Action Buttons

Action Button fields create buttons that run actions, which allow users to run all kinds of possible functions on command while performing normal record workflows. For example, you can use action buttons to send emails, update fields or linked records, print documents from a template, and more.

See Actions for more information on the available action types.

> **Example**
>
> The out-of-the-box system includes many action buttons, some of which are described below. Action buttons can:
>
> - Activate document conversion to MS Word format on an attached file field containing PDFs.
> - Create contract approvals. This can include a complex set of If-Then-Else action conditions that embed several conditional actions.
> - Convert a ticket into a service request using a Data Conversion action.
> - Create Adobe Sign or DocuSign envelopes for electronic signature.

Create an Action Button

You can access the Actions wizard in several ways, but the easiest way is to select Setup [Table] from the table where you want to create the action.

1. From the top nav bar, expand the table's drop-down and select Setup [Table].
2. Select the Actions tab in the Table wizard.
3. On the General tab of the wizard, give the button a name and label.

   > It's often a good idea to use a "Button:" prefix when labeling action buttons. This makes it clear which fields are buttons when managing layouts. It also groups all the buttons next to one another on the Fields tab of the Table wizard, which helps to organize your fields.

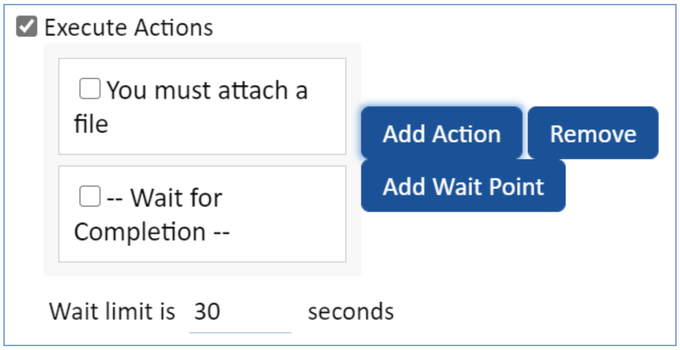
4. Select how the action button is displayed. The chosen display appears in both the record form and the table view. Action buttons in the action bar are always displayed as plain text links. Three display types are available:
   - **Button with text**: This shows a button labeled with the text you enter. For buttons, also select whether the button is primary or secondary, which determines the visual style of the button. Button styles can be configured in the Look and Feel on the Forms tab. You can also configure the style of navigation buttons, such as Save, Cancel, or Next.

    - **Image hyperlink**: This replaces the button with an uploaded image. These images are the same ones available in the Graphics tab of the Table wizard. Use a new image by clicking **Select/manage > Upload**.
    - **Text hyperlink**: This displays a text hyperlink using the text you enter. Sometimes using a text hyperlink can be useful to align the action button with other fields, save space, or better match the style of the form.

6. Define the system behavior before executing the action(s). You can choose to do nothing or save the record. If the button is clicked while editing the record, it's usually best to save. Otherwise, saving is unnecessary. See Saving Before Actions for more information about saving the record before executing the action(s).
7. Select the Execute Actions check box and click Add Action. The Actions screen opens.
8. Create a new action, or select an existing action and click Finish.

> ⚠ If you select multiple actions, they will be added in alphabetical order.

9. If desired, click Add Wait Point to add wait points between actions. Wait points force the system to wait for the preceding action(s) to complete before running the next action or moving to the final step defined in the "After executing actions" section. For example, if one of the actions brings up a window for manual conversion, and the following action makes use of the results of that conversion, adding a wait point between the actions can be useful. When you add a wait point, they are automatically added after the last action in the list, but they can be placed after any of the actions in the button sequence by dragging the wait point to the desired location.



☑ Execute Actions

□ You must attach a file

□ -- Wait for Completion --

Wait limit is 30 seconds

Wait point in action button

> ⚠ If you add wait points, you can also define the time limit in seconds to wait before the next action runs. With multiple wait points, they all use the same time limit.

10. Select whether the action button executes any additional tasks, such as adding a note, opening a URL, or others. See Additional Tasks for more information.
11. If the button runs a series of actions that take some time, and you don't need to show the user the end result, select the Run in Background checkbox to allow the actions to run without forcing the user to wait.
12. Define the system behavior after the action button executes its actions. You have a few options, including doing nothing, closing the record without saving, navigating within the record to a specified tab, or saving the record. If the button is clicked while editing the record, it's usually best to save. If you choose to save the record, see Saving After Actions for more information.

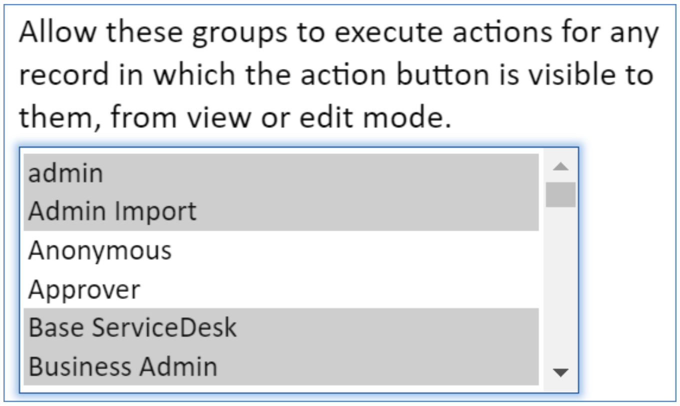
13. Enter Admin Notes describing the purpose of the button. Admin Notes are especially important for action buttons so that other administrators know the purpose of the button.

Options Tab

1. On the Options tab, select whether the field is deletable.
2. If desired, add one or more conditions to enable conditional visibility. Conditional visibility is often used with action buttons to control when they can be clicked, and it persists in table views. If the record doesn't meet the conditions for the button to appear, it doesn't appear in the table view, even if it has been added.
3. Skip the section to add conditional editing. Action buttons are not edited, so this section has no effect.

Permissions Tab

1. On the Permissions tab, select which groups are able to view the action button for each regular permission. If a group can view an action button, they can usually execute it. Action buttons cannot be edited, so you can ignore the edit permissions.
2. Configure the special "Allow these groups to execute actions..." permission. This is a unique setting that allows a group to see and therefore execute an action button on a record if they only have view access to the record. If this is active for a group, the user does not need to have edit permissions or record creator status for a record to execute the action button.



Allow these groups to execute actions for any record in which the action button is visible to them, from view or edit mode.

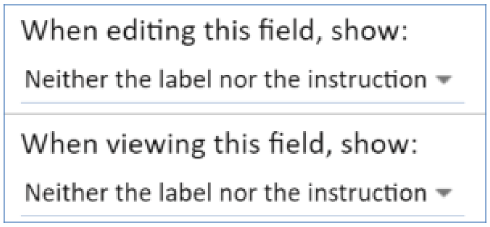
admin
Admin Import
Anonymous
Approver
Base ServiceDesk
Business Admin

Group permissions

> ⚠ Permissions are an important consideration when you create action buttons. For more information on how action buttons work with permissions, see Action Button Permissions.

Display Tab

1. On the Display tab, enter any pop-up text or input instructions.
2. Select whether to show the field label or input instructions. Usually, it's best to use the button text to give the user the necessary information to understand the button's actions. The most common exception is when an image hyperlink is used for the button without any text.

   When editing this field, show:
   Neither the label nor the instruction ▾

   When viewing this field, show:
   Neither the label nor the instruction ▾

   Label and instruction settings
3. If you choose to display the field label or input instructions, select their placement relative to the action button.
4. Click Finish to save the action button.

Saving Before and After Actions

When creating action buttons, the General tab of the Field wizard allows you to define when records are saved with action buttons and which operations are triggered. Triggering operations out of order after saving a record can cause unintended consequences, such as rules running on linked records before the user has finished updating a record.

Saving Before Actions

It's often a good idea to save the record first before clicking any actions buttons. Some changes made in the current editing session are only available to the actions in the action button if the record is saved before executing the actions. Although most field changes are stored in system memory and can be used by actions without saving first, some data types are not. Make sure to test the behavior of the button if you're choosing not to save the record.

> ⚠ If the action button is clicked from any context other than while editing a record, there is no need to save before executing the action(s). Even if the field option says that it saves before executing, the system does not save the record. Likewise, there is no need to save afterwards, unless you want to open the record after running the action(s). For more information on the contexts from which action buttons can be clicked, see Action Button Placement.

Saving After Actions

If the action button makes changes to the record during the editing process, it is best to save afterwards to confirm that any changes made by the button are saved to the record. Otherwise, if a user decides to cancel out of the record or fails to save the record at a later time, any changes made by the action button may be lost.

When you're saving the record after actions are executed, you can choose whether the save should close the record, open it again in edit mode, or open it in view mode.

- If you save it and open it again for viewing or editing, you can also define which record tab it should open to for power users and end users. This can help guide the workflow in cases where the record has a large number of tabs.
- If you save it and open it again for editing, you can choose whether to validate required fields. If you deselect this option, it acts similarly to an autosave.

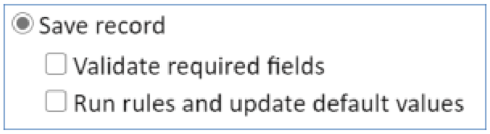
If you choose not to save the record, you can either choose to do nothing, close the record entirely, or navigate within the record to a specified tab and position.

Validate Required Fields and Run Rules

By default, record saving triggers the following operations, whether you save before or after actions run:

- The system checks to make sure that all mandatory fields have a valid entry.
- Any rules that run when a record is edited are triggered.
- The record default values are updated, such as the date modified.

You can disable these operations in the action button by clearing the "Validate required fields" and "Run rules and update default values" checkboxes.



● Save record
□ Validate required fields
□ Run rules and update default values

Save record options

> ⚠ Exercise caution when deselecting these options because there is a possibility for records to be canceled in the middle of being created, placed in a state where mandatory fields are incomplete, or prevented from being acted upon by normal table rules.

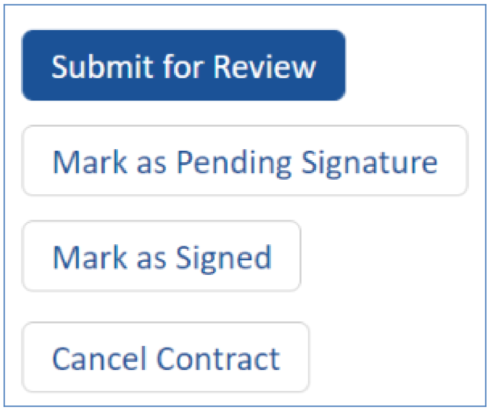
The "Run rules" option is particularly useful in cases where a record contains a large number of tabs. In these cases, a user may want to continue working on the record but use the action button to save the current state and run the associated action(s), triggering any edit rules.

The "Validate required fields" option is helpful in cases where records have one or more required fields. When you save after actions, this option is only available when choosing to also open the record for editing, which helps prevent the record from being closed in a state where mandatory fields are incomplete.

Action Button Placement

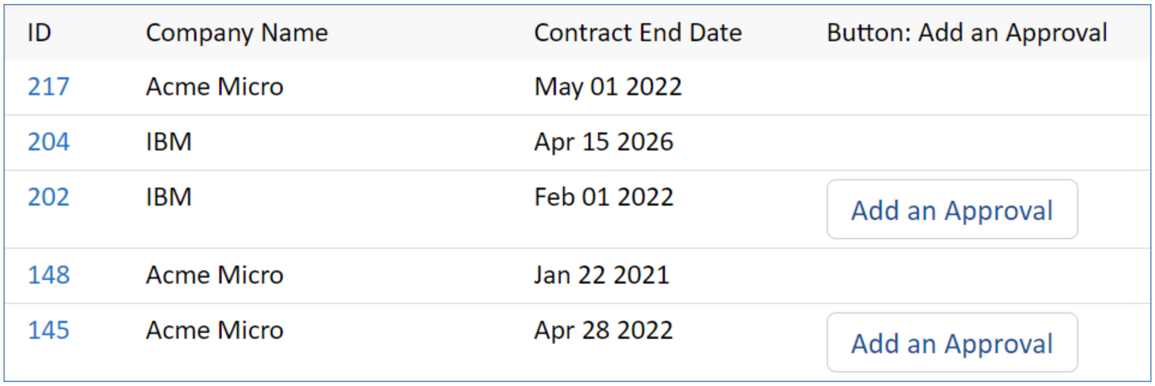
Action buttons can be placed in the following areas of the system:

- On the record layout, where the actions affect the current record. They can be activated in either view or edit mode of the record.



Action buttons in Contract record

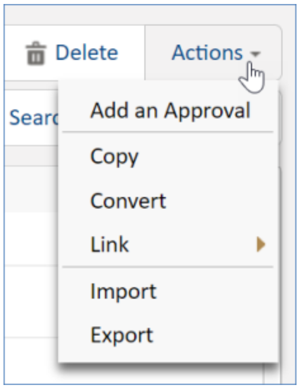
- In Table Views, where action buttons affect the record they are linked to.



ID	Company Name	Contract End Date	Button: Add an Approval
217	Acme Micro	May 01 2022	
204	IBM	Apr 15 2026	
202	IBM	Feb 01 2022	Add an Approval
148	Acme Micro	Jan 22 2021	
145	Acme Micro	Apr 28 2022	Add an Approval

Add an Approval button in table view

- In the table action bar, where they appear as hyperlinks. These actions affect all selected records in the table view.

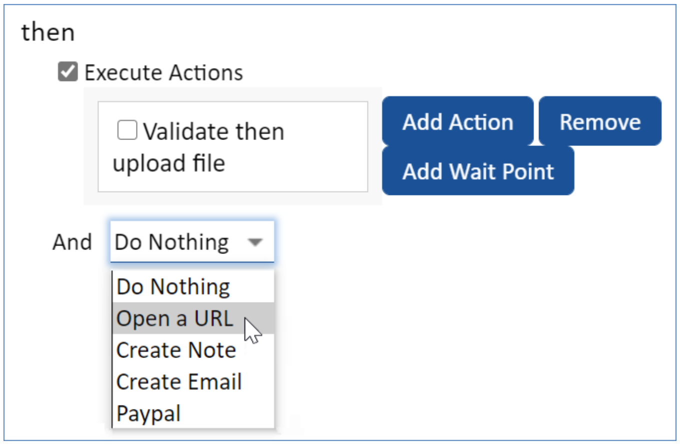


Actions drop-down menu

Because there are so many paths users can take to run actions, it's crucial to test every method. Consider these paths when deciding when the action button saves the record, and which actions are executed.

Additional Tasks

Aside from the available action types, there are a number of tasks that you can run with action buttons. To add a task, select it from the And drop-down menu on the General tab of the Field wizard for action buttons.



then

☑ Execute Actions

□ Validate then upload file

And Do Nothing

And drop-down list below the Execute Actions option on the General tab

Available tasks include:

- Running an ActiveX script
- Opening a URL
- Making a Skype call
- Opening the Working with Email wizard to create an email
- Sending a PayPal payment

Opening a URL

If you select Open a URL, the action buttons opens a window at the specified URL. The URL can be provided manually, by fields in a record, or with a Script action. If the action button opens a URL defined by a Script action, the last script in the list of execute actions may use `EWset::setRedirect` to define the URL.

If you enter the URL manually, the URL must use special syntax and, in some cases, a formula.

- If you want to open a static URL, enter your URL enclosed in double quotes, such as "https://www.agiloft.com" or "https://www.customurl.com."
- If you want to open a URL where the main address depends on record data, you can concatenate the parts of the URL. Enclose the static parts of the URL in double quotes, enter the field names prefixed with a "$" sign, and combine each part with a plus sign. For example, if your record contains a field named Company, you could specify a URL like "http://www." + $company + ".com" to go to the company site.
- If you want to open a URL where parameters that occur after the main address depend on record data, concatenate the parts of the URL and use the `urlEncode()` function. Enclose the static parts of the URL in double quotes, enclose the field names prefixed with a "$" in `urlEncode()`, and place a plus sign between the different parts of the URL.
    - URL parameters are easy to identify because they come after an equals (=) sign. For example, if you have a URL such as "http://www.agiloft.com?login=myLogin&password=myPassword," then "login" and "password" are URL parameters, with "myLogin" and "myPassword" being the parameter values.
    - To compose a URL where parameters occur after the main address and the URL uses fields from the current record, you could use a URL like "http://www.agiloft.com?login=" + urlEncode($myLogin) + "&password=" + urlEncode($myPassword).

> ⚠ Every web server detects parameters and their values in a URL string by locating "?," "&," and "=" characters. That means that a URL parameter value should not itself contain any of these characters, otherwise the web server cannot recognize the correct parameter value. To avoid this confusion, each parameter that uses a field variable with "$" should be URL-encoded so that offending characters in each parameter value are replaced with "safe" character sequences. The server understands these sequences and converts them back to source characters when reading the parameters string.
>
> If you add variables through the Formula wizard, variables are added with `urlEncode()` function for you. Always use this function if you are using the field or variable in the parameter part of a URL, such as "www.b2b.com?param=" + urlEncode($field). Do not use this function when your field or variable is used in the main address part of the URL, such as "http://www ." + $company + ".com."

Action Button Permissions

Action buttons have admin-level permissions by default and operate independently of a user's permissions. This makes it important to control who has view access to an action button field, as well as at what stage in the record workflow the action button should appear. If a user can view the button, they can execute it, with a few exceptions. However, the action button itself cannot be edited, so it is not necessary to control edit permissions.

ⓘ **Example**

- If the user does not have permission to change the Status button, they could still press the Cancel button and change the contract status to Cancelled.
- If the user does not have permission to access a different table in the knowledgebase, they could still trigger record changes in that table through a Linked Record action.
- If the user can't access emails on a table, an Email action could be embedded in a rule to automate emails.

View Permissions

View permissions for action buttons apply to both the record form and the table view, but they have a caveat when placed on the table view. If users have permission to view an action button and it's placed on the table view, it appears for every record, even if the user doesn't have permission to edit that record. If a user clicks an action button from the table view for a record they don't have permission to edit, the action doesn't run and the user receives an error message.

You can change this behavior by setting the Always Show Action Button in Views global variable to No. This forces the system to evaluate users' permissions and only display action buttons on the table view for records that they also have permission to edit. However, this may impact system performance if you have many records or complex permissions.