.

20 Edit the Furniture Request Save workflow, select State 3: Go To Customer Record, and add a **Go To Record** action.

21 Fill out the **Basic Information** and **Parameters** sections (leaving all other fields at their defaults unless otherwise noted):

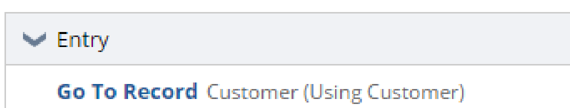
Basic Information	
TRIGGER ON	Entry **Note**: The default trigger is After Record Submit (unlike the Create Record and Send Email actions which default to Entry but are also supported at After Record Submit).
Parameters	
RECORD TYPE	Customer
FIELD	Customer **Note**: Represents a specific Customer ID selected when creating a new Furniture Request.
OPEN IN EDIT MODE	Unchecked **Note**: Checking this box opens the record in Edit mode. Unchecking this box opens the record in View mode.

22 Click on the icon to open the Visual Builder. Configure the following condition:

FIELD	COMPARE TYPE
Custom	checked

23 Click **Add** and **Save** on the Workflow Condition form.

24 Click **Save** on the Workflow Action configuration form. In the list of actions under the **State** tab, you should see the following:



Test Business Rule C

25 Test the Go To Record action by creating and saving a new custom Furniture Request record.

The user should be automatically taken to the Customer record selected in the Furniture Request.

Depending on the status of the OPEN IN EDIT MODE checkbox, the record will open in either Edit or View mode.

26 (**Optional**) Once you have successfully tested this business rule, you might consider inactivating the Go To Record action in State 3. This will prevent you from being navigated away from the newly entered Furniture Request record; making testing future exercises easier.

To do so, simply edit the Go To Record action and check the INACTIVE checkbox.

04: Modify the Estimated Project Cost Validation

> ⚠ This exercise requires *formulas* which are not covered in this course. However, you should be able to successfully complete the exercise using the steps below.

Scenario: SuiteDreams does not want the project cost validation to be exposed to users of NetSuite. The details of actions executed on client triggers, such as Before User Submit, can be seen by users simply by viewing the underlying HTML source code of a NetSuite page.

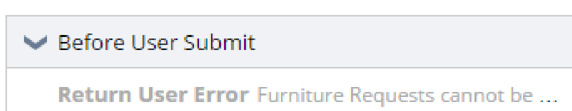
For this reason, SuiteDreams would like the ESTIMATED PROJECT COST validation moved to a trigger occurring after the form has been saved.

The additional business requirements to be implemented in this exercise are listed below:

Business Rule	Action/Condition Configuration
D	➢ Create a workflow field to store the $10,000 limit. This makes the workflow easier to maintain if the value is referred to in multiple places. ➢ Modify the condition in your action to evaluate against the workflow field instead of a hardcoded value. ➢ Include ESTIMATED PROJECT COST and the value of the workflow field as part of the error message. **Action**: Return User Error **Server Trigger**: Before Record Submit

Inactivate the Existing Estimated Project Cost Validation

1. Return to the list of workflows and edit the Furniture Request Entry workflow.
2. Select State 1: Furniture Request Entry and edit the Return User Error action configured on Before User Submit.
3. At the bottom of the **Basic Information** section, check the INACTIVE checkbox.
4. Click **Save**. You should see your inactivated Return User Error as the grayed out entry:



Inactivating actions allows you to preserve your earlier action configuration and easily re-activate later, if necessary.

Modify the Furniture Request Save Workflow

In this series of steps, you are going to modify the Furniture Request Save workflow so it initiates on Before Record Submit, rather than After Record Submit. This will ensure the new ESTIMATED PROJECT COST validation executes before the record is saved to the database.

5 Edit the Furniture Request Save workflow.

6 Click the edit link represented by the pencil icon ( ) located in the **Workflow** tab and make the following changes (leaving all other fields at their defaults unless otherwise noted):

Event Definition	
TRIGGER TYPE	Before Record Submit

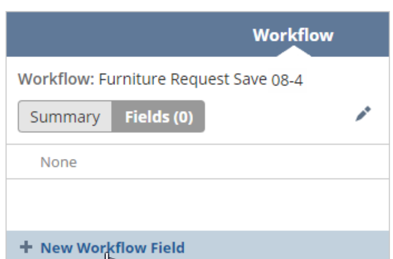
7 Click **Save**.

> If you do not change the TRIGGER TYPE to Before Record Submit, the validation will not execute because the Return User Error action is not supported on After Record Submit.

Implement Business Rule D

Create a New Workflow Field to Store Maximum Project Cost

8 Select **Fields** under the **Workflow** tab in the Context Panel (as shown below), and then click **+ New Workflow Field**.



9 Configure the new Workflow Field as follows:

LABEL	Maximum Project Cost
ID	_sdr_furn_req_max_proj_cost
DESCRIPTION	Enter a description of the workflow field.
TYPE	Currency
STORE VALUE	Checked
DEFAULT VALUE (**Validation & Defaulting** subtab)	10000

10 Click **Save** on the Workflow Field form. You should see the following:

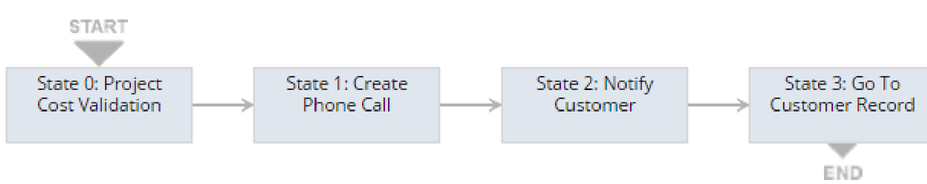
Create a New State to Perform the Validation

11 Create a new state by clicking on the **+ New State** button in the upper-left corner of the Workflow Manager.

12 Rename the state to "State 0: Project Cost Validation" and enable the START STATE checkbox.

13 Drag and drop the new state so it is positioned as the first (entry) state of the workflow.

14 Create a transition from State 0 to State 1. Your workflow should resemble the following diagram (your workflow may be stubbed out vertically, not horizontally as illustrated below):



Identify Field IDs for Estimated Project Cost and Maximum Project Cost

15 Create a new Furniture Request record to find out the field ID of the ESTIMATED PROJECT COST field. **Hint**: Click on the field's label.

ORACLE NETSUITE

To verify the field ID of the MAXIMUM PROJECT COST field, click on the "Maximum Project Cost" workflow field in the **Fields** list under the **Workflow** tab.

Enter the corresponding field IDs below:

Field	Field Name	Field ID
A	ESTIMATED PROJECT COST	
B	MAXIMUM PROJECT COST	

Configure Action to Perform the Validation

Now let's configure the same Return User Error action that was used for the original validation in the Furniture Request Entry workflow, only this time, using a server trigger.

16 Re-edit the Furniture Request Save workflow and add a **Return User Error** action to State 0: Project Cost Validation.

17 Fill out the **Basic Information** and **Parameters** sections (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Before Record Submit
Parameters	
TEXT	Configure the message text as indicated below by substituting the field IDs above. Make sure to enclose each field name in curly braces. *You entered an Estimated Project Cost of {field_A}. The maximum cost allowed is {field_B}.*

Fields are entered in the message text by enclosing their IDs in curly braces {}. This is not supported on client triggers, but it is possible here since you are validating on the Before Record Submit server trigger.

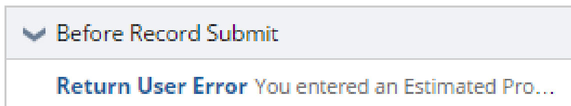
You can optionally embed HTML tags in the message text as well!

18 Click on the icon to open the Visual Builder. Configure the following condition:

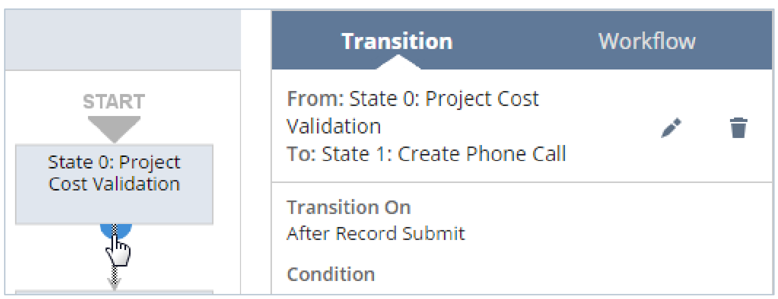
Condition		
FIELD	COMPARE TYPE	VALUE FIELD
Estimated Project Cost	greater than	Maximum Project Cost (Workflow)

19 Click **Add** and then **Save** on the Workflow Condition form.

20 Click **Save** on the Workflow Action configuration form. In the list of actions under the **State** tab, you should see the following:



21 Finally, select the transition between State 0 and State 1 and click the edit link represented by the pencil icon ( ) located in the **Transition** tab:



22 Change the TRANSITION ON configuration to "After Record Submit".

Note: This will ensure that the *trigger context* is changed to support the Send Email and Go To Record actions in States 2 and 3, respectively.

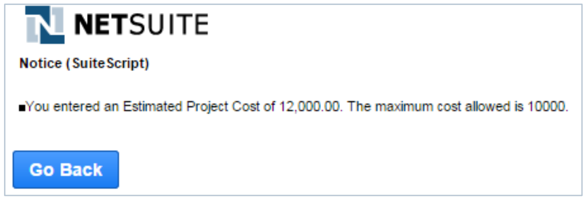
23 Click **Save** on the Workflow Transition form.

Test Business Rule D

Remember to clear cache using **CTRL+F5** (Windows) or **CMD+R** (Mac OS) since you inactivated the previous version of the Estimated Project Cost validation running in the browser.

24 Test the new validation by creating and saving a new custom Furniture Request record, making sure that the value entered for ESTIMATED PROJECT COST is greater than $10,000.

Upon save, an error should be displayed similar to the following:



Rather than receiving a user-friendly popup message, you are taken to a SuiteScript error page where the details of the Return User Error action are displayed.

This error page is a tell-tale sign that the Return User Error action has been configured on a server trigger rather than a client trigger.

05: Customize the Send Email Action (Optional)

This exercise requires the use of *formulas* which are not covered in this course. However, you should be able to complete the exercise using the detailed steps below.

Scenario: SuiteDreams wants to personalize their notifications by having the subject line and body of the email contain values from the Furniture Request, related Customer record, and Employee record.

SUBJECT	Welcome to SuiteDreams, *company_name!*
BODY	We look forward to working with you to create a *name_of_furniture* per your specifications. We have your phone number as *phone_number* and your current balance is *balance*. Please contact us if there are any discrepancies. Regards, *project_manager* *email_address* *phone_number*

The information in *italics* above will be dynamically replaced with data on the Furniture Request and related records.

Identify Field IDs

1 You can identify field IDs by opening a Furniture Request, Customer, and Employee record, locating the appropriate field, and clicking on each field's label.

If field IDs are not displaying when clicking on a field's label, you must enable it by navigating to **Home > Set Preferences** and check SHOW INTERNAL IDS in the **Defaults** section of the **General** subtab.

2 Locate and note the field IDs based on the records and field labels below:

Record	Field Label	Field ID
Furniture Request	CUSTOMER	
	FURNITURE PROJECT NAME	
	PROJECT MANAGER	
Customer	PHONE	
	BALANCE (**Financial** subtab > **Balance Information**)	
	EMAIL	
Employee (Project Manager)	EMAIL	

Modify the Text of the Email

3 Edit the Furniture Request Save workflow and edit the existing Send Email action in State 2: Notify Customer.

4 Edit the SUBJECT and BODY fields based on the text in the exercise scenario above; substituting the items in *italics* below with the proper field IDs.

> Welcome to SuiteDreams, *{company_name}*!
>
> We look forward to working with you to create a *{name_of_furniture}* per your specifications.
>
> We have your phone number as *{phone_number}* and your current balance is *{balance}*. Please contact us if there are any discrepancies.
>
> Regards,
> *{project_manager}*
> *{email_address}*
> *{phone_number}*

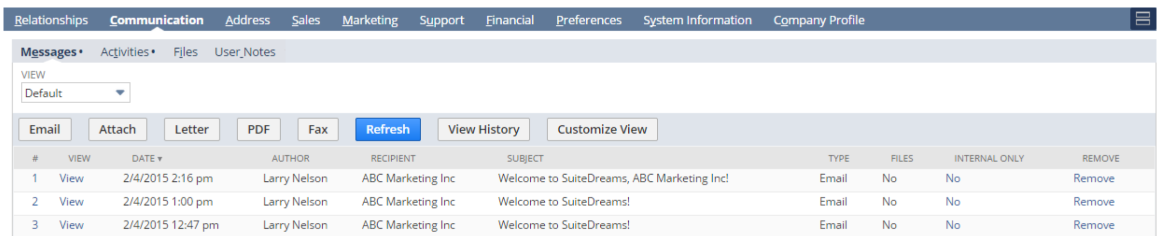
5 Click **Save**.

> Field IDs are always entered in all lowercase characters and must be surrounded by curly braces. For example, *{name}*
>
> Field IDs on related records have an additional identifier to represent the related record. When processing custom record types, these identifiers are the field IDs of the List/Record fields creating the relationship. For example, *{custrecord_sdr_furn_req_pm.phone}*.

> Use caution when copying field IDs from the field-level help directly to the body of the email message. This will copy the ID inclusive of any formatting which might prevent the system from fetching the actual information. To avoid this, copy/paste the field ID in a plain text editor (such as Notepad) first so that the formatting is removed before pasting it to the body of the email.

6 Test the updated email notification by creating and saving a new Furniture Request record.

7 From the customer record currently being displayed in View mode, select the **Communication** subtab and look for the new email notification entry in the **Messages** sublist.



8 Click on the **View** link on the email you want to open to view the whole message.

EXERCISE SOLUTIONS

Having trouble with this module's workflow?

1 Navigate to **Customization > Workflow > Workflows**.

2 Change the RECORD TYPE filter to "Furniture Request SOLUTION".

3 Click **Edit** next to the workflow you would like to analyze:

- SOL Furniture Request Save 07 (*Exercises 01-03*)
- SOL Furniture Request Save 07-4 (*Exercise 04*)
- SOL Furniture Request Save 07-5 (*Optional Exercise 05*)

02: Identify the Required Fields for the Create Record Action

3 Based on the fields listed, enter the description for each of the following field labels:

Label	Description
SUBJECT	Title
ORGANIZER	Assigned
DATE	Start Date
COMPANY (Related Records sublist)	Company

04: Modify the Estimated Project Cost Validation

18 Enter the corresponding field IDs below:

Field	Field Name	Field ID
A	ESTIMATED PROJECT COST	custrecord_sdr_furn_req_est_proj_cost
B	MAXIMUM PROJECT COST	custworkflow_sdr_furn_req_max_proj_cost

05: Customize the Send Email Action

1 Locate and note the field IDs based on the records and field labels below:

Record	Field Label	Field ID
Furniture Request	CUSTOMER	`custrecord_sdr_furn_req_customer`
	FURNITURE PROJECT NAME	`name`
	PROJECT MANAGER	`custrecord_sdr_furn_req_pm`
Customer	PHONE	`custrecord_sdr_furn_req_customer.phone`
	BALANCE (**Financial** subtab > **Balance Information**)	`custrecord_sdr_furn_req_customer.balance`
Employee (Project Manager)	EMAIL	`custrecord_sdr_furn_req_pm.email`
	PHONE	`custrecord_sdr_furn_req_pm.phone`

TROUBLESHOOTING WORKFLOW EXECUTION

Module Exercises	
01	Examine Active Workflows and Workflow History
02	Troubleshoot Workflow Execution using the Workflow Execution Log

01: Examine Active Workflows and Workflow History

Scenario: In this exercise, you will practice viewing **Active Workflows** and **Workflow History** using the Furniture Request Save workflow.

Viewing Active Workflows

1. Create and save a new Furniture Request record making sure to select a customer from the CUSTOMER dropdown list.
2. Return to the list of Furniture Request records and View/Edit the record you just created.
3. Select the **Active Workflows** subtab. You should see Furniture Request Entry and Furniture Request Load listed as currently running workflows on the record.

   Why is Furniture Request Save not listed?

   Answer:

Viewing Workflow History

4. Select the **Workflow History** subtab. Here you can see a more granular view of the individual states visited during the processing of each workflow.

   You should see entries for the Furniture Request Entry and Furniture Request Load single-state workflows, and the 4 states in the Furniture Request Save workflow. You can see that the Save workflow transitioned sequentially through each state.
5. Click on each **Log** link next to each state and view its corresponding *Workflow Execution Log*.

You should see that all actions were successfully processed in each state.

State 0: Project Cost Validation

If you entered an ESTIMATED PROJECT COST < $10,000 the workflow execution log for State 0 will show that the Return User Error action was considered but was not ultimately executed. More on this in a moment.

State 1: Create Phone Call

Entry	Result	Date/Time
Running ONENTRY trigger under BEFORESUBMIT (Event: CREATE; Context: USERINTERFACE)		3/18/2015 - 11:36:54.266
CREATERECORD: Phone Call (PHONECALL)	Executed	3/18/2015 - 11:36:54.319
Condition: Custom = T	True	3/18/2015 - 11:36:54.349
Record type: Phone Call (PHONECALL)		3/18/2015 - 11:36:54.363
title = Discuss project with customer.		3/18/2015 - 11:36:54.666
assigned = -5		3/18/2015 - 11:36:54.689
startdate = 3/21/2015		3/18/2015 - 11:36:54.699
company = 119		3/18/2015 - 11:36:54.742
New record ID: 398		3/18/2015 - 11:36:54.855
Transition to State 2: Notify Customer	Executed	3/18/2015 - 11:36:54.874
Running ONEXIT trigger under BEFORESUBMIT (Event: CREATE; Context: USERINTERFACE)		3/18/2015 - 11:36:54.884

State 2: Notify Customer

Entry	Result	Date/Time
Running ONENTRY trigger under AFTERSUBMIT (Event: CREATE; Context: USERINTERFACE)		3/18/2015 - 12:00:57.290
SENDEMAIL	Executed	3/18/2015 - 12:00:57.311
Condition: Custom = T And Project Manager : E-mail Is Not Empty And Customer : E-mail Is Not Empty	True	3/18/2015 - 12:00:57.538
From = -5 To = 325		3/18/2015 - 12:00:57.568
Transition to State 3: Go To Customer Record	Executed	3/18/2015 - 12:00:58.009
Running ONEXIT trigger under AFTERSUBMIT (Event: CREATE; Context: USERINTERFACE)		3/18/2015 - 12:00:58.078

State 3: Go To Customer Record

Entry	Result	Date/Time
Running ONENTRY trigger under BEFORESUBMIT (Event: CREATE; Context: USERINTERFACE)		3/18/2015 - 11:36:55.003
Running AFTERSUBMIT trigger (Event: CREATE; Context: USERINTERFACE)		3/18/2015 - 11:36:55.298
GOTORECORD: CUSTOMER	Executed	3/18/2015 - 11:36:55.311
Condition: Custom = T	True	3/18/2015 - 11:36:55.331
Customer (Using Customer)		3/18/2015 - 11:36:55.345
Workflow finished		3/18/2015 - 11:36:55.380

02: Troubleshoot Workflow Execution using the Workflow Execution Log

Scenario: In this exercise, you will examine a pre-built (and broken) workflow called "Furniture Request Troubleshooting" to see how the configuration of triggers, actions, and transitions affect overall workflow execution. After running the broken workflow, you will view the Workflow Execution Log to understand the logging behind the workflow's processing.

> The Furniture Request Troubleshooting workflow is a modified of the Furniture Request Save workflow. To avoid conflicts during workflow execution, you must first disable Furniture Request Save and enable Furniture Request Troubleshooting.

Disable the Furniture Request Save Workflow

1 Return to the list of workflows and edit the Furniture Request Save workflow.

2 Click the edit link represented by the pencil icon ( ) located in the **Workflow** tab.

3 Change the RELEASE STATUS from "Released" to "Not Initiating".

4 Click **Save**.

Enable the Furniture Request Troubleshooting Workflow

This workflow is currently in a "Not Initiating" status. Let's change the status to "Released" so we can effectively troubleshoot its execution.

5 Once again, return to the list of workflows and edit the pre-built "Furniture Request Troubleshooting" workflow.

6 Click the edit link represented by the pencil icon ( ) located in the **Workflow** tab.

7 Change the RELEASE STATUS from "Not Initiating" to "Released".

8 Select "Always" from the KEEP INSTANCE AND HISTORY dropdown list.

9 Make sure ENABLE LOGGING is checked.

10 Click **Save**.

Execute the Workflow and Test

11 Create a new Furniture Request record.

The "broken" Furniture Request Troubleshooting workflow has just been initiated on the new record.

If you had not previously inactivated the Go To Record action in State 3, the first thing you may have noticed is that the Customer record associated with the Furniture Request is not displayed. Instead, you are taken to the newly created Furniture Request record in View mode. This should be a clue that something has changed during workflow execution.

Let's consult the Workflow Execution Log to see what happened.

Troubleshoot Workflow Execution

12 Click on the **Workflow History** subtab.

You should now see that the new Furniture Request Troubleshooting workflow is now executing instead of the Furniture Request Save workflow.

13 Click the **Log** link in the "State 0: Project Cost Validation" row. This represents the entry state of the Troubleshooting workflow.

14 On the Workflow Log page for the selected state, enable the **Show Rejected Actions/Transitions** checkbox.

The following log details will be displayed:

Entry	Result	Date/Time
Workflow initiated		3/6/2015 - 11:29:23.848
Running ONENTRY trigger under BEFORESUBMIT (Event: CREATE; Context: USERINTERFACE)		3/6/2015 - 11:29:23.871
RETURNUSERERROR: You entered an Estimated Project Cost of {custrecord_sdr_furn_est_proj_cost}. Th...	Considered	3/6/2015 - 11:29:23.888
Condition: Estimated Project Cost > Workflow : Maximum Project Cost	False	3/6/2015 - 11:29:23.908
Transition to State 1: Create Phone Call	Executed	3/6/2015 - 11:29:23.919
Running ONEXIT trigger under BEFORESUBMIT (Event: CREATE; Context: USERINTERFACE)		3/6/2015 - 11:29:23.930

Notice that the Return User Error action was considered for execution. However, you will see that the condition, *"If Estimated Project Cost is greater than Maximum Project Cost"* (which is 10,000), evaluated *false*. The action was considered for execution but was ultimately not executed (since you likely did not enter an ESTIMATED PROJECT COST greater than 10,000).

15 Click the **Log** link in the "State 1: Create Phone Call" row.

Entry	Result	Date/Time
Running ONENTRY trigger under AFTERSUBMIT (Event: CREATE; Context: USERINTERFACE)		3/18/2015 - 13:13:04.681
CREATERECORD: Phone Call (PHONECALL)	Executed	3/18/2015 - 13:13:04.714
Condition: Custom = T	True	3/18/2015 - 13:13:04.732
Record type: Phone Call (PHONECALL)		3/18/2015 - 13:13:04.743
title = Discuss project with customer.		3/18/2015 - 13:13:04.885
assigned = -5		3/18/2015 - 13:13:04.910
startdate = 3/21/2015		3/18/2015 - 13:13:04.921
company = 325		3/18/2015 - 13:13:04.939
New record ID: 402		3/18/2015 - 13:13:05.013
Transition to State 2: Notify Customer	Executed	3/18/2015 - 13:13:05.038
Running ONEXIT trigger under AFTERSUBMIT (Event: CREATE; Context: USERINTERFACE)		3/18/2015 - 13:13:05.049

Analyze the triggers and actions executed. Do you see any problems at this state?

Answer:

16 Click the **Log** link in the "State 2: Notify Customer" row.

Entry	Result	Date/Time
Running ONENTRY trigger under BEFORESUBMIT (Event: CREATE; Context: USERINTERFACE)		3/6/2015 - 11:29:24.549
SENDEMAIL action skipped. Not executable with trigger BEFORESUBMIT	Skipped	3/6/2015 - 11:29:24.564
Transition to State 3: Go To Customer Record	Executed	3/6/2015 - 11:29:24.578
Running ONEXIT trigger under BEFORESUBMIT (Event: CREATE; Context: USERINTERFACE)		3/6/2015 - 11:29:24.588

Here you will see that the Send Email action was skipped. This is because State 2 inherited the Before Record Submit trigger from State 0. However, if you recall from our discussion about the Send Email action, it can only be executed on After Record Submit.

17 Finally, click the **Log** link in the "State 3: Go To Customer Record" row.

Entry	Result	Date/Time
Running ONENTRY trigger under BEFORESUBMIT (Event: CREATE; Context: USERINTERFACE)		3/6/2015 - 11:29:24.625
GOTORECORD action skipped. Not executable with trigger BEFORESUBMIT	Skipped	3/6/2015 - 11:29:24.641
Running AFTERSUBMIT trigger (Event: CREATE; Context: USERINTERFACE)		3/6/2015 - 11:29:24.829
Workflow finished		3/6/2015 - 11:29:24.847

If the Go To Record action in State 3 is still executing, you will notice a similar situation as with the Send Email action in State 2. Once again, the Before Record Submit trigger was inherited from State 2, and the Go To Record action is not supported on Before Record Submit, so it too, was skipped.

Fix the Workflow

Now let's fix the workflow so all actions are executed when they should be.

18 Go back and edit the Furniture Request Troubleshooting workflow and select the transition from State 0 to State 1 and click the edit link represented by the pencil icon ( ) in the **Transition** tab.

19 Select "After Record Submit" from the TRANSITION ON dropdown list.

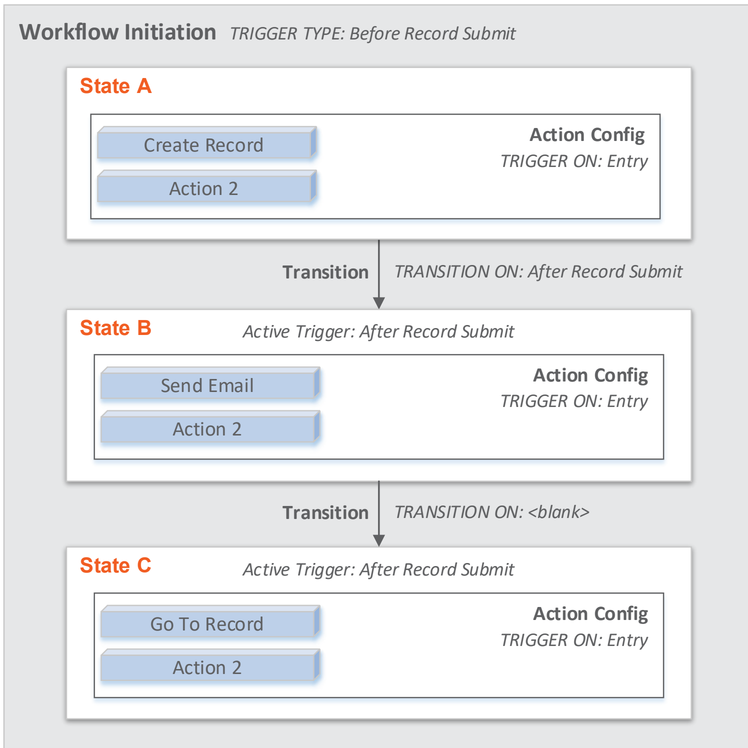
20 Click **Save**.

With the previous TRANSITION ON configuration of *<blank>*, State 1 inherited the Before Record Submit trigger from workflow initiation. This has no effect on the Create Record action configured in State 1, since Create Record actions are supported by all server triggers.

By changing the TRANSITION ON configuration from *<blank>* to After Record Submit, the workflow's active trigger is changed from Before Record Submit (inherited by the initiating workflow) to After Record Submit upon its entry to State 1.

The Before Record Submit trigger was carried into the remaining states and transitions which ultimately affected the Send Email and Go To Record actions in State 2 and State 3, respectively.

The following diagram helps illustrate the behaviors described above:



Re-test the Workflow

Verify that the workflow and all configured actions are executing properly.

21 Create one final Furniture Request record and verify the Workflow Execution Logs. They should all look nice and clean, with all actions executing normally again.

22 (**Optional**) Play around with different actions and transition configurations and see how they behave using the Troubleshooting workflow.

Reset Workflow Release Statuses

Before continuing, be sure to update the RELEASE STATUS of the Furniture Request Save workflow from "Not Initiating" back to "Released" and set the status of Furniture Request Troubleshooting to "Not Initiating".

Create a new Furniture Request record and verify that the Furniture Request Save workflow is processing correctly.

EXERCISE SOLUTIONS

Having trouble with this module's workflows?

1. Navigate to **Customization > Workflow > Workflows**.
2. Change the RECORD TYPE filter to "Furniture Request SOLUTION".
3. Click **Edit** next to the workflow you would like to analyze:
   - SOL Furniture Request Troubleshooting 08 (*Exercise 02*)

01: Examine Active Workflows and Workflow History

3. Why is Furniture Request Save not listed?

   Answer: Because it has already completed processing. Remember, **Active Workflows** only shows workflows currently running on a record. Entry and Load are both single-state workflows with the DO NOT EXIT WORKFLOW option enabled, so the workflow continues to run on the record.

   Furniture Request Save is a multi-state workflow that does not have DO NOT EXIT WORKFLOW enabled on any of its end states. As a result, the workflow transitions through each state until completion.

02: Troubleshoot Workflow Execution using the Workflow Execution Log

Entry	Result	Date/Time
Running ONENTRY trigger under AFTERSUBMIT (Event: CREATE; Context: USERINTERFACE)		3/18/2015 - 13:13:04.681
CREATERECORD: Phone Call (PHONECALL)	Executed	3/18/2015 - 13:13:04.714
Condition: Custom = T	True	3/18/2015 - 13:13:04.732
Record type: Phone Call (PHONECALL)		3/18/2015 - 13:13:04.743
title = Discuss project with customer.		3/18/2015 - 13:13:04.885
assigned = -5		3/18/2015 - 13:13:04.910
startdate = 3/21/2015		3/18/2015 - 13:13:04.921
company = 325		3/18/2015 - 13:13:04.939
New record ID: 402		3/18/2015 - 13:13:05.013
Transition to State 2: Notify Customer	Executed	3/18/2015 - 13:13:05.038
Running ONEXIT trigger under AFTERSUBMIT (Event: CREATE; Context: USERINTERFACE)		3/18/2015 - 13:13:05.049

15. Analyze the triggers and actions executed. Do you see any problems at this state?

    Answer: There are no processing errors for State 1: Create Phone Call.

Building Approval Workflows

Module Exercises	
01	Prepare Users with Login Access
02	Create a Basic Approval Processing Workflow
03	Configure Actions to Support Approval Processing
04	Configure Multiple Approvers
05	Add Sales Manager Approval
Optional Exercises	
06	Allow Submitters to Recall Furniture Requests
07	Allow Submitters to Resubmit after Rejection

The exercises in this module will walk you through the creation of an approval processing workflow that will serve as a foundation for the more complex approval processing workflows you will create later in this module.

01: Prepare Users with Login Access

Scenario: Several SuiteDreams users with various types of user roles are involved in a typical approval process. This exercise prepares your SuiteDreams account with additional user login access and assigns them to specific roles. This allows you to mirror real-world approval processing.

Configure Login Access

DO NOT reference the logins used by the instructor during the walkthrough demonstrations as they are specific to the instructor's account. You must complete the steps below to generate your own account logins.

The *TSTDRVnnnnnn* portion of the email address is always unique to the specific training account you are using to avoid conflicts with other student logins.

1. Navigate to **Setup > Custom > Configure Login Access**. The Setup Login Access page will be displayed.
2. Select the following employees in the EMPLOYEES list on the left side of the page by pressing the **CTRL** key and apply the "Sales Person" role in the ROLES list on the right side of the page:

- Aubrey Pober
- Edwin Goldwasser

3 Click the **Generate Employee Access** button. The Setup Login Access page re-displays with links to the IDs containing login access.

Make a note of the unique number for your training account that follows *TSTDRV*:

*TSTDRV*___________________

Copy them to Notepad for use later.

4 Select the following employees in the EMPLOYEES list on the left side of the page by pressing the **CTRL** key and apply the "Sales Manager" role in the ROLES list on the right side of the page:

- Fiona Small
- James Rollings
- John Rittman
- Matt Nolan

5 Click the **Generate Employee Access** button. The Setup Login Access page re-displays with links to the IDs containing login access.

6 Copy them to Notepad for use later. You should have these user account logins.

apober@TSTDRVnnnnnn.com
egoldwasser@TSTDRVnnnnnn.com
fsmall@TSTDRVnnnnnn.com
jrollings@TSTDRVnnnnnn.com
jrittman@TSTDRVnnnnnn.com
mnolan@TSTDRVnnnnnn.com

Note: The password for all users is *training1*.

The rest of the email address is always based on first name initial + last name, as follows:

Aubrey Pober	*apober@TSTDRVnnnnnn.com*
Edwin Goldwasser	*egoldwasser@TSTDRVnnnnnn.com*
Fiona Small	*fsmall@TSTDRVnnnnnn.com*
James Rollings	*jrollings@TSTDRVnnnnnn.com*
John Rittman	*jrittman@TSTDRVnnnnnn.com*
Matt Nolan	*mnolan@TSTDRVnnnnnn.com*

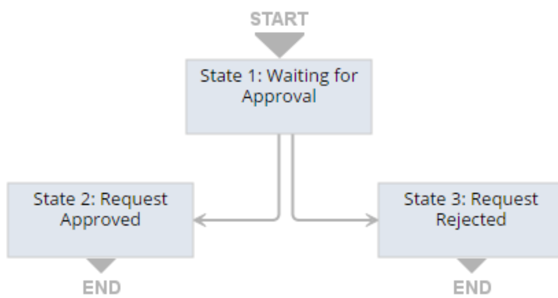
You will use these login IDs later when testing specific business rules associated with the approval processing workflow. Matt Nolan's login will only be used to test *Exercise 04*.You can re-run the Setup Login Access tool at any time to add another user with login access or to modify the list of roles.

When logging in as each of these users, you will be prompted to "Set Password Security Questions". You can click **Remind Me Later** when prompted.

02: Create a Basic Approval Processing Workflow

Scenario: In this exercise, you will create the basic structure of your approval processing workflow. It will serve as the foundation for the implementation of the remaining business rules in future exercises. You will stub out the states of the workflow, configure the required buttons to support approval and rejection processing, and configure button-triggered transitions.

Upon completion of this exercise, your workflow will resemble following:



Create a New Workflow

1 Create a new workflow as follows (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
NAME	Furniture Request Approval
SCRIPT ID	_sdr_furn_req_approval
RECORD TYPE	Furniture Request
DESCRIPTION	Enter a description of the workflow.
EXECUTE AS ADMIN	Checked
RELEASE STATUS	Released
KEEP INSTANCE AND HISTORY	Always
ENABLE LOGGING	Checked
Event Definition	
ON CREATE	Checked
TRIGGER TYPE	After Record Submit

2 Click **Save**.

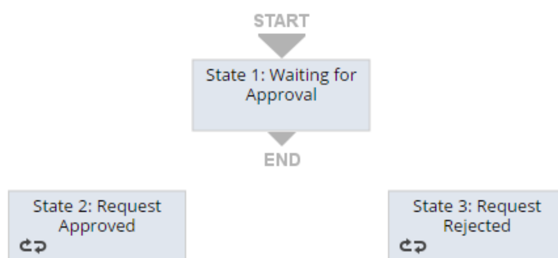
Create the Required States

3 Using the Workflow Manager, add and arrange the following three states:

- State 1: Waiting for Approval
- State 2: Request Approved
- State 3: Request Rejected

Note: Make sure you enable DO NOT EXIT WORKFLOW for each end state, but do not configure any actions at this time. You will do that later.

When completed, your workflow should resemble the diagram below:



Configure Buttons

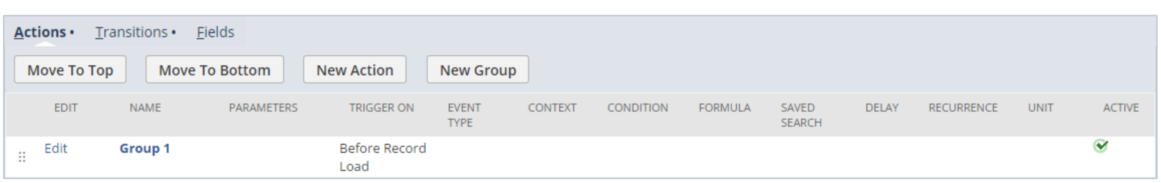
Buttons will be added so the Project Manager can approve or reject Furniture Request records. You will practice using *Action Groups* to configure both Add Button actions.

4 Edit State 1: Waiting for Approval.

5 Select the **Actions** subtab, click the **New Group** button, and enter the following information:

Basic Information	
SCRIPT ID	_sdr_furn_req_btn_grp
TRIGGER ON	Before Record Load

6 Click **Save**. You should see the following Action Group listed in the **Actions** subtab:



7 Click the **Edit** link next to your new action group.

8 At the bottom of the Workflow Action Group form, click the **New Action** button.

9 Select the **Add Button** action and configure it as follows (leaving all other fields at their defaults unless otherwise noted):

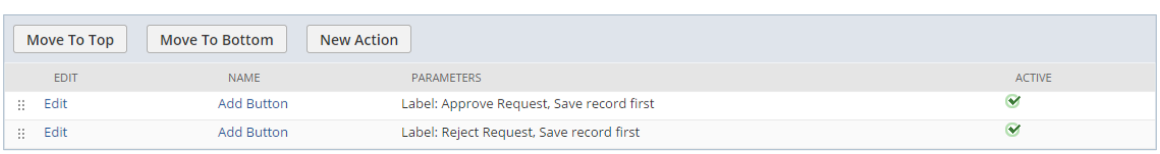
Basic Information	
SCRIPT ID	_sdr_furn_req_btn_approve
Parameters	
LABEL	Approve Request
SAVE RECORD FIRST	Unchecked (or Checked) This option ensures that any changes to a form are saved when clicking a button in Edit mode. This allows the approver to update the record as needed before approving or rejecting, but ensures those changes are committed to the database during approval. The buttons are seen as an additional option on the **Save** button, appearing as *Save & <button label>*.

10 Click **Save**. You should see your new "Approve Request" button:

11 Repeat the steps above for the "Reject Request" button. Enter the following information:

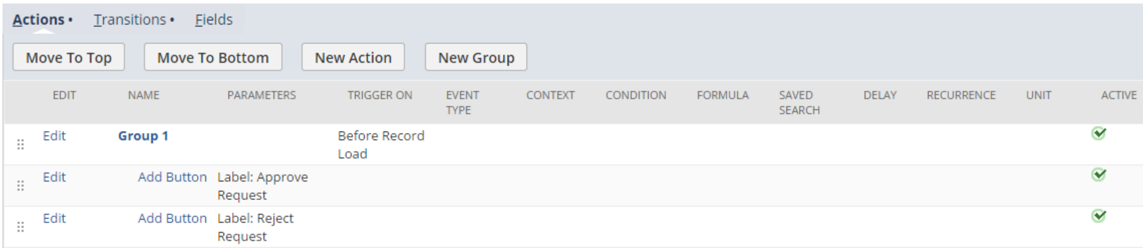
Basic Information	
SCRIPT ID	_sdr_furn_req_btn_reject
Parameters	
LABEL	Reject Request
SAVE RECORD FIRST	Unchecked (or Checked)

You should now see both buttons in your Action Group:



12 Click **Save** on the Workflow Action Group form. You may receive a message indicating the record has not changed.

13 Click **OK**. The State 1: Waiting for Approval action list should resemble the following:


Actions • Transitions • Fields

Move To Top | Move To Bottom | New Action | New Group

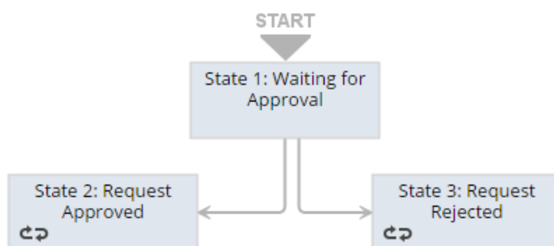
EDIT	NAME	PARAMETERS	TRIGGER ON	EVENT TYPE	CONTEXT	CONDITION	FORMULA	SAVED SEARCH	DELAY	RECURRENCE	UNIT	ACTIVE
Edit	**Group 1**		Before Record Load									✓
Edit	Add Button	Label: Approve Request										✓
Edit	Add Button	Label: Reject Request										✓

14 Click **Cancel** on the Workflow State form to return to the Workflow Manager.

Configure Transitions

Now let's add transitions between the three states and configure each button to trigger the appropriate transition.

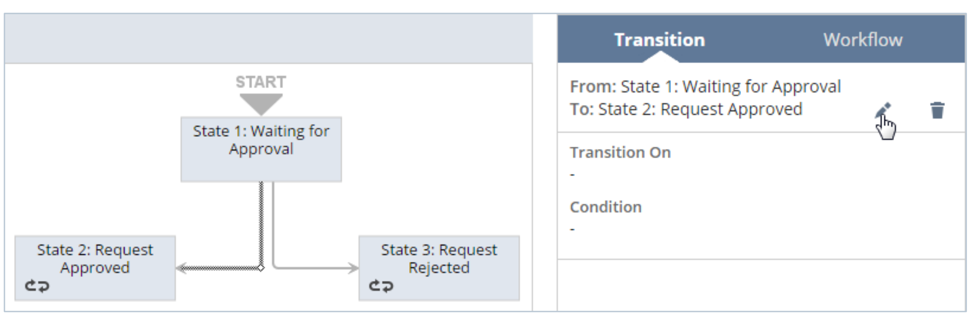
15 Create a transition from State 1 to State 2, and from State 1 to State 3. Your workflow should now resemble the following diagram:



What is the only sensible configuration for TRANSITION ON in this case?

Answer:

16 Select the transition between State 1 and State 2 and click the edit icon ( ) located in the **Transition** tab:



ORACLE NETSUITE

17 On the Workflow Transition form, enter the following transition information:

Basic Information	
TRANSITION ON	<*blank*> (no selection necessary)
Condition	
EXECUTE ON BUTTON	Approve Request

18 Click **Save**.

19 Repeat the steps above for the transition between State 1 and State 3:

Basic Information	
TRANSITION ON	<*blank*>
Condition	
EXECUTE ON BUTTON	Reject Request

Execute the Workflow and Test

20 Create and save two new Furniture Request records (one for approval and one for rejection), navigate to the list of Furniture Request records and open each record.

You should see your **Approve Request** and **Reject Request** buttons.

21 Select the **Active Workflows** subtab. You should see that your new approval workflow is currently waiting in State 1: Waiting for Approval.

22 Click the **Approve Request** button:

- Upon approval, the Furniture Request record will be re-displayed in View mode. Notice that the **Edit** button is available. You will configure a Lock Record action to prevent the edit of records after approval in a future exercise.
- Also notice that the Furniture Request Approval workflow is listed under **Active Workflows** and State 2: Request Approved is the current state.
- Select the **Workflow History** subtab. Notice that the approval workflow is currently in its approved state, State 2: Request Approved.

23 Repeat the steps above to test the rejection process.

03: Configure Actions to Support Approval Processing

Scenario: In this exercise, you will implement business rules to restrict the approval of Furniture Requests to the Project Manager, track the approval status of these records, and send email notifications to both the Project Manager and customer notifying them of the status of the project.

SuiteDreams has provided the following business requirements for approval processing:

Business Rule	Action/Condition Configuration
A	The Project Manager should be the only user who can either approve or reject a custom Furniture Request. ➢ The Project Manager may modify all fields on the Furniture Request record while deciding whether to approve or reject. ➢ No other user may modify the Furniture Request record while it is waiting to be approved/rejected by the Project Manager. **Action**: Set Field Value, Lock Record **Server Trigger**: Entry, Before Record Load
B	Create an "Approval Status" field to track the status of all custom Furniture Requests: ➢ Set to "Pending-Approval" while Project Manager is waiting to approve/reject. ➢ Set to "Approved" when the record has been approved. ➢ Set to "Rejected" when the record has been rejected. **Action**: Set Field Value **Server Trigger**: Entry
C	Configure Email Notifications: ➢ The Project Manager should be notified via email that a custom Furniture Request is awaiting their approval. ➢ Notify the customer (if exists) once their custom Furniture Request has been approved or rejected. **Action**: Send Email **Server Trigger**: Entry
D	Prevent all users from modifying the Furniture Request once it has reached the "Request Approved" or "Request Rejected" end states. **Action**: Lock Record **Server Trigger**: Before Record Load

Implement Business Rule A

Create a Workflow Field to Store the Current Approver

In this series of steps, you configure the workflow so the initial approver is always the Project Manager. Creating a generic field to hold the value of the current approver allows the workflow to scale better as more complexity is added to the approval process.

1 Edit the Furniture Request Approval workflow.

2 On the **Workflow** tab, select **Fields**, and then click **New Field** to add a workflow field as follows:

LABEL	Current Approver
ID	_sdr_furn_req_current_appr **Note**: Workflow fields must have unique IDs across workflows. If you have many workflows using the concept of current approver, you will need to differentiate the ID value across workflows
TYPE	List/Record
LIST/RECORD	Employee
STORE VALUE	Checked

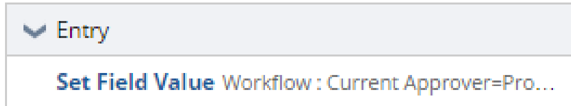
3 Click **Save**.

Configure an Action to Set Current Approver = Project Manager

4 Select State 1: Waiting for Approval and configure a **Set Field Value** action as follows (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Entry
Parameters	
FIELD	Current Approver (Workflow)
Value (FROM FIELD)	RECORD (JOIN FIELD): Current Record FIELD: Project Manager

5 Click **Save** on the Workflow Action configuration form. In the list of actions under the **State** tab, you should see the following:



You will use "Current Approver (Workflow)" in place of Project Manager when restricting actions to the approver.

Conditionally Execute the Add Button Actions based on the value of Current Approver

In this series of steps, you will identify the conditions necessary to determine when the Add Button actions should be executed. Ask yourself this question, "Who is the only user who should be able to click the Approve Request or Reject Request buttons?"

Per SuiteDreams' business requirements, the only user who should be able to approve/reject a Furniture Request is the Project Manager. Therefore, SuiteDreams only wants the buttons to be added when the currently logged in user is the Project Manager.

Since you just created a workflow field to store the Current Approver as the Project Manager (Current Approver = Project Manager), we simply need to create a condition that verifies the user is the Current Approver.

6. With State 1: Waiting for Approval still selected, click on **Group Actions** on the Before Record Load trigger in the **State** tab.
7. Configure the following condition using the Visual Builder:

Condition			
FIELD	COMPARE TYPE	SELECTION	VALUE FIELD
User	any of		Current Approver (Workflow)

8. Click **Add** and then **Save** on the Workflow Action Group form.

Make Furniture Request Records Read-Only for Non-Approvers

Now you will configure a Lock Record action to make Furniture Requests "read-only" for all users, except the current approver, while it is waiting to be approved/rejected.

9. With State 1: Waiting for Approval still selected, configure a **Lock Record** action as follows (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Before Record Load

Create the following condition:

Condition		
FIELD	COMPARE TYPE	VALUE FIELD
User	none of	Current Approver (Workflow)

10 Click **Add** and then **Save** on the Workflow Condition form.

11 Click **Save** on the Workflow Action configuration form.

In the list of actions under the **State** tab, you should see the following:

Test Business Rule A

To test this business rule, you will create two Furniture Request records and assign a different user as the Project Manager. Based on the selected user's role, you will be able to approve/reject the record or the record will be locked as read-only.

12 As Larry Nelson, create a new Furniture Request record and select "Aubrey Pober" as the Project Manager.

13 Locate and open the new record.

You should not see the **Approve Request** or **Reject Request** buttons and the record should be locked. This is because you are not logged in as the specified Project Manager, in this case, Aubrey Pober.

14 Logout as Larry Nelson and log back in as Aubrey Pober using the login information generated in *Exercise 01*.

15 Open the new Furniture Request by navigating to **Customers > Other > Furniture Request** and click the **Edit** link next to the record.

You should see both custom buttons.

Note: If you enabled the SAVE RECORD FIRST option for both buttons, the buttons will not be displayed as separate "Approve" and "Reject" buttons. They will be grouped under the **Save** option as **Save & Approve**/**Save & Reject**.

You should also see that **Active Workflows** and **Workflow History** shows the record is waiting for approval at State 1 and Aubrey Pober is the Current Approver.

16 Logout as Aubrey Pober and log back in as Larry Nelson.

Implement Business Rule B

Create and Assign a Default Approval Status

In this series of steps, you will create a custom list representing the approval status of all Furniture Requests.

17 Navigate to **Customization > Lists, Records, & Fields > Lists > New**. The Custom List form will be displayed.

18 Enter the following information (leaving all other fields at their defaults unless otherwise noted):

NAME	Furniture Approval Status
ID	_sdr_furn_req_appr_status
DESCRIPTION	Enter a description for the custom list.
VALUES	Pending Approval Approved Rejected

19 Click **Save**.

Now let's create a new record/form field on the custom Furniture Request record to store the Furniture Approval Status.

20 Navigate to **Customization > Lists, Records, & Fields > Record Types**.

21 Select the custom **Furniture Request** record type by clicking on its link.

22 In the **Fields** sublist, click the **New Field** button.

23 Enter the following information (leaving all other fields at their defaults unless otherwise noted):

LABEL	Approval Status
ID	_sdr_furn_req_appr_status
TYPE	List/Record
LIST/RECORD	Furniture Approval Status
STORE VALUE	Checked
SHOW IN LIST	Checked
DISPLAY TYPE (Display subtab)	Inline Text

24 Click **Save**.

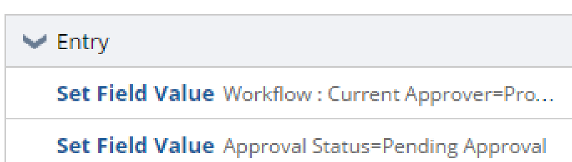
Now let's configure the appropriate action to set the status to "Pending Approval" for all new Furniture Request records.

25 Return to the Furniture Request Approval workflow.

26 Select State 1: Waiting for Approval and configure a new **Set Field Value** action as follows (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Entry
Parameters	
FIELD	Approval Status
Value (Static Value)	
SELECTION	Pending Approval

27 Click **Save** on the Workflow Action configuration form. In the list of actions under the **State** tab, you should see the following:



Now let's repeat the Set Field Value action for each of the workflow end states to set the Approval Status accordingly.

28 Select State 1: Waiting for Approval and click on the Set Field Value action that you just created above.

29 Click the **Make Copy** button and select "State 2: Request Approved" from the STATE dropdown list. You will receive the following informational message:

30 Click **OK** to acknowledge the message.

31 Configure the **Set Field Value** action as follows (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
STATE	State 2: Request Approved
Parameters	
FIELD	Approval Status
Value (Static Value)	
SELECTION	Approved

32 Click **Save** on the Workflow Action configuration form.

33 Repeat the steps above to set the APPROVAL STATUS to "Rejected" for "State 3: Request Rejected":

Basic Information	
STATE	State 3: Request Rejected
Parameters	
FIELD	Approval Status
Value (Static Value)	
SELECTION	Rejected

Test Business Rule B

34 Create two new Furniture Request records and test that the appropriate APPROVAL STATUS is set at each end state by approving one record and rejecting the other.

Implement Business Rule C

Incorporate Email Notifications

You are almost finished configuring basic approval processing for SuiteDreams Furniture Requests. The last business rule to be implemented involves adding email notifications to the Project Manager and the customer.

The table below summarizes the statuses that should be set, the recipient of the email notification, and the condition in which to execute each action:

Approval Status	Recipient	Condition
Pending Approval	Current Approver	After the Furniture Request record has been created. A good place to set this is upon entry to the state representing requests waiting for approval.
Approved	Customer	When the Furniture Request has been approved.
Rejected	Customer	When the Furniture Request has been rejected.

> It will be faster if you use the **Make Copy** (or **Save & Copy**) feature in combination with changing the value of the State field, as you did earlier. This way, the action you created in one state is moved to the appropriate state.

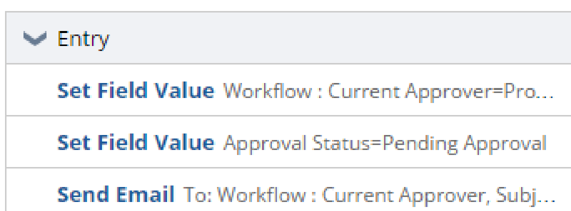
35 Edit the Furniture Request Approval workflow once again.

36 Select State 1: Waiting for Approval and configure a **Send Email** action as follows (leaving all other fields at their defaults unless otherwise noted):

> **BE CAREFUL WHERE YOU CONFIGURE THIS ACTION!**
>
> Actions configured on the same trigger are executed in the order listed in the actions list.

Basic Information		
TRIGGER ON	**Hint**: Send the email notification once, when the Current Approver first receives the record to approve or reject.	
Parameters		
Sender (FROM FIELD)	RECORD (JOIN FIELD): Current Record FIELD: User	
Recipient (FROM FIELD)	RECORD (JOIN FIELD): Current Record FIELD: Current Approver (Workflow)	
Content (CUSTOM)	SUBJECT: You have a Furniture Request to approve. BODY: A new Furniture Request is waiting for your approval. INCLUDE VIEW RECORD LINK: Checked	
Condition		
RECORD	FIELD	COMPARE TYPE
User	E-mail	not empty
Current Approver (Workflow)	E-mail	not empty

37 Click **Save** on the Workflow Action configuration form. In the list of actions under the **State** tab, you should see the following:



Now let's configure the email notification to the customer indicating the approval status of their Furniture Request. First, the approved email.

38 Select State 2: Request Approved and configure a **Send Email** action as follows (leaving all other fields at their defaults unless otherwise noted):

Basic Information		
TRIGGER ON	**Hint**: Send the email notification once, when the Current Approver first receives the record to approve or reject.	
Parameters		
Sender (FROM FIELD)	RECORD (JOIN FIELD): Current Record FIELD: User	
Recipient (FROM FIELD)	RECORD (JOIN FIELD): Current Record FIELD: Customer	
Content (CUSTOM)	SUBJECT: Great news! Your custom Furniture Request has been approved. BODY: We are pleased to inform you that your custom Furniture Request has been approved. INCLUDE VIEW RECORD LINK: Checked	
Condition		
RECORD	FIELD	COMPARE TYPE
User	E-mail	not empty
Customer	E-mail	not empty

39 Click **Save** when finished.

40 Use **Make Copy** to copy the Send Email action above for rejected Furniture Requests (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
TRIGGER ON	Entry
Parameters	
Sender (FROM FIELD)	RECORD (JOIN FIELD): Current Record FIELD: User
Recipient (FROM FIELD)	RECORD (JOIN FIELD): Current Record FIELD: Customer
Content (CUSTOM)	SUBJECT: Your custom Furniture Request has been rejected. BODY: Your custom Furniture Request has been rejected. INCLUDE VIEW RECORD LINK: Checked

Test Business Rule C

41 Create two more Furniture Request records and test the email notifications for approval and rejection, verifying they show on both the Customer and Employee records:

- **Communication > Messages** sublist on a Customer record.
- **Communication > Messages** sublist on an Employee record (Larry Nelson has many future dated messages preloaded into the training account. Page through the messages to find the email created by the workflow.

Make sure to select a customer that contains an email address. Best practice is to configure the actions with a condition that verifies the existence of an email address, (e.g. a currently logged in user must have an email address; email addresses may exist in other places if they have been made a required field).

Implement Business Rule D

Prevent all users from modifying the Furniture Request once it has reached the "Request Approved" or "Request Rejected" end states.

42 Configure the appropriate action in each end state of the workflow to make Furniture Request records *read-only* in both end states.

Test Business Rule D

43 Test by creating two new Furniture Request records—one for approval and one for rejection.

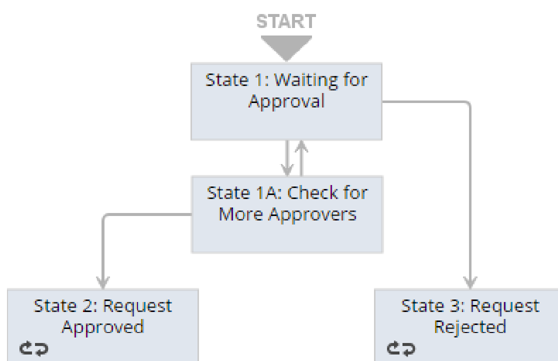
44 Verify that the records are read-only (locked) in both end states.

04: Configure Multiple Approvers

Scenario: Custom Furniture Requests are currently approved by the Project Manager. SuiteDreams has experienced some cost overruns in their projects, and because of this, are now requiring signoff by additional levels of management prior to the start of a project.

- Approvers are required up through the top of the supervisor hierarchy.
- The state representing the Project Manager approver should be reused to represent each of the other approvers.
- Each approver is sent an email about a Furniture Request waiting for their approval, as is currently the case with the Project Manager.
- A new state is added to determine whether or not additional approvals are required.

Upon completion of this exercise, your workflow should resemble the following:



Disable the Furniture Request Approval Workflow

In order to prevent workflow conflicts during the testing of this exercise, let's temporarily set the RELEASE STATUS of the Furniture Request Approval workflow to "Not Initiating".

1 Navigate to **Customization > Workflow > Workflows** and edit the Furniture Request Approval workflow.

2 Change the RELEASE STATUS from "Released" to "Not Initiating".

3 Click **Save**.

Create a New Workflow using "Make Copy"

4 From the **More** link in the upper-right corner of the Workflow Manager, click **Make Copy**. A new workflow will be created with the following name:

Furniture Request Approval (2)

5 Click the edit link represented by the pencil icon ( ) located in the **Workflow** tab and rename the workflow as follows (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
NAME	Furniture Request Multiple Approvals
ID	*customworkflow_sdr_furn_req_approva_2* **Note**: This ID will be auto-generated.
DESCRIPTION	Enter a description of the workflow.
RELEASE STATUS	Released
KEEP INSTANCE AND HISTORY	Always
ENABLE LOGGING	Checked

6 Click **Save**.

Create a New State (Check for More Approvers)

7 Add a new state called "State 1A: Check for More Approvers". Place the state before State 2 and State 3.

8 Modify the transition for the **Approve Request** button. The transition should go to this new state upon approval of the Furniture Request. The remaining transitions will be configured shortly.

Add Fields and Actions to the New State

9 Select State 1A: Check for More Approvers and add a state field to track additional approvers in the supervisor hierarchy.

> State fields are added by selecting **Fields** in the **State** subtab of the Context Panel and then clicking on the **+ New State Field** button. Unlike workflow fields, which are available to all states in the workflow, state fields are only available to the state in which they are configured.

LABEL	More Approvers
ID	_sdr_furn_req_more_appr
TYPE	Check Box
STORE VALUE	Checked

10 Click **Save**.

11 Add the appropriate action to State 1A that sets More Approvers to *true* (checked) or *false* (unchecked) based on whether the Current Approver's Supervisor field is empty:

- More Approvers should be checked when the Current Approver has a supervisor.
- More Approvers should be unchecked when the Current Approver does not have a supervisor.

12 Now add an action to set the Current Approver to the Current Approver's supervisor when the Current Approver has a supervisor.

Configure Transitions

13 Transition from State 1A back to State 1 when More Approvers is checked.

This transition is made when the workflow continues to the next approver in the hierarchy.

14 Transition from State 1A to State 2 when More Approvers is unchecked.

This indicates there are no additional approvals necessary and approval processing is complete.

Adjust Existing Actions in State 1

15 The first Set Field Value action in State 1 sets the Current Approver to the Project Manager. However, this should only execute the first time State 1 is entered, not on the transition from State 1A back to State 1.

Make the necessary adjustments to set Current Approver only when it is empty.

Test the Approval Process

> **BEFORE TESTING THIS WORKFLOW...**
>
> Remember the Return User Error action you configured back in the *Configuring Actions on Record Load* module that restricts the creation of Furniture Request records to Administrators and members of the Installation/Fitout department?
>
> Well, that same action will prevent Aubrey Pober from initiating the multiple approver workflow, because she is not an administrator or a member of the Installation/Fitout department.
>
> In order to successfully test your workflow, go back to the Furniture Request Load workflow and inactivate the Return User Error action (INACTIVE = Checked).

The following supervisor hierarchy will be used to test your multiple approvers workflow:

Aubrey Pober > Fiona Small > James Rollings > John Rittman > Larry Nelson (You)

16 Login as Aubrey Pober and create a new Furniture Request:

- Select "Fiona Small" as the PROJECT MANAGER. She will be the first approver.
- You are restricted to choosing customers whose Sales Rep is Aubrey Pober. For this test, select customer "Best Buy Software".
- Complete the remaining fields and save the record. **Workflow History** should show Fiona Small as the Current Approver.
- Logout as Aubrey Pober and back in as Fiona Small.
- Approve the request as Fiona. Approval routing moves to James Rollings, then to John Rittman, and finally to you (Larry Nelson). Keep logging in and out as each user to approve the request until the top of the supervisor hierarchy is reached.
- The Furniture Request should be set to "Approved" only after Larry Nelson has approved.
- Refer to **Workflow History** as you go through each approval process to verify proper approval routing.

17 (**Optional**) Test rejection processing in the same way.

05: Add Sales Manager Approval

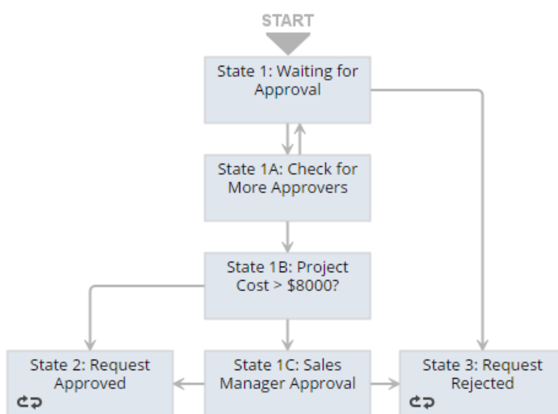
Scenario: Furniture Requests with an ESTIMATED PROJECT COST > $8,000 must be approved by the SVP, Sales, Matt Nolan (Sales Manager role). This occurs following all approvals by the Project Manager and his/her supervisors, but only when the ESTIMATED PROJECT COST is greater than $8,000.

There are two approaches to adding the additional approver:

- Modify conditions in the transitions of the first approver (if the **Approve Request** button is clicked and ESTIMATED PROJECT COST > 8,000).
- Keep the conditions in the existing transitions as is and add a new state to evaluate the new condition (if ESTIMATED PROJECT COST > 8,000). This mirrors the diamond decision shape that is typical of business process flowcharts).

The second approach is taken in this exercise since it decouples the new approver from the state containing the first approver, and it is easier to view the business process from a visual perspective.

Upon completion of this exercise, your workflow should resemble the diagram below:



Make the Necessary Adjustments

1. Edit the Furniture Request Multiple Approvals workflow.
2. Add two new states between State 1A and State 2:
   - State 1B: Project Cost > $8000?
   - State 1C: Sales Manager Approval
3. Modify the transition coming out of State 1A to go to State 1B.

4 Add two transitions out of State 1B:

- Transition to State 1C when ESTIMATED PROJECT COST > $8,000.
- Transition to State 2 when ESTIMATED PROJECT COST <= $8,000.

Extra Credit: Use a custom formula instead of the Visual Builder to define each transition condition.

5 Configure actions in State 1C:

- Add two buttons which are only available to users with the "Sales Manager" role. **Hint**: You might create an Action Group for the buttons with a shared condition!
- Make records read-only for all users except Sales Managers.

Script ID (Basic Information)	Button Label
_sdr_furn_req_smgr_appr	Sales Manager Approve
_sdr_furn_req_smgr_rej	Sales Manager Reject

6 Add the following transitions out of State 1C:

- Transition to State 2 when the **Sales Manager Approve** button is clicked.
- Transition to State 3 when the **Sales Manager Reject** button is clicked.

Execute the Workflow and Test

7 Test by creating two new Furniture Requests—one with ESTIMATED PROJECT COST > $8000 and another with ESTIMATED PROJECT COST <= $8,000.

8 Approve each Furniture Request as Larry Nelson (you).

Upon initial approval:

- If ESTIMATED PROJECT COST > $8,000, the workflow should transition to State 1C: Sales Manager Approval, and you (Larry Nelson) should not have access to the **Sales Manager Approve/Reject** buttons.
- If ESTIMATED PROJECT COST <= $8,000, the workflow should transition to State 2: Request Approved.

9 Logout as Larry Nelson and login as Matt Nolan to approve or reject.

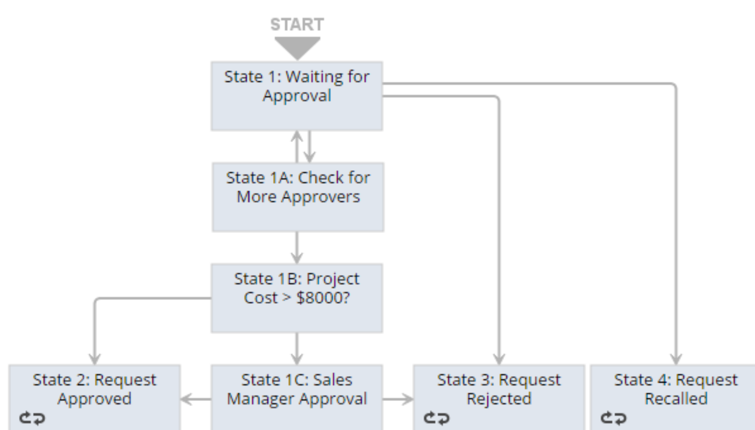
06: Allow Submitters to Recall Furniture Requests (Optional)

Scenario: SuiteDreams would like to allow submitters of Furniture Requests to recall their request as long as it is in a "Pending Approval" status. This situation applies to cases where the submitter is not the Project Manager.

SuiteDreams' business requirements for recall processing are provided below:

- The Project Manager may approve or reject the Furniture Request, as is currently allowed.
- The submitter may recall the Furniture Request. The submitter is not currently tracked but can be by setting the Current User to the submitter at the time of form submittal.
- A new "Recalled" status is used to track recalled records.
- After a Furniture Request is recalled:
  - Set the status of all recalled records to "Recalled".
  - Freeze the record in the "Request Recalled" state.
  - The record should be read-only to all users.

Upon completion of this exercise, your workflow should resemble the diagram below:



Temporarily Disable the Furniture Request Multiple Approvals Workflow

In order to prevent workflow conflicts during the testing of this exercise, let's temporarily set the RELEASE STATUS of the Furniture Request Multiple Approvals workflow.

1. Edit the Furniture Request Multiple Approvals workflow.
2. Change the RELEASE STATUS from "Released" to "Not Initiating".
3. Click **Save**.

Create a New Workflow using "Make Copy"

4 From the **More** link in the upper-right corner of the Workflow Manager, click **Make Copy**. A new workflow will be created with the following name:

Furniture Request Multiple Approvals (2)

5 Click the edit link represented by the pencil icon ( ) located in the **Workflow** tab and rename the workflow as follows (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
NAME	Furniture Request Recall
DESCRIPTION	Enter a description of the workflow.
RELEASE STATUS	Released
KEEP INSTANCE AND HISTORY	Always
ENABLE LOGGING	Checked

6 Click **Save**.

Add a "Recall Request" Button

7 Add a "Recall Request" button (*_sdr_furn_req_btn_recall*) to State 1.

Note: Do not add this button to the existing action group. It requires a different condition which you will configure shortly.

Create a New "Recalled" State

8 Add a new end state called "State 4: Request Recalled". You will add the actions later.

SuiteDreams' business requirements dictate that the record must remain at this end state for the life of the record. What needs to be configured to ensure this is the case?

Answer:

9 Configure the transition between State 1 and State 4 when the **Recall Request** button is clicked.

10 Run a quick test to make sure the user can reach State 4 and that the workflow remains active in the "Request Recalled" end state.

Create a Workflow Field to Display the Submitter

11 Add another workflow field just like you did for Current Approver:

LABEL	Submitter
ID	_sdr_furn_req_submitter
TYPE	List/Record
LIST/RECORD	Employee
STORE VALUE	Checked

Make the Recall Button Available to the Submitter

12 Set the current user in the SUBMITTER field when entering a new Furniture Request. This should only happen when the SUBMITTER has not already been set.

What is the most appropriate trigger for this action?

Answer:

13 Add a condition to the **Recall Request** button so it displays only when the Submitter is viewing the record.

14 Modify the condition on the Lock Record action in State 1 so that the record is editable for both the Current Approver and the Submitter.

> You may need to re-arrange your actions so that the Set Field Value action on the SUBMITTER field happens before the Lock Record action.

Configure Actions for the Recalled State

15 Add an action to make the record read-only to everyone.

16 Add "Recalled" to the list of approval statuses and add an action that sets the status accordingly.

Execute the Workflow and Test

17 Open up the Customer record you intend to use in your testing. For example, "B&B Design".

- Verify that Edwin Goldwasser is the SALES REP in the **Primary Information** section.

- Verify the customer is in the "US – West" subsidiary in the **Classification** section.

18 Make any necessary changes and click **Save**.

> The default behavior of the Sales Rep role restricts users to working with customers that are assigned to them via the SALES REP field. Edwin Goldwasser is in the "US – West" subsidiary, so these must be aligned as well.

19 Log in as Edwin Goldwasser and navigate to **Customers > Other > Furniture Request > New** to create a new Furniture Request record:

- Select you (Larry Nelson) as the Project Manager and "B&B Design" as the Customer.
- The **Recall Request** button should display for Mr. Goldwasser, but not for the selected Project Manager. To verify, login as Larry Nelson and make sure you do not see the **Recall Request** button.

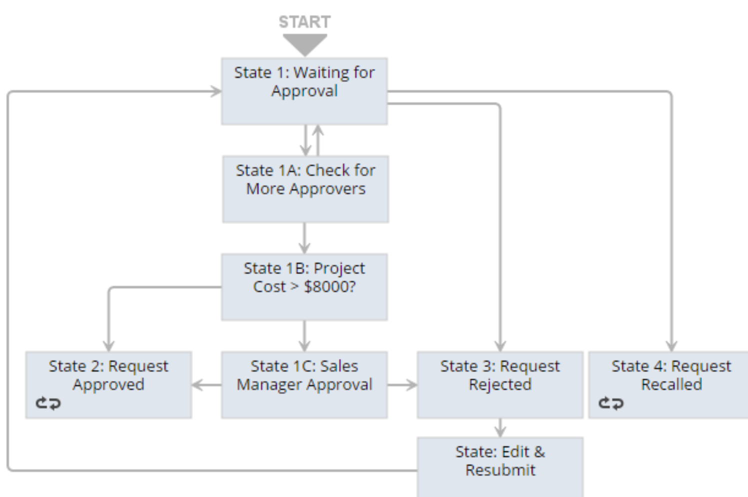
20 Click the **Recall Request** button and verify the APPROVAL STATUS is set to "Recalled" when entering State 4 and that the record is locked.

07: Allow Submitters to Resubmit after Rejection (Optional)

Scenario: The submitter of a custom Furniture Request should be allowed to resubmit in the event it was previously rejected. The overall workflow process for the submitter is described below:

- The submitter should be able to click a **Resubmit Request** button if the APPROVAL STATUS is "Rejected".
- The submitter should be able to update all fields except PROJECT MANAGER, CUSTOM, and CUSTOMER. The record should be read-only for all other users.
- The submitter saves any changes and the approval process restarts with the Project Manager as first approver and continues with the normal approval process.

Upon completion of these steps, your workflow should resemble the diagram below:



Temporarily Disable the Furniture Request Recall Workflow

In order to prevent workflow conflicts during the testing of this exercise, let's temporarily set the RELEASE STATUS of the Furniture Request Recall workflow.

1 Edit the Furniture Request Recall workflow and change the RELEASE STATUS from "Released" to "Not Initiating".

Create a New Workflow using "Make Copy"

2 From the **More** link in the upper-right corner of the Workflow Manager, click **Make Copy**.

3 Rename the workflow as follows (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
NAME	Furniture Request Resubmit
DESCRIPTION	Enter a description of the workflow.
RELEASE STATUS	Released
KEEP INSTANCE AND HISTORY	Always
ENABLE LOGGING	Checked

4 Click **Save**.

Add a "Resubmit Request" Button

5 Add a button called "Resubmit Request" (*_sdr_furn_req_btn_resubmit*) to State 3 that is only available to the submitter.

Add a New "Edit & Resubmit" State

6 Add a new state called "State: Edit & Resubmit" as depicted in the workflow diagram above.

Configure Actions and Transitions

7 Configure the following actions in State: Edit & Resubmit:

- Make the record read-only for all users except the submitter.
- Modify the display of the following fields when the current user is the submitter:
  - Hide the CUSTOM field
  - Disable the CUSTOMER and PROJECT MANAGER fields
- Clear the value of Current Approver so the approval process begins again with the Project Manager upon transition back to State 1.

8 Configure transitions to/from State: Edit & Resubmit:

- Transition to State: Edit & Resubmit from State 3 when the **Resubmit Request** button is clicked.
- Transition from State: Edit & Resubmit to State 1 upon edit/save.

Execute the Workflow and Test

9 Login as Mr. Goldwasser and create a new Furniture Request, selecting "Larry Nelson" as the Project Manager and "B&B Design" as the Customer.

- Login as the Project Manager and reject the Furniture Request so the workflow ends in State 3.
- Log back in as Edwin Goldwasser and open the Furniture Request. The record should be read-only, but you should see the **Resubmit Request** button.
- Click **Resubmit Request**.
- Open **Workflow History** and verify the record is currently in State: Edit & Resubmit.
- Click **Edit** to update the record. All fields should be editable except for PROJECT MANAGER, CUSTOM (hidden), and CUSTOMER.
- Modify one or more values and click **Save**.
- Open **Workflow History** and verify the record is now back in State 1.
- Log back in as the Project Manager (assuming the Current Approver was reset to this person).
- The Project Manager should see the **Approve Request** and **Reject Request** buttons as before.

Reset Release Statuses

Before continuing, be sure to change the RELEASE STATUS of the Furniture Request Resubmit workflow to "Not Initiating" and the Furniture Request Approval workflow from "Not Initiating" back to "Released".

All future exercises in the course are dependent upon the original Furniture Request Approval workflow.

EXERCISE SOLUTIONS

Having trouble with this module's workflow?

1 Navigate to **Customization > Workflow > Workflows**.

2 Change the RECORD TYPE filter to "Furniture Request SOLUTION".

3 Click **Edit** next to the workflow you would like to analyze:

- SOL Furniture Request Approval 09 (*Exercises 02-03*)
- SOL Furniture Request Multiple Approvals 09-4 (*Exercise 04*)
- SOL Furniture Request Multiple Approvals 09-5 (*Exercise 05*)
- SOL Furniture Request Recall 09-6 (*Optional Exercise 06*)
- SOL Furniture Request Resubmit 09-7 (*Optional Exercise 07*)

02: Create a Basic Approval Processing Workflow

16 What is the only sensible configuration for TRANSITION ON in this case?

Answer: *<blank>*

06: Allow Submitters to Recall Furniture Requests

8 SuiteDreams' business requirements dictate that the record must remain at this end state for the life of the record. What needs to be configured to ensure this is the case?

Answer: You must check the DO NOT EXIT WORKFLOW checkbox so the record remains in the "Recalled" end state.

12 What is the most appropriate trigger for this action?

Answer: Entry. We only want to set the current user as the submitter the first time entering the approval workflow upon save of the record.

UPDATING RECORD SUBLIST LINES

Module Exercises	
01	Update Sublist Field Values
02	Apply Sublist Line Discounts
03	Create a New Record based on Sublist Field Values
04	Create an Items Sublist Line
05	Create a Sublist Line Approval Workflow
06	Perform Line Validations during Data Entry
Optional Exercises	
07	Automate Line Item Discounts using Formulas
08	Close Orders with Inactive Items
09	Create an Expense Line Approval Workflow (Challenge)

As you create each modular workflow in this module, it can be helpful to test each workflow individually. This reduces the number of workflows running on the Transaction record and greatly facilitates workflow testing.

Although not instructed to do so in the exercises, consider setting the RELEASE STATUS of the workflows you have already verified to "Not Initiating". Once all sublist workflows have been tested individually, set their statuses to "Released".

01: Update Sublist Field Values

Scenario: In this exercise, you will create a workflow specifically used to process **Items** sublist lines on Transaction records.

SuiteDreams has noticed that when certain dining chairs are purchased by customers, they are almost always purchased in quantities of 4. As a result, SuiteDreams would like to facilitate the order entry process by setting the line item quantity to 4 by default when specific dining tables have been added to the order.

Create the Workflow

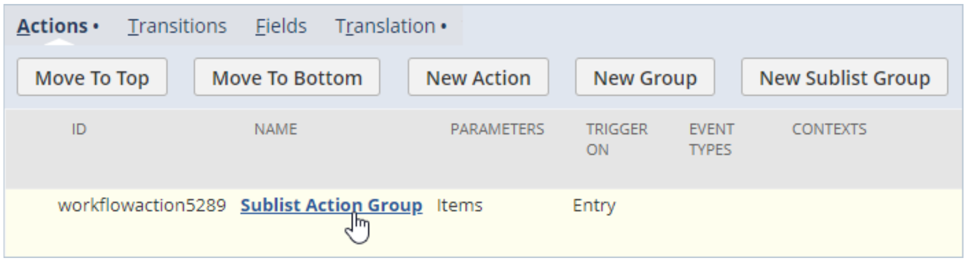
1. Create a new workflow and enter the following information (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
NAME	Update Sublist Lines
ID	_sdr_update_sublist_lines
RECORD TYPE	Transaction
SUB TYPES	Order
DESCRIPTION	Updates sublist lines on sales order transactions.
RELEASE STATUS	Released
KEEP INSTANCE AND HISTORY	Always
ENABLE LOGGING	Checked
Event Definition	
ON CREATE	Checked
TRIGGER TYPE	Before Record Submit

2. Save the workflow definition.

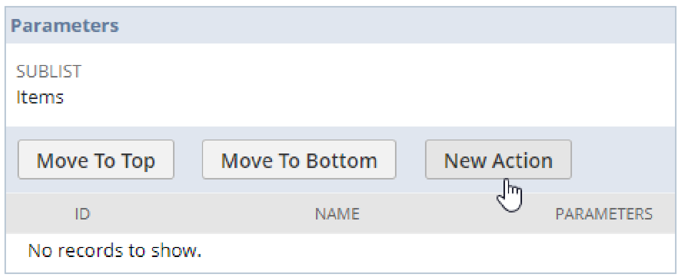
Configure Sublist Action Groups, Actions, and Conditions

3. Rename State 1 to "State 1: Update Sublist Fields" and click **Save**.
4. Re-edit State 1, select the **Actions** subtab, and click **New Sublist Group**.
5. Select "Items" from the SUBLIST dropdown list in the **Parameters** section.
6. Configure a *group-level* condition that restricts processing to only orders where "ABC Marketing" is the selected customer.
7. Save the new sublist action group.
8. Click on the Sublist Action Group link on the **Actions** subtab.



This will allow you to add actions to the group.

9 At the bottom of the Sublist Action Group form, click **New Action**.



10 Select the **Set Field Value** action and enter the following action configuration (leaving all other fields at their defaults unless otherwise noted):

Parameters	
FIELD	Quantity (Line)
Value	
STATIC VALUE	TEXT: 4

11 Enter the following *action-level* condition to restrict action processing to only sublist lines with the Armless Dining Chair:

Condition		
FIELD	COMPARE TYPE	SELECTION
Item (Line)	any of	Armless Dining Chair

12 Save the Workflow Action configuration.

13 Save the Sublist Action Group configuration.

14 Save the Workflow State configuration.

 ORACLE NETSUITE

Execute the Workflow and Test

15 Enter a new sales order (**Transactions > Sales > Enter Orders**) leaving all other fields at their defaults unless otherwise noted):

- CUSTOMER = "ABC Marketing"
- ITEM (Line 1) = Select any dining table item
- ITEM (Line 2) = "Armless Dining Chair"

16 Save the record and examine the **Items** sublist.

The Armless Dining Chair sublist line now displays a quantity of 4.

17 Select the **System Information** subtab, **Workflow History** sublist and open the Workflow Execution Log:

ENTRY	RESULT	DATE/TIME
Workflow initiated		**5/18/2020 - 07:41:14.739**
Running ONENTRY trigger under BEFORESUBMIT (Event: CREATE; Context: USERINTERFACE)		**5/18/2020 - 07:41:14.743**
SUBLISTACTIONGROUP: Items	Executed	5/18/2020 - 07:41:14.747
Condition: Customer = ABC Marketing Inc	True	5/18/2020 - 07:41:14.755
Line #1		5/18/2020 - 07:41:14.767
Line #2		5/18/2020 - 07:41:14.789
SETFIELDVALUE	Executed	5/18/2020 - 07:41:14.798
Condition: Line : Item = Armless Dining Chair	True	5/18/2020 - 07:41:14.800
quantity = 4		5/18/2020 - 07:41:14.894
Running AFTERSUBMIT trigger (Event: CREATE; Context: USERINTERFACE)		**5/18/2020 - 07:41:21.221**
Workflow finished		**5/18/2020 - 07:41:21.226**

Here you can see the evaluation of each condition (group and action) and the resulting QUANTITY field update on Line 2, representing the Armless Dining Chair.

18 Return to the Update Sublist Lines workflow, edit the sublist action group, and **remove** the group-level condition that restricts processing to ABC Marketing.

19 Save the sublist action group.

02: Create an Items Sublist Line

Scenario: In this exercise, you will create a new promotional sublist line item based on the presence of specific items added to an order. In this particular use case, items eligible for the promotion as well as the promotional items themselves are hard-coded in the workflow configuration using a combination of custom workflow fields and Set Field Value actions.

Modify the Update Sublist Lines Workflow

1. Edit the Update Sublist Lines workflow and add a new state called "State 2: Add Promo Item(s)".
2. Create a transition between State 1 and State 2.

Create the Necessary Workflow Fields

3. Create two custom workflow fields as follows:

   Workflow field **1 of 2**:

LABEL	Chair Promotion
ID	_sdr_chair_promo
TYPE	Check Box
STORE VALUE	Checked

   Workflow field **2 of 2**:

LABEL	Eligible Item
ID	_sdr_eligible_item
TYPE	Check Box
STORE VALUE	Checked

Configure the Sublist Action Groups and Set Field Value Actions

In order to add the promotional line items, the workflow must detect the presence of eligible items in the order using the custom workflow fields you just created.

4. Create a new Sublist Action Group in State 2. No conditions are required.
5. Configure a **Set Field Value** action within the new group that sets the Chair Promotion (Workflow) field to *true* when the order contains 4 or more Armless Dining Chairs.

6 Create a second Sublist Action Group in State 2. Again, no conditions are required.

7 Configure another **Set Field Value** action within this new group that sets the Eligible Item (Workflow) field to *true* when the order contains the Loft Dining Table.

8 Finally, create a third Sublist Action Group and configure a **Return User Error** action that alerts the user when more than one promotional item is added to the order.

For example: *You cannot add more than one promotional item to an order. Please remove one of the items*.

Configure the Create Line Actions

Now let's configure the necessary Create Line actions to add each promotional item accordingly.

9 With State 2 still selected, create two Create Line actions.

Create Line action **1 of 2**:

Parameters subtab		
SUBLIST	Items	
POSITION	After Last Line	
FIELD	TEXT	SELECTION
Item (Line)		Armless Dining Chair
Quantity (Line)	2	
Rate (Line)	0.00	
Description (Line)	Chair Promotion	
Condition		
FIELD	COMPARE TYPE	
Chair Promotion (Workflow)	checked	

Create Line action **2 of 2**:

Parameters subtab		
SUBLIST	Items	
POSITION	After Last Line	
FIELD	TEXT	SELECTION
Item (Line)		Armless Dining Chair
Quantity (Line)	4	
Rate (Line)	0.00	
Description (Line)	Chair Promotion	
Condition		
FIELD	COMPARE TYPE	
Eligible Item (Workflow)	checked	

> ⚠ Be aware of the order of action execution in State 2, particularly between the Return User Error and Create Line actions.

Execute the Workflow and Test

10 Enter a new order and add the Loft Dining Table **and** Armless Dining Chair to the order.

A SuiteScript Error page will be displayed as a result of the Return User Error action that checks for the presence of multiple promotional items.

11 Return to the order, re-select a customer, and add the Loft Dining Table to the order.

12 Save the order and examine the **Items** sublist.

Four Armless Dining Chairs are added to the order as promotional items.

13 Enter another order and add the following items to the order:

- Armless Dining Chair | QTY: 4

14 Save the order and examine the **Items** sublist.

Two additional Armless Dining Chairs are added to the order as promotional items.

03: Apply Sublist Line Discounts

Scenario: In this exercise, you will practice updating sublist lines in a transaction by referencing transaction body and sublist line fields in *formulas*. SuiteDreams would like to offer select customer discounts on select dining tables when they purchase a specific number of chairs.

Modify the Update Sublist Lines Workflow

1 Edit the Update Sublist Lines workflow and create a custom workflow field as follows:

LABEL	Quantity Met
ID	_sdr_item_qty_met
TYPE	Check Box
STORE VALUE	Checked

This field will be used to determine if the quantity of a particular item has been met.

2 Save the workflow field.

3 Select State 1 and add another **Set Field Value** action to the existing Sublist Action Group that sets the Quantity Met (Workflow) field to *true* when the quantity of the Armless Dining Chair line item is greater than or equal to 4.

4 Save the condition, Workflow Action configuration, and Sublist Action Group.

5 Add a new state called "State 3: Update Pricing based on Qty" and create a transition from State 2 to State 3.

Configure Sublist Action Groups, Actions, and Conditions

6 Add a new Sublist Action Group in State 3 and configure a group-level condition so actions are only executed when the specified item's quantity has been met.

7 Configure a **Set Field Value** action within the group that applies a 15% discount on the sublist RATE field using a formula. Configure a formulaic condition on the action that only applies the discount to line items with a rate greater than $1500.

8 Save the action configuration, Sublist Action Group, and state configuration.

Execute the Workflow and Test

9 Enter a new sales order (**Transactions > Sales > Enter Orders**) leaving all other fields at their defaults unless otherwise noted):

- ITEM (Line 1) = "Barrow Dining Table"
  Note the item's current price ($1,664.08) in the RATE column
- ITEM (Line 2) = "Armless Dining Chair"

10 Save the record and examine the **Items** sublist.

Note the updated RATE for the Barrow Dining Table item.

11 Select the **System Information** subtab, **Workflow History** sublist and open the Workflow Execution Log for State 3:

ENTRY	RESULT	DATE/TIME
Running ONENTRY trigger under BEFORESUBMIT (Event: CREATE; Context: USERINTERFACE)		**7/14/2020 - 09:38:43.498**
SUBLISTACTIONGROUP: Items	Executed	7/14/2020 - 09:38:43.501
Condition: Workflow : Quantity Met = T	True	7/14/2020 - 09:38:43.508
Line #1		7/14/2020 - 09:38:43.509
SETFIELDVALUE	Executed	7/14/2020 - 09:38:43.515
Condition: Line : Rate > 1500.00	True	7/14/2020 - 09:38:43.519
rate = 1414.468		7/14/2020 - 09:38:43.523
Line #2		7/14/2020 - 09:38:43.545
SETFIELDVALUE	Considered	7/14/2020 - 09:38:43.549
Condition: Line : Rate > 1500.00	False	7/14/2020 - 09:38:43.551
Line #3		7/14/2020 - 09:38:43.551
SETFIELDVALUE	Considered	7/14/2020 - 09:38:43.558
Condition: Line : Rate > 1500.00	False	7/14/2020 - 09:38:43.560
Transition to State 4: Notify Sales Rep	Executed	7/14/2020 - 09:38:43.562
Running ONEXIT trigger under BEFORESUBMIT (Event: CREATE; Context: USERINTERFACE)		**7/14/2020 - 09:38:43.563**

Here you can see the evaluation of each condition (group and action) and the resulting RATE field update on Line 1, representing the Barrow Dining Table. You can also see that the Set Field Value action on the RATE field was also considered on Line 2, but the action condition was not met.

04: Create a New Record based on Sublist Field Values

Scenario: In this exercise, you will configure a Sublist Action Group that executes a Create Record action to create a Phone Call record. The Phone Call record will be associated with the selected customer's assigned sales rep.

SuiteDreams would like to make sure the action is only executed when the order amount is greater than the selected customer's credit limit and any line amount is greater than the customer's available credit. This conditional logic will be configured as a formula referencing both transaction body and sublist line fields.

Modify the Update Sublist Lines Workflow

1. Edit the Update Sublist Lines workflow and add a new state called "State 4: Notify Sales Rep".
2. Create a transition between State 3 and State 4.

Configure Sublist Action Groups, Actions, and Conditions

3. Add a Sublist Action Group to the new state.
4. Configure a **Create Record** action in the group as follows (leaving all other fields at their defaults unless otherwise noted):

Parameters					
RECORD TYPE	Phone Call				
FIELD	TEXT	DATE	SELECTION	JOIN	VALUE FIELD
Title	Customer to exceed credit limit!				
Status			Scheduled		
Assigned				Customer	Sales Rep
Start Date		today			

Message	FORMULA: `'Order #' || {transactionnumber} || ' contains a line item (' || {line.item} || ') whose line amount is greater than ' || {entity.entityid} || ' overdue balance of $' || {entity.overduebalance} || '. Please contact this customer immediately.'`

5 Configure a *formulaic condition* that restricts the creation of the Phone Call record to only when the order total is greater than the customer's credit limit less their overdue balance **and** only for the first line item added to the order.

Hint: The *Line (Line)* field can be used to process individual line items.

6 (**Optional**) Add a Send Email action that notifies the Sales Rep on the order that the phone call has been scheduled and is awaiting completion.

Configure Transitions

In order for the Create Record action to execute properly, consider the workflow triggers supported by the Create Record action. The Create Record action is typically configured on Before Record Submit or After Record Submit.

Now consider the nature of the formula you created to populate the MESSAGE field on the Phone Call record. One of the first thing it prints is the Order #, which is not generated until the record has been saved to the database.

7 Select the appropriate trigger on the transition from State 3 to State 4 to support the Create Record action.

Execute the Workflow and Test

8 Navigate to **Transactions > Sales > Enter Orders** and enter the following information (leaving all other fields at their defaults unless otherwise noted):

- CUSTOMER = "Crown Solutions"
- ITEM (Line 1) = "Barrow Dining Table" | QUANTITY = 2

  NetSuite detects that the order total is likely to exceed the selected customer's credit limit and displays the appropriate alert message.

- ITEM (Line 2) = "Armless Dining Chair" | QUANTITY = 2

9 Save the order.

10 Select the **System Information** subtab, **Workflow History** sublist and examine the Workflow Execution Log for State 4.

ENTRY	RESULT	DATE/TIME
Running ONENTRY trigger under AFTERSUBMIT (Event: CREATE; Context: USERINTERFACE)		**7/14/2020 - 10:07:15.618**
SUBLISTACTIONGROUP: Items	Executed	7/14/2020 - 10:07:15.624
Line #1		7/14/2020 - 10:07:15.626
CREATERECORD: Phone Call (PHONECALL)	Executed	7/14/2020 - 10:07:15.634
Condition: {total} > {entity.creditlimit}-{entity.overduebalance} AND {line.line} = 1	True	7/14/2020 - 10:07:15.647
Record type: Phone Call (PHONECALL)		7/14/2020 - 10:07:15.647
title = Customer to exceed credit limit!		7/14/2020 - 10:07:15.689
status = SCHEDULED		7/14/2020 - 10:07:15.693
assigned = -5		7/14/2020 - 10:07:15.694
startdate = 7/14/2020		7/14/2020 - 10:07:15.703
message = Order #2212 contains a line item (DINING ROOM : Barrow Dining Table) whose line amount is greater than Crown Solutions overdue balance of $2900. Please contact this customer immediately.		7/14/2020 - 10:07:15.703
New record ID: 446		7/14/2020 - 10:07:15.751
Line #2		7/14/2020 - 10:07:15.761
CREATERECORD: Phone Call (PHONECALL)	Considered	7/14/2020 - 10:07:15.764
Condition: {total} > {entity.creditlimit}-{entity.overduebalance} AND {line.line} = 1	False	7/14/2020 - 10:07:15.775
Line #3		7/14/2020 - 10:07:15.775
CREATERECORD: Phone Call (PHONECALL)	Considered	7/14/2020 - 10:07:15.781
Condition: {total} > {entity.creditlimit}-{entity.overduebalance} AND {line.line} = 1	False	7/14/2020 - 10:07:15.791
Workflow finished		**7/14/2020 - 10:07:15.795**

Notice that the related Phone Call record is only created as a result of the processing of *Line ID = 1* (for the Barrow Dining Table). Not adding the Line (Line) = 1 condition would result in 3 Phone Call records being created…one for each line item.

11 Navigate to **Activities > Scheduling > Phone Calls**.

You should see the scheduled Phone Call record with the same message text as printed in the Workflow Execution Log.

05: Create a Sublist Line Approval Workflow

Scenario: In this exercise, you will use Sublist Action Groups and custom Transaction Line fields to configure a sublist line approval workflow. During the entry of a new transaction, each line in the transaction will be evaluated to determine whether the amount of each line requires manager approval and/or VP approval based on the following conditions:

- Sublist lines with an AMOUNT <= 500.00 do not require approval
- Sublist lines with an AMOUNT > 500.00 and <=1000.00 require manager approval
- Sublist lines with an AMOUNT > 1000.00 require manager approval and VP approval

Create Line Approval Statuses and a Custom Transaction Line Field

In this series of steps, you will create the necessary SuiteBuilder components to support the update and approval of transaction sublist lines. Specifically, you will create a custom list for each approval status and then add a custom transaction Line field to the Items sublist on transaction records that indicates the line's current approval status.

1 Login as Larry Nelson (Administrator), navigate to **Customization > Lists, Records, & Fields > Lists > New**, and enter the following information (leaving all other fields at their defaults unless otherwise noted):

NAME	Line Approval Status
ID	_sdr_line_appr_status
DESCRIPTION	Enter a brief description of the custom list.
Values subtab	
VALUE	No Approval Required Pending Manager Approval Pending VP Approval Pending Manager and VP Approval Approved Rejected

2 Save the custom list.

3 Navigate to **Customization > Lists, Records, & Fields > Transaction Line Fields > New** and enter the following information (leaving all other fields at their defaults unless otherwise noted):

LABEL	Line Approval Status
ID	_sdr_line_appr_status
DESCRIPTION	Enter a brief description of the custom transaction line field.
TYPE	List/Record
LIST/RECORD	Line Approval Status
STORE VALUE	Checked
Applies To subtab	
SALE ITEM	Checked **Note**: Other Transaction sub-types can be selected as desired.

4 Save the custom transaction line field.

Create the Sublist Line Approval Workflow

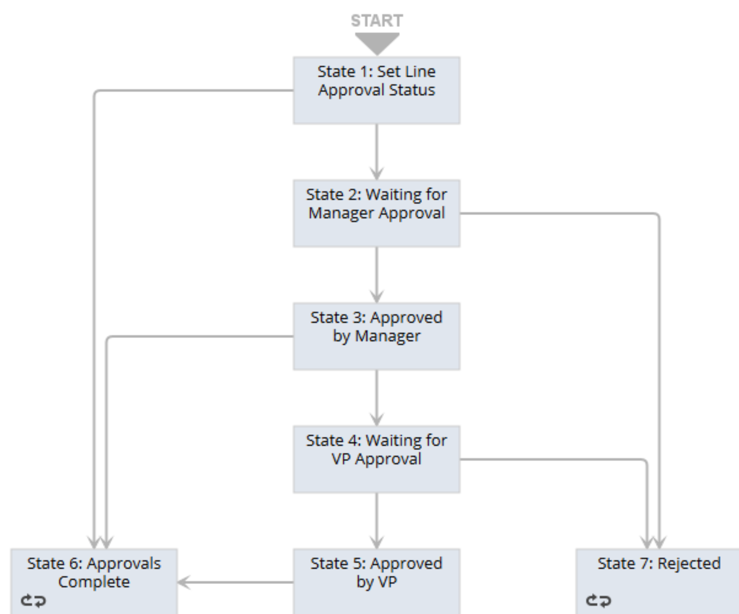
5 Create a new workflow with the following configuration (keeping all other fields at their defaults unless otherwise noted):

Basic Information	
NAME	Sublist Line Approval
ID	_sdr_sublist_line_appr
RECORD TYPE	Transaction
SUB TYPES	Order
DESCRIPTION	Workflow to control the approval of sublist lines based on line amount.
RELEASE STATUS	Released
KEEP INSTANCE AND HISTORY	Always
ENABLE LOGGING	Checked

Event Definition	
ON CREATE	Checked
TRIGGER TYPE	Before Record Submit

6 Save the new workflow definition.

7 Build out the remaining workflow states and transitions, as illustrated below. You will configure the individual groups, actions, transitions, and conditions shortly.



Configure the Necessary Workflow Fields

In order to flag which sublist lines require manager and/or VP approval, you will create four workflow instance fields that will be populated when a sublist line meets one of the approval conditions above.

8 Create the following workflow fields:

- Manager Approval Required (`_sdr_sag_mgr_appr_req`)
- VP Approval Required (`_sdr_sag_vp_appr_req`)

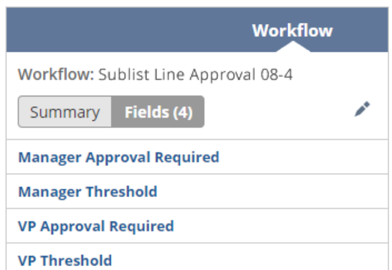
TYPE	Check Box
STORE VALUE	Checked

9 Create two more workflow fields:

- Manager Threshold (`_sdr_sag_mgr_threshold`) = 500.00
- VP Threshold (`_sdr_sag_vp_threshold`) = 1000.00

TYPE	Currency
STORE VALUE	Checked
Validation & Defaulting subtab	
DEFAULT VALUE	Enter threshold amount above.

When you have finished, the **Workflow** tab should resemble the following:



Configure States, Actions, and Conditions

Configure State 1

When a new order is saved, a Sublist Action Group iterates over the **Items** sublist and sets the LINE APPROVAL STATUS based on the conditions provided in the exercise scenario. Let's configure the group and necessary Set Field Value actions.

10 Edit State 1 and add a Sublist Action Group that will only be executed during the initial creation of the Transaction record.

11 Configure the following three **Set Field Value** actions that set the appropriate approval status for each sublist line based on the AMOUNT field.

Set Field Value action **1 of 3**:

Parameters		
FIELD	Line Approval Status (Line)	
STATIC VALUE	SELECTION: No Approval Required	
Condition		
FIELD	COMPARE TYPE	VALUE FIELD
Amount (Line)	less than or equal	Manager Threshold (Workflow)

Set Field Value action **2 of 3**:

Parameters		
FIELD	Line Approval Status (Line)	
STATIC VALUE	SELECTION: Pending Manager Approval	
Condition		
FIELD	COMPARE TYPE	VALUE FIELD
Amount (Line)	greater than	Manager Threshold (Workflow)
Amount (Line)	less than or equal	VP Threshold (Workflow)

Set Field Value action **3 of 3**:

Parameters		
FIELD	Line Approval Status (Line)	
STATIC VALUE	SELECTION: Pending Manager and VP Approval	
Condition		
FIELD	COMPARE TYPE	VALUE FIELD
Amount (Line)	greater than	VP Threshold (Workflow)

Configure two more **Set Field Value** actions that set each workflow instance field based on the value of each line's AMOUNT field.

Set Field Value action **1 of 2**:

Parameters subtab		
FIELD	Manager Approval Required (Workflow)	
STATIC VALUE	CHECKED: Enabled	
Condition		
FIELD	COMPARE TYPE	VALUE FIELD
Amount (Line)	greater than	Manager Threshold (Workflow)

Set Field Value action **2 of 2**:

Parameters subtab		
FIELD	VP Approval Required (Workflow)	
STATIC VALUE	CHECKED: Enabled	
Condition		
FIELD	COMPARE TYPE	VALUE FIELD
Amount (Line)	greater than	VP Threshold (Workflow)

Pay close attention to the order of the Set Field Value actions in the Sublist Action Group. One set of actions is dependent on the other!

12 Configure a transition from State 1 to State 2 when manager approval is *required*.

13 Configure a transition from State 1 to State 6 when manager approval is *not required*.

Configure State 2

Now configure the necessary actions and transitions for State 2, which sets the status of the order to "Pending Approval" and presents the approver with **Approve** and **Reject** buttons.

14 Select State 2: Waiting for Manager Approval and add a **Set Field Value** action that sets ORDER STATUS to "Pending Approval".

15 Add a standard Workflow Action Group (*Group 1*) and configure two **Add Button** actions—one labeled "Approve" and the other labeled "Reject". Configure the group condition so the buttons only display for users with a Sales Manager role and when manager approval is required.

Do you remember what the preferred server trigger is for the Workflow Action Group containing the button actions? **Answer**: ______________________

16 (**Optional, but recommended**) To reduce the number of buttons displayed on the form and avoid confusion between approval buttons, add a second Workflow Action Group (*Group 2*) and add three **Remove Button** actions to remove the following standard NetSuite buttons:

- Approve (this is so you do not get confused as to which Approve button to click)
- Create Deposit
- Manage Revenue Recognition

17 Add a Lock Record action to Group 2.

18 Configure two transitions—one from State 2 to State 3 and another from State 2 to State 7—based on each button click.

Configure State 3

19 Create a new Sublist Action Group and configure two **Set Field Value** actions that set the LINE APPROVAL STATUS sublist field to either "Approved" or "Pending VP Approval".

Set Field Value action **1 of 2**:

Parameters subtab		
FIELD	Line Approval Status (Line)	
STATIC VALUE	SELECTION: Approved	
Condition		
FIELD	COMPARE TYPE	SELECTION
Line Approval Status (Line)	any of	Pending Manager Approval

Set Field Value action **2 of 2**:

Parameters subtab		
FIELD	Line Approval Status (Line)	
STATIC VALUE	SELECTION: Pending VP Approval	
Condition		
FIELD	COMPARE TYPE	SELECTION
Line Approval Status (Line)	any of	Pending Manager and VP Approval

20 Configure a third **Set Field Value** action on entry to the state that sets the Manager Approval Required (Workflow) field to *false*.

21 Configure two transitions—from State 3 to State 4 and another from State 3 to State 6—based on whether VP approval is required or not.

Configure State 4

22 Once again, add a standard Workflow Action Group (*Group 1*) and configure two **Add Button** actions for both Approve/Reject buttons. Configure the group condition so the buttons only display for users with the "VP WW Sales – HQ" role and when manager approval is not required.

23 (**Optional, but recommended**) Add another Workflow Action Group (*Group 2*) and add three **Remove Button** actions to remove the following NetSuite buttons:

- Approve (this is so you do not get confused as to which Approve button to click)
- Create Deposit
- Manage Revenue Recognition

24 Add a Lock Record action to Group 2.

25 Configure two transitions—one from State 4 to State 5 and another from State 4 to State 7—based on each button click.

Configure State 5

26 Create a new Sublist Action Group and configure a single **Set Field Value** action that sets the Line Approval Status (Line) sublist field to "Approved" when the line approval status is "Pending VP Approval".

27 Configure another **Set Field Value** action on entry to the state that sets the VP Approval Required (Workflow) field to *false*.

28 Verify that you have a transition in place from State 5 to State 6. No further configuration is necessary.

Configure State 6

29 Configure a single **Set Field Value** action on entry to the state that sets the order status to "Pending Fulfillment", the next step in the order fulfillment process.

Configure State 7

30 Configure a single **Set Field Value** action on entry to the state that sets the order status to "Closed".

Execute the Workflow and Test

BEFORE TESTING THIS EXERCISE...

Edit John Rittman's Employee record and assign him the "VP WW Sales – HQ" user role on the **Access** subtab. This is to ensure he is able to approve and reject sales orders at the VP level.

31 Logout as Larry Nelson and log back in as Aubrey Pober or Edwin Goldwasser (Sales Persons). Both users can enter new orders and initiate the line approval process.

32 Enter a new sales order by navigating to **Opportunities > Transactions > Orders > New** and add the following line-items:

- Urban Dining Table ($1571.58, requires both manager and VP approval)
- White Shag Rug 10x5 ($513.38, requires manager approval)
- Armless Dining Chair ($62.90, requires no approval)

33 Save the record and note the order #.

34 Examine the approval statuses in the LINE APPROVAL STATUS column in the **Items** sublist. Each status is the result of the Set Field Value conditions against each sublist line's AMOUNT field.

35 Logout as Aubrey Pober/Edwin Goldwasser and log back in as Fiona Small (Sales Manager).

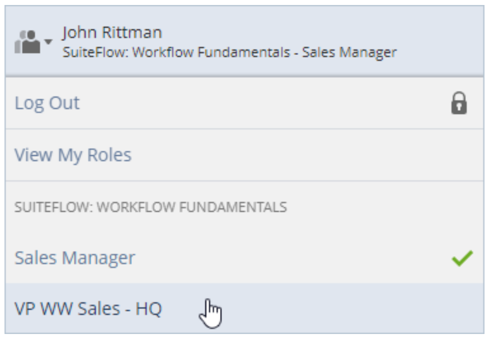
36 In the NetSuite **Search** field, enter *order:<order#>* **or** navigate to **Opportunities > Transactions > Orders** and open the order created above.

37 Click **Approve** and re-examine the LINE APPROVAL STATUS column for the White Shag Rug line item.

Notice that it is now "Approved", yet the Urban Dining Table with an amount greater than $1000 is still waiting for VP approval. Also notice that the order's overall status is still "Pending Approval".

38 Logout as Fiona Small and log back in as John Rittman (Sales Manager/VP).

39 By default, John Rittman will be logged in with the *Sales Manager* role. Change his role to "VP WW Sales – HQ".



40 Navigate to **Opportunities > Transactions > Orders**, open the order, and click **Approve**.

41 Re-examine the LINE APPROVAL STATUS column for the Urban Dining Table line item.

Notice that all sublist lines requiring approval have been approved and the order status has been updated to "Pending Fulfillment".

06: Perform Line Validations during Data Entry

Scenario: In this exercise, you will perform data entry validations based on sublist line field values using the Return User Error and Set Field Value actions configured on the After Field Edit client trigger.

First you will restrict the quantity of select items to 2 per sales order. If the user attempt to enter a quantity greater than 2 in the QUANTITY sublist line field, an alert message will be displayed indicating the item's quantity cannot be greater than 2. Then you will automatically set the LINE APPROVAL STATUS to "Approved" for any line item less than $100.

Create the Workflow

1 Create a new workflow as follows (leaving all other fields at their defaults unless otherwise noted):

Basic Information	
NAME	Line Item Entry Validations
ID	_sdr_line_item_entry
RECORD TYPE	Transaction
SUB TYPES	Order
DESCRIPTION	Workflow to perform sublist line field validations during entry of a sales order.
RELEASE STATUS	Released
KEEP INSTANCE AND HISTORY	Always
ENABLE LOGGING	Checked
Event Definition	
ON CREATE	Checked
ON VIEW OR UPDATE	Checked
TRIGGER TYPE	Before Record Load

2 Save the new workflow definition.

Configure States, Actions, and Conditions

3 Rename State 1 to "State 1: Validate Line Quantity" and enable the DO NOT EXIT WORKFLOW checkbox.

4 Select State 1 and add a **Return User Error** action as follows (leaving all other fields at their defaults unless otherwise noted):

TRIGGER ON	After Field Edit	
TRIGGERING CLIENT FIELDS	SUBLIST: Checked | Items TRIGGERING CLIENT FIELDS: Quantity (Line)	
Parameters		
TEXT	The quantity limit for the selected item is 2. Please adjust the quantity accordingly.	
Condition		
FIELD	COMPARE TYPE	SELECTION
Item (Line)	any of	Barrow Dining Table Loft Dining Table Parsons Dining Table Teak Dining Table Urban Dining Table
Quantity (Line)	greater than	2

5 Save the action configuration.

6 Now add a **Set Field Value** action as follows (leaving all other fields at their defaults unless otherwise noted):

TRIGGER ON	After Field Edit	
TRIGGERING CLIENT FIELDS	SUBLIST: Checked | Items TRIGGERING CLIENT FIELDS: Rate (Line)	
Parameters		
FIELD	Line Approval Status (Line)	
VALUE	STATIC VALUE | SELECTION: Approved	
Condition		
FIELD	COMPARE TYPE	SELECTION
Rate (Line)	less than	100.00

7 Save the action configuration.

Execute the Workflow and Test

8 Create a new sales order transaction (**Transactions > Sales > Enter Orders**) and select a customer from the CUSTOMER dropdown list.

9 Add any one of the dining tables configured in the condition above to the order and set the value of the sublist QUANTITY field greater than 2.

Upon tabbing out (or *removing focus*) from the sublist QUANTITY field the Return User Error action is executed.

10 Acknowledge the message. The quantity entered remains in the QUANTITY field.

11 Set the quantity to 2 and tab out of the field. The Return User Error action is not executed.

12 Add the Armless Dining Chair as a second line item.

Once you select the item from the dropdown list, scroll to the right in the sublist and verify that the Line Approval Status has been set to "Approved".

It is not necessary to complete the submission of the order.

07: Automate Line Item Discounts using Formulas (Optional)

Scenario: In this optional exercise, you will construct several formulas as part of Set Field Value actions configured within Sublist Action Groups.

SuiteDreams would like to automate the line item discount process during order entry using workflow. They would like to offer customers 15% off any item over $1000. When their order is entered, they would like to notify the customer with the order details, such as a list of discounted items and the total discount amount.

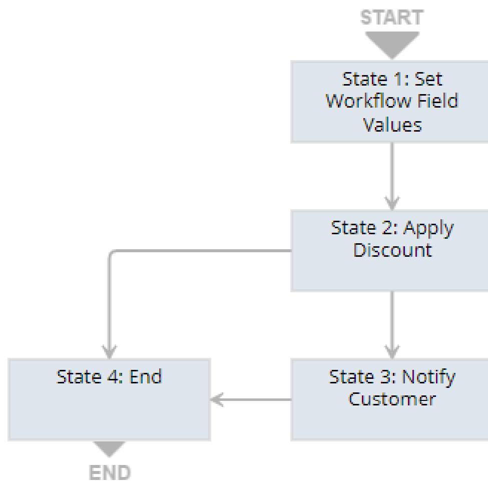
Create the Line Discount Workflow

1. Create a new workflow by entering the following information (keeping all other fields at their defaults unless otherwise noted):

Basic Information	
NAME	Sublist Line Discount
ID	_sdr_sublist_line_disc
RECORD TYPE	Transaction
SUB TYPES	Order
DESCRIPTION	Provides select customers with 15% off when they buy $1000 or more.
RELEASE STATUS	Released
KEEP INSTANCE AND HISTORY	Always
ENABLE LOGGING	Checked
Event Definition	
ON CREATE	Checked
TRIGGER TYPE	Before Record Submit

2. Save the new workflow definition.

3 Stub out the remaining workflow states and transitions, as illustrated below. You will configure the individual groups, actions, transitions, and conditions shortly.



***Note**: There will be no configured actions in State 4.*

Configure the Necessary Workflow Fields

4 Configure the following workflow fields, making sure the STORE VALUE option is enabled for each field:

- Discount Percent (`_sdr_sub_disc_percent`) | TYPE = Percent
- Discount Threshold (`_sdr_sub_disc_threshold`) | TYPE = Currency
- Discount Total (`_sdr_sub_disc_total`) | TYPE = Free-Form Text
- Original Total (`_sdr_sub_orig_total`) | TYPE = Currency
- Discounted Items (`_sdr_sub_disc_items`) | TYPE = Free-Form Text

Now you will configure the individual Sublist Action Groups, actions, transitions and associated conditions for each state of the workflow.

Configure States, Actions, and Conditions

Configure State 1

5 Add three **Set Field Value** actions that set initial values for the following workflow fields:

- Discount Percent (Workflow) = 15%
- Discount Threshold (Workflow) = 1000.00
- Original Total (Workflow) = Current Record | Total

Configure State 2

6 Create a Sublist Action Group and configure two **Set Field Value** actions as follows:

- Construct a formula that applies a 15% discount to the RATE field for any item with a RATE greater than the Discount Threshold:

  Rate (Line) = ____________________________________

- Construct a formula that populates the Discounted Items workflow field with any line item with a RATE greater than the Discount Threshold:

  Discounted Items (Workflow) = ____________________________________

7 Configure the proper transition from State 2 to State 3 to support a Send Email action, which you will configure shortly.

8 Configure the transition from State 2 to State 4 when Discounted Items is empty (no discounts applied).

Configure State 3

9 Add a **Set Field Value** action that calculates the Discount Total (Workflow) using a formula.

Discount Total (Workflow) = ____________________________________

10 Add a **Send Email** action with the following configuration:

Parameters | Sender	
FROM FIELD	RECORD (JOIN FIELD): Current Record FIELD: Sales Rep
Parameters | Recipient	
FROM FIELD	RECORD (JOIN FIELD): Current Record FIELD: Customer
Content	
SUBJECT	You received a discount!
BODY	You received a discount on the following items: `{custworkflow_sdr_sub_disc_items}` You saved `{custworkflow_sdr_sub_disc_total}`
Condition	
Just as you have done several times in this course, protect the Send Email action by defining a condition that verifies the sender and recipient both have an email address on their respective Entity records.	

11 Save the Send Email action configuration.

Execute the Workflow and Test

12 Enter a new sales order (leaving all other fields at their defaults unless otherwise noted):

- CUSTOMER = "ABC Marketing Inc"
- ITEM (Line 1) = Select any item **greater than** $1000
- ITEM (Line 2) = Select any item **greater than** $1000
- ITEM (Line 3) = Select any item **greater than** $500 but **less than** $1000
- ITEM (Line 4) = Select any item **less than** $500

13 Save the record and examine each state's log in **Workflow History**:

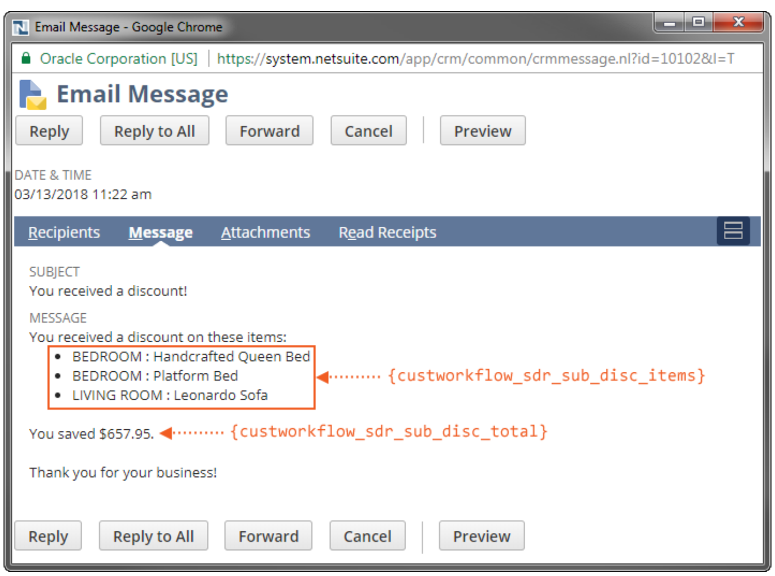
WORKFLOW	STATE NAME INFO	DATE ENTERED STATE ▾	DATE EXITED STATE	OPTIONS
Sublist Line Discounts 08-4	State 4: End	3/13/2018 11:22 am	3/13/2018 11:22 am	Discount Threshold: 1000.00 Discount Percent: 15.0% Discount Total: $657.95 Discounted Items: • BEDROOM : Handcrafted Queen Bed • BEDROOM : Platform Bed • LIVING ROOM : Leonardo Sofa Original Total: 4781.33
Sublist Line Discounts 08-4	State 3: Notify Customer	3/13/2018 11:22 am	3/13/2018 11:22 am 3	Discount Threshold: 1000.00 Discount Percent: 15.0% Discount Total: $657.95 Discounted Items: • BEDROOM : Handcrafted Queen Bed • BEDROOM : Platform Bed • LIVING ROOM : Leonardo Sofa Original Total: 4781.33
Sublist Line Discounts 08-4	State 2: Apply Discount	3/13/2018 11:22 am	3/13/2018 11:22 am 2	Discount Threshold: 1000.00 Discount Percent: 15.0% Discounted Items: • BEDROOM : Handcrafted Queen Bed • BEDROOM : Platform Bed • LIVING ROOM : Leonardo Sofa Original Total: 4781.33
Sublist Line Discounts 08-4	State 1: Set Workflow Field Values	3/13/2018 11:22 am	3/13/2018 11:22 am 1	Discount Threshold: 1000.00 Discount Percent: 15.0% Original Total: 4781.33

(1) Initial workflow field values are set.
(2) Eligible items are discounted and item names are stored in the Discounted Items (Workflow) field.
(3) The Discount Total (Workflow) field is calculated and a Send Email action is executed that sends the details of the discounts to the customer.

Consult the Workflow Execution Logs for each state for processing details.

14 Open the associated Customer record and select the **Communication** subtab, **Messages** sublist. You should see a new email listed with the subject "You received a discount!"

15 Click **View** to view the email's contents:



Here you can see the output from the two workflow fields entered during the configuration of the Send Email action in State 3.

08: Close Orders with Inactive Items (Optional)

Scenario: As part of their scheduled data maintenance processes, SuiteDreams would like to automatically close open sales orders with inactive line items. They would like to have this process execute every day at midnight with no end date.

In addition, SuiteDreams would also like to close open sales orders with line item amounts equal to $0.00. This often occurs when free promotional items are added to an order that require no specific fulfillment processing. As a result, these line items often keep orders open indefinitely, since an order cannot be closed until all line items have been fulfilled.

Create the Saved Search

1 Navigate to **Lists > Search > Saved Searches > New**, select the "Transaction" record type, and enter the following information:

SEARCH TITLE	Sales Orders with Inactive Items
ID	_sdr_ss_inactive_lines
PUBLIC	Checked
Criteria subtab (**Standard** sublist)	
FILTER	DESCRIPTION
Type	is Order
Billing Status	is Open
Main Line	is false
Item : Inactive	is true
Closed	is false
Results subtab	
SORT BY	Date
There is no special results sorting required. Adjust as desired.	

2 Select **Save & Run**. If no records are returned in the search results, perform ***either*** of the following steps:

- Enter 2-3 orders with a variety of items, making sure to include any of the following inactive items: *Modern Sofa*, *Sofa Repadding*, and/or *Furniture Repolishing*.

or…

- Edit a couple Item records and enable the INACTIVE option on the **System Information** subtab, then enter 2-3 orders with a variety of items, making sure to include the newly inactivated items above.

Note each of the order numbers created.

Create the Workflow

3 Create a new workflow by entering the following information (keeping all other fields at their defaults unless otherwise noted):

Basic Information	
NAME	Close Orders - Inactive Items
ID	_sdr_sub_close_inactive
RECORD TYPE	Transaction
SUB TYPES	Order
DESCRIPTION	Enter a description of the workflow.
RELEASE STATUS	Released
KEEP INSTANCE AND HISTORY	Always
ENABLE LOGGING	Checked
Initiation	
SCHEDULED	Checked
Schedule	
SAVED SEARCH FILTER	Sales Orders with Inactive Items
REPEAT	Checked
FREQUENCY	Daily
SCHEDULED FROM DATE	*<todays date>*
EXECUTION TIME	*12:00 am*

4 Save the new workflow definition.

5 Rename the default State 1 to "State 1: Close Inactive Lines".

Configure Actions, and Conditions

6 Create a Sublist Action Group and configure a single **Set Field Value** action that sets the CLOSED sublist line field to *true* (checked) for any inactive line item **or** any line item with a line amount equal to $0.00.

RECORD	FIELD *	COMPARE TYPE	VALUE
Item (Line)	Inactive	checked	
	Amount (Line)	equal	0.00
		equal	

Execute the Workflow and Test

7 Edit the workflow definition and click the **Execute Now** button. This button is only available for scheduled workflows and facilitates initial workflow testing.

8 View a couple of the orders above and examine **Workflow History**:

ENTRY	RESULT	DATE/TIME
Workflow initiated		**5/18/2020 - 09:57:15.170**
Running ONENTRY trigger under SCHEDULED (Event: SCHEDULE; Context: null)		**5/18/2020 - 09:57:15.172**
SUBLISTACTIONGROUP: Items	Executed	5/18/2020 - 09:57:15.174
Line #1		5/18/2020 - 09:57:15.176
SETFIELDVALUE	Considered	5/18/2020 - 09:57:15.182
Condition: Line : Item : Inactive = T Or Line : Amount = 0.00	False	5/18/2020 - 09:57:15.367
Line #2		5/18/2020 - 09:57:15.367
Line #3		5/18/2020 - 09:57:15.380
Line #4		5/18/2020 - 09:57:15.390
Line #5		5/18/2020 - 09:57:15.401
SETFIELDVALUE	Executed	5/18/2020 - 09:57:15.407
Condition: Line : Item : Inactive = T Or Line : Amount = 0.00	True	5/18/2020 - 09:57:15.412
isclosed = T		5/18/2020 - 09:57:15.418
Workflow finished		**5/18/2020 - 09:57:15.445**

9 Return to the **Items** sublist and verify the enablement of the CLOSED sublist line field for each inactive line item.

09: Create an Expense Line Approval Workflow (Optional)

Scenario: In this optional exercise, you will build two workflows to support the approval of expense lines in the **Expenses** sublist on the Expense Report record.

> **Note**: This optional exercise is considered a *challenge* exercise…meaning it does not provide step-by-step instruction on how to build each expense approval workflow. You will be provided with general guidance only.

Business Requirements

SuiteDreams has determined that delays in expense approvals and subsequent employee reimbursement have increased due to missing receipts on expense reports. Employees are often unaware when a receipt is required, resulting in unnecessary emails between manager and employee.

As a result, SuiteDreams wants to create a workflow that verifies that all required receipts have been provided prior to approval.

- Employees should be able to easily identify which expenses require receipts and which ones do not.
- Employees should be notified via email when submitting an expense report with missing receipts.
- Managers should be notified when there are employee expense reports waiting for their approval.
- Approvals (both manager and accounting) should only be possible when all receipts have been provided.

To meet these requirements, you will use Sublist Action Groups and custom sublist line fields to support expense line approvals using two workflows.

Create SuiteBuilder Components

In this series of steps, you will create the necessary SuiteBuilder components to support the Expense Report routing and approval process.

Create the Necessary Custom Lists

1 First, create the following two custom lists:

- Expense Approval Status (`_sdr_exp_appr_status`)
- Expense Report Type (`_sdr_exp_rep_type`)

Expense Approval Status should have the following list values:

- Auto-Approved
- Pending Manager Approval
- Pending Accounting Approval
- Pending Approval - Missing Receipts
- Manager Approved
- Accounting Approved
- Rejected

Report Type should have values similar to the following (feel free to modify as desired):

- Conference
- Training Event
- Customer On-Site
- Vendor On-Site
- Other

Now you can create the necessary custom fields and link them to each list.

Create a Custom Transaction Body Field

2. Create a custom transaction body field called "Report Type" (`_sdr_exp_rep_type`) and associate it with the custom Report Type list you just defined.

   Make sure the field is enabled for the EXPENSE REPORT record type and have it display in the **Main** section of the Expense Report form.

Custom Transaction Line Field

3. Create a custom transaction line field called "Expense Approval Status" (`_sdr_exp_appr_status`) and associate it with the custom Expense Approval Status list.

   Make sure the field is enabled for the EXPENSE REPORT record type and set its display type to "Disabled".

4. Create two more custom transaction line fields, both of type Check Box:

   - Receipt Required (`_sdr_exp_rec_required`)
   - Receipt Attached (`_sdr_exp_rec_attached`)

   These fields will allow the employee to see when an expense item requires a receipt and whether or not they have provided all required receipts.

Custom Transaction Form (Expense Report)

While you certainly could use the Standard Expense Report form, you will create a custom form that hides several transaction body and sublist fields that will not be needed to support the expense approval process.

5 Customize the Standard Expense Report form (**Customization > Forms > Transaction Forms**) to create a custom Expense Report form based on the configuration below:

NAME	Custom Expense Report
ID	_sdr_exp_rep_entry
STORE FORM WITH RECORD	Checked
FORM IS PREFERRED	Checked
Screen Fields subtab, **Main** sublist	
Hide the Supervisor Approval, Accounting Approval, and Next Approver fields.	
Set the DISPLAY TYPE of the Approval Status field to "Inline Text".	
Enable the Report Type field and move it to the top of the **Classification** field group.	

Expenses (1st sublist)
Hide the Use Multi-Currency and Exchange Rate fields.
Expenses (2nd sublist)
Hide the Ref No, Expense Account, Foreign Amount, and Exchange Rate fields.
Hide all fields from Department → Non-reimbursable.
Enable the Receipt Required, Receipt Attached, and Expense Approval Status fields.   **Note**: You may customize the form any way you see fit. However, you must add the three fields above and in the order they are listed.

6 Navigate to **Transactions > Employees > Enter Expense Reports** and examine the effects of the form customizations on the transaction body and **Expenses** sublist fields.

With these SuiteBuilder components complete, you can now create 2 workflows—a client-side workflow that further manipulates the Expense Report entry form and sets initial statuses and sublist field values and a server-side workflow to support the verification of receipts and the approval of expense report lines.

Create the Client-Side Workflow

During the entry of a new expense report, each line will be evaluated to determine whether the amount of each expense requires manager approval or accounting approval based on the following conditions:

- Sublist lines where amount <= 50.00 will be auto-approved
- Sublist lines where amount > 50.00 and <= 1000.00 will require manager approval
- Sublist lines where amount > 1000.00 will require accounting approval

7 Create a new workflow called "Expense Report Entry" on the *Transaction > Expense Report* record sub-type and configure the workflow's initiation parameters as follows:

- The workflow should execute on record load during the creation of new records and when viewing/editing existing Expense Report records.
- Set the workflow status to "Testing" and enable logging.

Configure State 1

8 Edit State 1 and rename it to "State: Manipulate Fields/Set Values" and configure it as a non-exiting state.

9 Configure the following actions on record load to further manipulate the display of transaction body and sublist line fields. No conditions are required.

Action	Configuration
Set Field Mandatory	Make the Category (Line) sublist field mandatory.
Set Field Display Label	Change the label of the Category (Line) sublist field to "Expense Type".
Set Field Display Label	Change the label of the Memo (Line) sublist field to "Expense Notes".
Set Field Display Type	Disable the Currency (Line) sublist field.
Set Field Display Type	Disable the Receipt Required (Line) sublist field.
Set Field Display Type	Disable the Expense Approval Status (Line) sublist field.

10 Configure the following actions on After Field Edit to evaluate and set the appropriate expense report statuses and sublist field values based on expense amount:

Action	Configuration
Set Field Value	Set Receipt Required = T when the expense line amount is greater than $50.
Set Field Value	Set Receipt Required = F when the expense line amount is less than or equal to $50.
Return User Error	Return the alert message below to the employee when an expense amount greater than $1000 is entered. "Expense line amounts greater than $1000 will require accounting approval prior to reimbursement."
Set Field Value	Set Currency (Line) equal to the expense report currency when selecting an Expense Type.
Set Field Value	Set Expense Approval Status (Line) to "Auto-Approved" when the expense line amount is less than or equal to $50.
Set Field Value	Set Expense Approval Status (Line) to "Pending Manager Approval" when the expense line amount is greater than $50 and less than or equal to $1000.
Set Field Value	Set Expense Approval Status (Line) to "Pending Accounting Approval" when the expense line amount is greater than $1000.
Set Field Value	Set Expense Approval Status (Line) to "Pending Approval - Missing Receipts" when an expense line item requires a receipt and the receipt is not attached.
Set Field Value	Set Expense Approval Status (Line) to "Pending Manager Approval" on the update of the Receipt Attached (Line) field when: ➢ Expense Approval Status (Line) equal to "Pending Approval - Missing Receipts" ➢ All required receipts have a receipt attached ➢ Expense line amount is greater than $50 ➢ Expense line amount is less than or equal to $1000
Set Field Value	Set Expense Approval Status (Line) to "Pending Accounting Approval" on the update of the Receipt Attached (Line) field when: ➢ Expense Approval Status (Line) equal to "Pending Approval - Missing Receipts" ➢ All required receipts have a receipt attached ➢ Expense line amount is greater than $1000

Action	Configuration
Set Field Value	Set the Expense Approval Status (Line) to "Pending Approval - Missing Receipts" on the update of the Receipt Attached (Line) field when: ☑ USE EXPRESSIONS Receipt Required = T AND Receipt Attached = F AND (Expense Approval Status (Line) = Pending Manager Approval OR Expense Approval Status (Line) = Pending Accounting Approval)

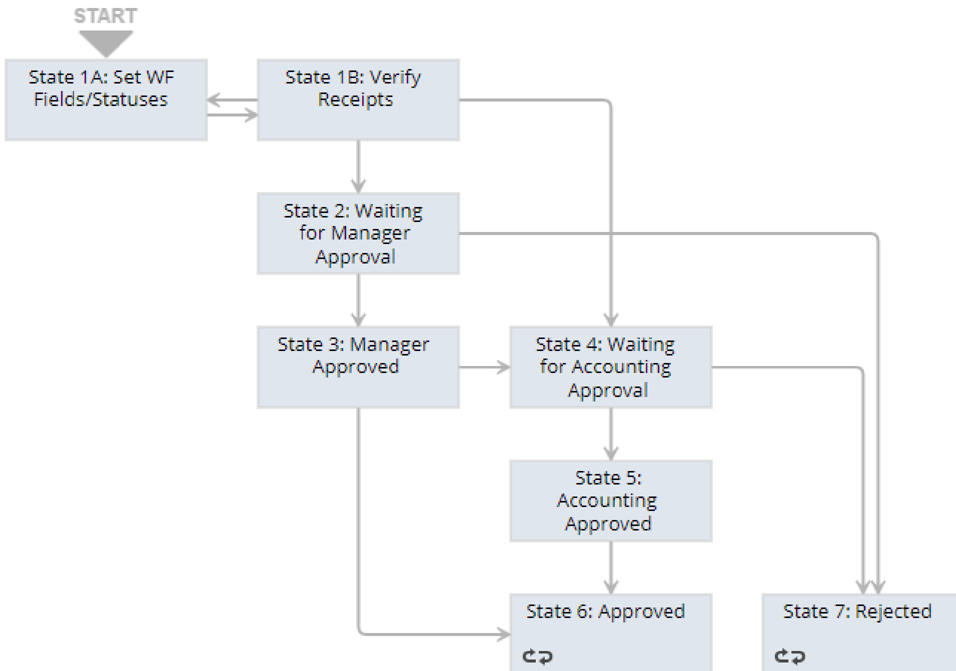
Create the Server-side Workflow

Now create a server-side workflow that verifies that all required receipts have been provided and routes the record to the appropriate approvers based on the statuses set in the Entry workflow.

11 Create a workflow called "Expense Line Approval" on the *Transaction > Expense Report* record sub-type and configure the workflow's initiation parameters as follows.

- The workflow should execute on record save during the creation of new Expense Report records only.
- Set the workflow status to "Testing" and enable logging.

12 Create, rename, and arrange the remaining states and transitions as shown below:



Configure the Necessary Workflow Fields

13 To identify which sublist lines require manager and/or accounting approval and which sublist lines are auto-approved, create the following Check Box workflow fields:

- Manager Approval Required (`_sdr_exp_mgr_appr_req`)
- Accounting Approval Required (`_sdr_exp_acct_appr_req`)
- Receipts Provided (`_sdr_exp_rec_provided`)

14 Now create the following Currency workflow fields to store approval threshold values:

- Auto-Approval Threshold (`_sdr_exp_auto_appr_limit`) = 50.00
- Manager Approval Threshold (`_sdr_exp_mgr_appr_limit`) = 50.00
- Accounting Approval Threshold (`_sdr_exp_acct_appr_limit`) = 1000.00

15 Finally, create the following Free-Form Text workflow fields to store missing receipts:

- Missing Receipts (`_sdr_exp_missing_rec`)
- Missing Receipts - Saved (`_sdr_exp_missing_rec_saved`)

Configure State 1A

16 Add a Sublist Action Group on the **Expenses** sublist. No group conditions are necessary.

17 Within the new Sublist Action Group, configure two Set Field Value actions that set the appropriate Expense Approval Status based on the amount of each line.

Set Field Value action #1:

- Enable the Manager Approval Required field when expense line amount is greater than the Manager Approval Threshold.

Set Field Value action #2:

- Enable the Accounting Approval Required field when expense line amount is greater than the Accounting Approval Threshold.

Configure State 1B

State 1B evaluates whether all required receipts have been attached. Expense reports cannot be approved until all required receipts have been provided by the employee.

18 Add a Sublist Action Group (**Expenses** sublist) on Before Record Submit and configure the following three Set Field Value actions that set the Missing Receipts and Receipts Provided workflow fields.

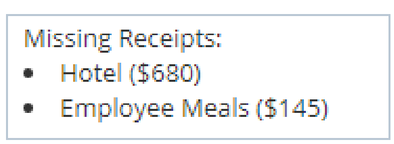
Set Field Value action #1:

- Set the Missing Receipts workflow field with each expense item that has a missing receipt for any expense line with an Expense Approval Status of "Pending Approval - Missing Receipts".

Using a formula:

```
{custworkflow_sdr_exp_missing_rec} || '<li>' || {line.category}
|| ' ($' || {line.amount} || ')' || '</li>'
```

This generates the following when viewed in Workflow History:



Set Field Value action #2:

- Disable the Receipts Provided workflow field when an expense line item requiring a receipt does not have a receipt attached.

Set Field Value action #3:

- Enable the Receipts Provided workflow field when an expense line item requiring a receipt has a receipt attached.

19 Configure a state-level Set Field Value action (also on Before Record Submit) that sets the Missing Receipts - Saved to the current value of the Missing Receipts when all receipts have been provided and Missing Receipts is not empty and not equal to "None".

20 Configure another state-level Set Field Value action that sets the Missing Receipts workflow field to "None" when all receipts have been provided.

> The order of the Sublist Action Group, its actions, and each state-level action in State 1B is critical to the workflow's verification of required receipts.

Now let's notify the employee and manager accordingly.

21 Configure two state-level Send Email actions (on After Record Submit)—one that notifies the employee that their expense report contains missing receipts and another that notifies the manager when they have an expense report to approve.

Send Email action #1 to employee when receipts have not been provided:

SUBJECT	Expense report (#{tranid}) has missing receipts
BODY	Your recent expense report (#{tranid}) contains expense items with missing receipts. To ensure prompt approval and reimbursement, please update your expense report to include all missing receipts. {custworkflow_sdr_exp_missing_rec_saved}

Send Email action #2 to manager when receipts have been provided:

SUBJECT	Expense report (#{tranid}) is waiting for your approval
BODY	One of your direct reports ({entity.employee}) submitted Expense Report #{tranid} and it is now waiting for your approval.

22 Configure a transition from State 1A to State 1B on Before Record Submit.

This ensures the workflow stays in a Before Record Submit context and does not transition to an After Record Submit context before re-evaluating receipts.

23 Configure a transition from State 1B back to State 1A on After Record Submit and only when all receipts have been provided.

This ensures the record remains in State 1A until all receipts have been provided.

24 Configure a transition from State 1B to State 2 when all receipts have been provided and manager approval is required.

25 Configure a transition from State 1B to State 4 when all receipts have been provided, manager approval is not required, and accounting approval is required.

Configure State 2

Now configure the necessary actions and transitions in State 3 to support manager approval of employee expenses.

26 Add a standard Workflow Action Group (*Group 1*) and configure two Add Button actions—one labeled "Approve" and the other labeled "Reject".

Configure the group condition so the buttons only display when the user's role is "Employee Manager", manager approval is required, and all receipts have been provided.

27 Configure a state-level Lock Record action that prevents all users except those with the following roles from modifying the record when all receipts have been provided:

- Accounting Manager
- Administrator
- Employee Manager

28 Re-order the action so the Lock Record action is executed before the button group.

29 Configure two transitions, from State 2 to State 3 and State 7, based on each button click.

Configure State 3

30 Add a Sublist Action Group (**Expenses** sublist) and configure a single Set Field Value action that updates each expense line status to "Manager Approved" for all lines currently in a status of "Pending Manager Approval".

Note: This may seem overly simplistic…manager clicks **Approve** and the status is updated to "Manager Approved". But remember how you got here…

All the logic that controls manager approval is configured in the transition from State 1B to State 2, which only executes when all receipts have been provided <u>and</u> manager approval is required. So, in State 3 of the workflow, the only remaining thing to do is set the Expense Approval Status of each expense item to "Manager Approved" which are currently in a "Pending Manager Approval" status.

31 Add a state-level Set Field Value action that sets transaction body the APPROVAL STATUS field to "Approved" when accounting approval is not required.

32 Configure a transitions from State 3 to:

- State 4, when accounting approval is required and all receipts have been provided.
- State 6, when accounting approval is not required.

Configure State 4

33 Add a standard Workflow Action Group (*Group 1*) and configure two Add Button actions—one labeled "Approve" and the other labeled "Reject".

Configure the group condition so the buttons only display when the user's role is "Accounting Manager", accounting approval is required, and all receipts have been provided.

34 Configure a transitions from State 4 to State 5 and State 7, based on each button click.

Configure State 5

35 Add a Sublist Action Group (**Expenses** sublist).

36 Within the new Sublist Action Group, configure the following Set Field Value action that updates each expense line status to "Accounting Approved" for all expense lines currently in a status of "Pending Accounting Approval".

Just as with manager approval, all of the logic that controls accounting approval is configured in the transition from State 3 to State 4, which only executes when all receipts have been provided <u>and</u> accounting approval is required. So, in State 5 of the workflow, the only remaining thing to do is set the Expense Approval Status of each expense item to "Accounting Approved" that are currently in a "Pending Accounting Approval" status.

37 Verify there is a transition from State 5 to State 6. No further configuration is necessary.

Configure State 6

38 Configure a state-level Set Field Value action on entry to the state that sets APPROVAL STATUS to "Approved".

39 (**Optional**) Configure a Lock Record action for all users <u>except</u> those with the Accounting Manager user role.

Configure State 7

40 Configure a state-level Set Field Value action on entry to the state that sets APPROVAL STATUS to "Rejected".

41 (**Optional**) Configure a Lock Record action for all users <u>except</u> those with the Accounting Manager user role.

Configure Employee Users and Roles

Create Accounting Manager Employee

42 Create a new Employee record for the "Accounting Manager" as follows:

NAME	Accounting Manager
SUBSIDIARY	HEADQUARTERS

43 Save the Employee record. You will configure the user's login access in the next series of steps.

Configure Login Access

Grant login access to Fiona Small (Employee Manager) and the Accounting Manager (Accounting Manager).

44 Navigate to **Setup > Custom > Configure Login Access**.

45 Select "Fiona Small" in the EMPLOYEES list on the left side of the page.

46 Select the "Employee Manager" role in the ROLES list on the right side of the page.

47 Click the **Generate Employee Access** button and note the generated login email.

fsmall@TSTDRVnnnnnn.com (password: training1)

48 Select "Accounting Manager" in the EMPLOYEES list on the left side of the page.

49 Select the "Accounting Manager" role in the ROLES list on the right side of the page.

50 Click the **Generate Employee Access** button and note the generated login email.

amanager@TSTDRVnnnnnn.com (password: training1)

51 (**Optional**) Edit the Accounting Manager's Employee record and change the auto-generated email address to *acct-mgr@TSTDRVnnnnnn.com*.

Execute the Workflows and Test

Use Case #1

52 Enter a new expense report (**Transactions > Employees > Enter Expense Reports**) that resembles the following:

DATE *	EXPENSE TYPE *	CURRENCY *	AMOUNT *	RECEIPT REQUIRED	RECEIPT ATTACHED	EXPENSE APPROVAL STATUS
10/5/2020	Airfare	US Dollars	1,027.00	Yes	Yes	Pending Accounting Approval
10/5/2020	Hotel	US Dollars	846.00	Yes	Yes	Pending Manager Approval
10/5/2020	Employee Meals	US Dollars	129.00	Yes	Yes	Pending Manager Approval
10/5/2020	Taxi/Ride Share	US Dollars	67.00	Yes	Yes	Pending Manager Approval
10/5/2020	Taxi/Ride Share	US Dollars	48.00			Auto-Approved

To make workflow testing a bit easier, select one of the employees below, each of whom have Fiona Small as their manager and an email address on their Employee record:

- Aubrey Pober
- Baruch Rollins
- Caitlin McGyver
- Edwin Goldwasser
- Kim Huffman
- Sara Torrance

As you enter each expense line, note the updates to transaction body and sublist fields based on the Set Field Mandatory, Set Field Display Label, and Set Field Display Type actions configured on the Before Record Load trigger in the Entry workflow.

Also note the actions executing on the After Field Edit trigger based on the Amount (Line) field and whether or not the Receipt Attached (Line) field is enabled. Also note the auto-approval of the expense item less than $50.

53 Save the Expense Report record and note the APPROVAL STATUS of "Pending Approval".

54 **In a different browser**, login as employee manager, Fiona Small (*fsmall@TSTDRVxxxxxxx.com*) and click on the View Expense Reports link in the Expense Reports portlet.

55 Click View to open the new expense report and note the addition of the **Approve** and **Reject** buttons.

56 Click **Approve**. The status remains "Pending Approval" and each expense line item that was waiting for manager approval now shows an Expense Approval Status of "Manager Approved".

57 Logout as Fiona Small and login as Accounting Manager (*acct-mgr@TSTDRVxxxxxxx.com*).

58 Click on the Expense Reports link in the Shortcuts portlet.

59 Click View to open the new expense report and note the addition of the **Approve** and **Reject** buttons.

60 Click **Approve**. The status is updated to "Approved" and each expense line item that was waiting for accounting approval now shows an Expense Approval Status of "Accounting Approved".

61 (**Optional**) Create another expense report and test rejection processing.

Use Case #2

For this use case, you will perform many of the same steps as in the first use case, except you will verify the receipt verification process and its effect on approvals.

62 Enter a new expense report (**Transactions > Employees > Enter Expense Reports**) that resembles the following:

DATE *	EXPENSE TYPE *	CURRENCY *	AMOUNT *	RECEIPT REQUIRED	RECEIPT ATTACHED	EXPENSE APPROVAL STATUS
10/5/2020	Airfare	US Dollars	1,027.00	Yes	Yes	Pending Accounting Approval
10/5/2020	Hotel	US Dollars	846.00	Yes	Yes	Pending Manager Approval
10/5/2020	Employee Meals	US Dollars	129.00	Yes		Pending Approval - Missing Receipts
10/5/2020	Taxi/Ride Share	US Dollars	67.00	Yes		Pending Approval - Missing Receipts
10/5/2020	Taxi/Ride Share	US Dollars	48.00			Auto-Approved

Note that two of the expense items require receipts but they have not been attached to the expense report.

63 Save the Expense Report record.

64 Select the **System Information** subtab, **Active Workflow** sublist.

Both workflows are currently executing on this record. The Approval workflow is waiting for manager approval in State 2 and the name of each expense item missing a required receipt has been stored in the Missing Receipts workflow field.

65 Select the **Workflow History** sublist and review the Workflow Execution Logs associated with each state.

66 **In a different browser**, login as employee manager, Fiona Small (*fsmall@TSTDRVxxxxxxx.com*) and click on the View Expense Reports link in the Expense Reports portlet.

67 Click View to open the new expense report. Do you see the **Approve** and **Reject** buttons?

You should not see them, since all required receipts have not been provided (Receipts Provided (Workflow) = F).

68 **Return to the other browser** and edit the Expense Report record.

69 Enable the RECEIPT ATTACHED sublist field for all required receipts and re-save the record.

70 Select the **System Information** subtab, **Workflow History** sublist.

- Note the transitions between State 1A and State 1B and back to State 1A during edit of the record.
- Receipts Provided (Workflow) = Yes (T) upon saving the record with all receipts attached.
- Missing Receipts - Saved (Workflow) is set to the value stored in Missing Receipts (Workflow).
- Missing Receipts (Workflow) is set to "None".

71 Return to the browser where you are logged in as Fiona Small and refresh the list of Expense Reports.

72 Click View to open the new expense report. Do you see the **Approve** and **Reject** buttons now?

You should see them, since all required receipts have been provided (Receipts Provided (Workflow) = T).

73 Click **Approve**. The status remains "Pending Approval" and each expense line item that was waiting for manager approval now shows an Expense Approval Status of "Manager Approved".

74 Logout as Fiona Small and log back in as Accounting Manager (*acct-mgr@TSTDRVxxxxxxx.com*).

75 Click on the Expense Reports link in the Shortcuts portlet.

76 Click View to open the new expense report and note the addition of the **Approve** and **Reject** buttons.

77 Click **Approve**. The status is updated to "Approved" and each expense line item that was waiting for accounting approval now shows an Expense Approval Status of "Accounting Approved".

78 Examine the Workflow Execution Log for each state of the Approval workflow to verify approval routing and processing events.

Use Case #3

In this use case, you will create an expense report that contains expense items requiring manager approval only. Accounting approval will be skipped and final approval will be automatically granted, since there is nothing for accounting to approve.

79 Enter a new expense report (**Transactions > Employees > Enter Expense Reports**) that resembles the following:

DATE *	EXPENSE TYPE *	CURRENCY *	AMOUNT *	RECEIPT REQUIRED	RECEIPT ATTACHED	EXPENSE APPROVAL STATUS
10/5/2020	Hotel	US Dollars	846.00	Yes	Yes	Pending Manager Approval
10/5/2020	Employee Meals	US Dollars	129.00	Yes	Yes	Pending Manager Approval
10/5/2020	Taxi/Ride Share	US Dollars	67.00	Yes	Yes	Pending Manager Approval
10/5/2020	Taxi/Ride Share	US Dollars	48.00			Auto-Approved

80 Save the Expense Report record and note the APPROVAL STATUS of "Pending Approval".

81 **In a different browser**, login as employee manager, Fiona Small (*fsmall@TSTDRVxxxxxxx.com*) and click on the View Expense Reports link in the Expense Reports portlet.

82 Click View to open the new expense report and note the addition of the **Approve** and **Reject** buttons.

83 Click **Approve**. The status is immediately updated to "Approval" and each expense line item that was waiting for manager approval now shows an Expense Approval Status of "Manager Approved".

84 Examine the Workflow Execution Log for each state of the Approval workflow to verify approval routing and processing events.

Use Case #4

Finally, create an expense report that contains expense items requiring accounting approval only. Manager approval will be skipped and final approval will be automatically granted, since there is nothing for the manager to approve.

85 Enter a new expense report (**Transactions > Employees > Enter Expense Reports**) that resembles the following:

DATE*	EXPENSE TYPE*	CURRENCY*	AMOUNT*	RECEIPT REQUIRED	RECEIPT ATTACHED	EXPENSE APPROVAL STATUS
10/5/2020	Airfare	US Dollars	1,027.00	Yes	Yes	Pending Accounting Approval
10/5/2020	Hotel	US Dollars	1,006.00	Yes	Yes	Pending Accounting Approval
10/5/2020	Taxi/Ride Share	US Dollars	48.00			Auto-Approved

86 Save the Expense Report record and note the APPROVAL STATUS of "Pending Approval".

87 **In a different browser**, login as employee manager, Fiona Small (*fsmall@TSTDRVxxxxxxx.com*) and click on the View Expense Reports link in the Expense Reports portlet.

88 Click View to open the new expense report.

89 Notice that there are no **Approve** and **Reject** buttons.

Note: If you look in **Workflow History**, you will see that the record is currently in State 4: Waiting for Accounting Approval, but you are not logged in as a user with the Accounting Manager user role. Therefore, no buttons are displayed.

90 Logout as Fiona Small and log back in as Accounting Manager (*acct-mgr@TSTDRVxxxxxxx.com*).

91 Click on the Expense Reports link in the Shortcuts portlet.

92 Click View to open the new expense report. Voila…buttons!

93 Click **Approve**. The status is updated to "Approved" and each expense line item that was waiting for accounting approval now shows an Expense Approval Status of "Accounting Approved".

94 Examine the Workflow Execution Log for each state of the Approval workflow to verify approval routing and processing events.

EXERCISE SOLUTIONS

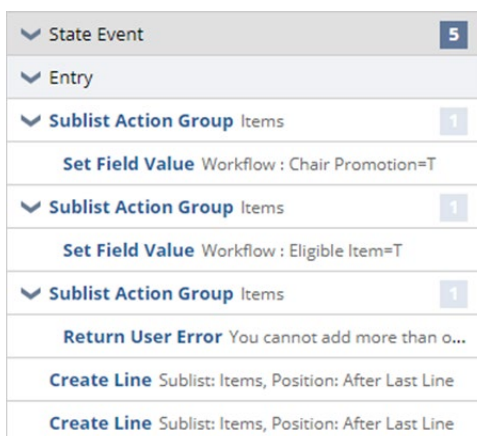
Having trouble with this module's workflows?

1. Navigate to **Customization > Workflow > Workflows**.
2. Change the RECORD TYPE filter to "Transaction".
3. Click **Edit** next to the workflow you would like to analyze:
   - SOL Update Sublist Lines 10 (*Exercises 01*)
   - SOL Update Sublist Lines 10-2 (*Exercise 02*)
   - SOL Update Sublist Lines 10-3 (*Exercise 03*)
   - SOL Update Sublist Lines 10-4 (*Exercise 04*)
   - SOL Sublist Line Approval 10-5 (*Exercise 05*)
   - SOL Line Item Entry Validations 10-6 (*Exercise 06*)
   - SOL Sublist Line Discount 10-7 (*Optional Exercise 07*)
   - SOL Close Orders - Inactive Items 10-8 (*Optional Exercise 08*)

Note: There are no solution workflows for challenge *Optional Exercise 09*.

02: Create an Items Sublist Line

9. Order of actions in State 2: Add Promo Item(s):



03: Apply Sublist Line Discounts

7. Configure a **Set Field Value** action within the group that applies a 15% discount on the sublist RATE field using a formula. Configure a formulaic condition on the action that only applies the discount to line items with a rate greater than $1500.

Rate (Line) = `{line.rate} - {line.rate} * .15`

Condition: `{line.rate} > 1500.00`

04: Create a New Record based on Sublist Field Values

5 Configure a *formulaic condition* that restricts the creation of the Phone Call record to only when the order total is greater than the customer's credit limit less their overdue balance **and** only for the first line item added to the order.

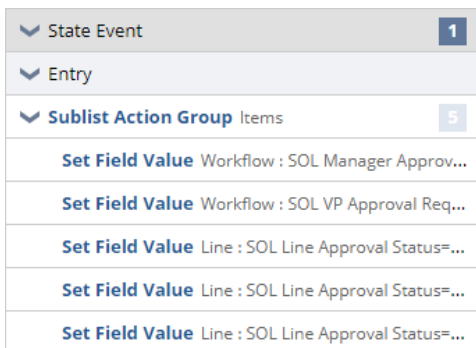
```
{total} > {entity.creditlimit} - {entity.overduebalance} AND {line.line} = 1
```

7 Select the appropriate trigger on the transition from State 3 to State 4 to support the Create Record action.

TRANSITION ON = After Record Submit

05: Create a Sublist Line Approval Workflow

11 Order of Set Field Value actions in State 1:



15 Do you remember what the preferred server trigger is for the Workflow Action Group containing the button actions?

Answer: Before Record Load

06: Perform Line Validations during Data Entry

No exercise solutions. Refer to solution workflow.

07: Automate Line Discounts using Formulas (Optional)

6 Create a Sublist Action Group and configure two **Set Field Value** actions as follows:

Construct a formula that applies a 15% discount to the RATE field for any item with a RATE greater than the Discount Threshold:

Rate (Line) = `{line.rate} - ({line.rate} * {custworkflow_sdr_sub_disc_percent})`
Condition: Rate (Line) > Discount Threshold (Workflow)

Construct a formula that populates the Discounted Items workflow field with any line item with a RATE greater than the Discount Threshold:

Discounted Items (Workflow) = `{custworkflow_sdr_sub_disc_items} || '<li>' || {line.item} || '</li>'`
Condition: Rate (Line) > Discount Threshold (Workflow)

9 Add a **Set Field Value** action that sets the Discount Total (Workflow) to the original total minus total.

Discount Total (Workflow) = `{custworkflow_sdr_sub_orig_total} – {total}`

08: Close Orders with Inactive Items (Optional)

No exercise solutions. Refer to solution workflow.