CDS Authoring Tool

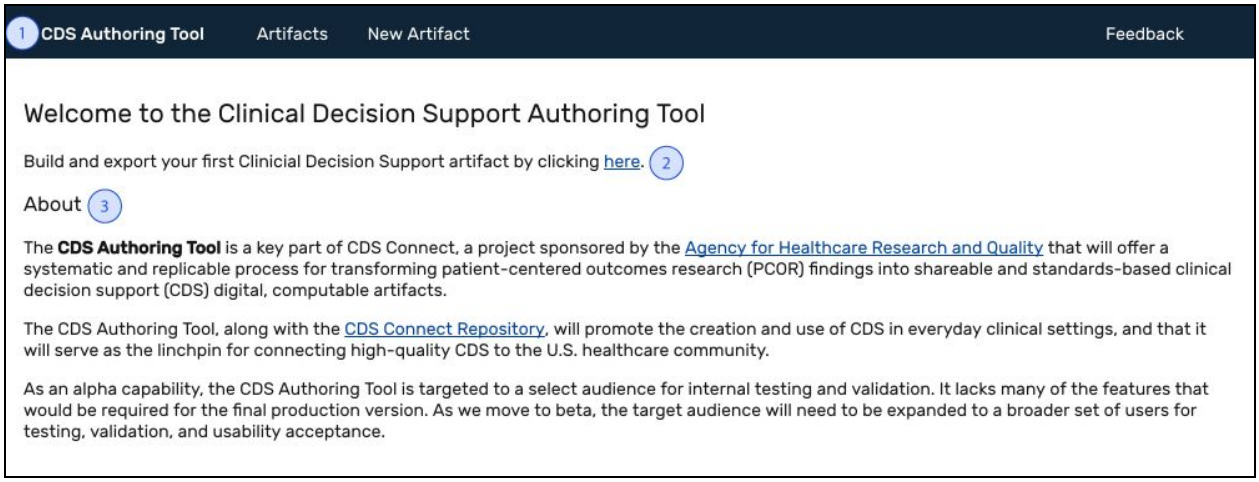
User Guide

This document helps new users of the Clinical Decision Support Authoring Tool navigate the features of the application and acquire the knowledge of how to use the tool to build CDS artifacts.

Eric Benoit, Craig McGinley, Juhan Sonin – 31.Jul.17 – GoInvo –
cds-authoring-list@lists.mitre.org

1.1 Homepage

The homepage is the first page arrived at when entering the CDS Authoring Tool application and serves as a starting point for the user.



1. Main Navigation
2. A quick link to get started on a new artifact.
3. A section explaining the role of the CDS Authoring Tool in the CDS Connect ecosystem, with links to the “Agency for Healthcare Research and Quality” and “CDS Connect Repository” website.

1.2 Main Navigation

This dark blue bar is present across the top of all pages and allows the user to traverse between different sections of the application.

1. Clicking “CDS Authoring Tool” will return user to the Homepage.
2. Clicking “Artifacts” navigates to the Artifacts page. When currently on this page, the background of this tab will be black.
3. Clicking “New Artifact” navigates to the Builder page with a new, blank artifact. When currently on this page, the background of this tab will be black.
4. Clicking “Feedback” opens the user’s mail application to email questions, concerns, or general feedback to the CDS Authoring Tool creators.

2.1 Artifacts page

The Artifacts page presents a list of all the user’s existing artifacts. It includes the ability to add, edit, and delete artifacts.

1. “New Artifact” form
    a. The field to enter the new artifact’s name, which is required to create a new artifact.
    b. The field to enter the new artifacts version number.
    c. Submit button to add the new artifact to the list below.
2. Artifacts list
    a. Pencil button opens a modal to edit the artifact information (edit modal detailed below).
    b. The name of the artifact. This also serves as a link, which upon clicking will open the artifact in the Builder page.
    c. The version number of the artifact.
    d. The date/time the artifact was last updated.
    e. Button to delete the artifact entirely.

2.2 Edit Artifact modal

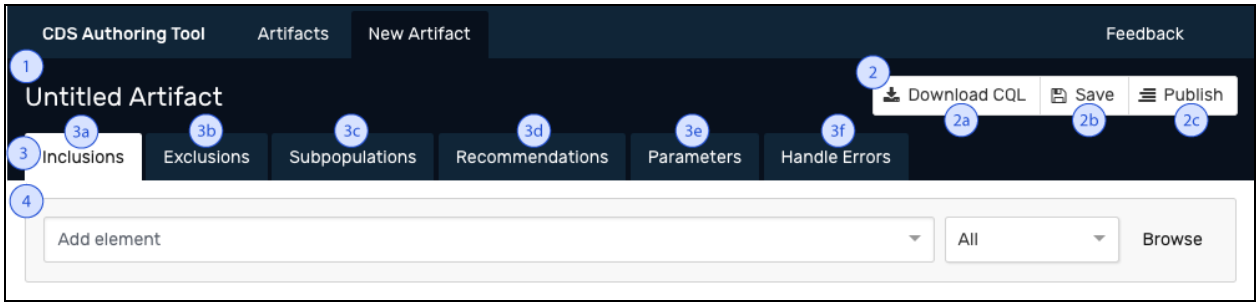
Upon clicking the pencil button, this modal will open allowing the user to edit an existing artifacts name and/or version number.

1. The field to edit the artifact’s name, which cannot be blank.
2. The field to edit the artifact’s version number.
3. “Save” button to confirm and save changes made to the artifact and close the modal.

4. “X” button to cancel changes made to the artifact and close the modal.

3.1 Builder page

The Builder page is the workspace for building artifacts in the CDS Authoring Tool.



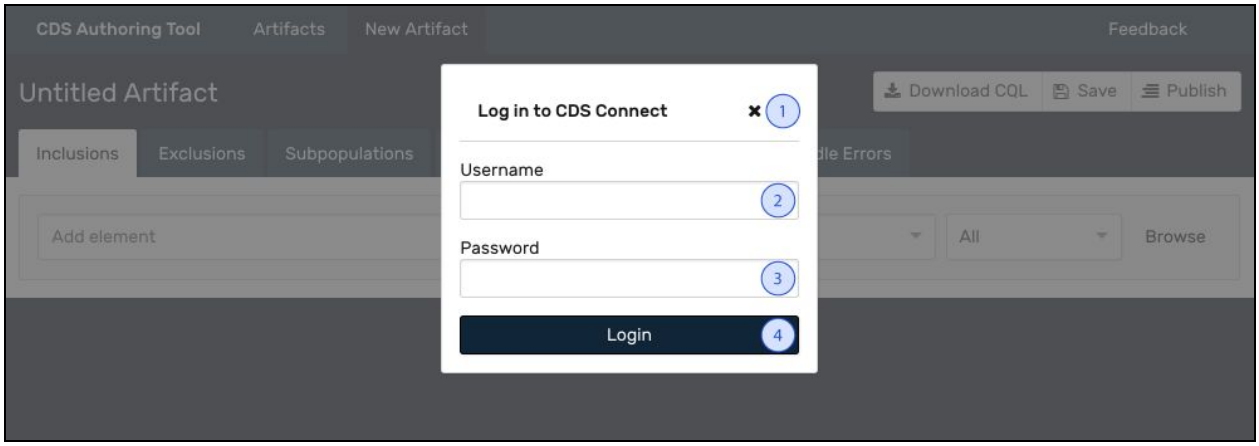
1. The Builder page header which contains the name of the artifact currently being worked on and the artifact menu bar (2). This header will be present across all Builder page tabs (3).
2. The Builder page menu bar, which allows the user to perform actions on the entire artifact.
    a. Clicking “Download CQL” will generate the CQL code for the current artifact and download the result onto their computer inside a zip file. The zip file will contain a folder with the artifact’s CQL code as well as any necessary dependency files.
    b. Clicking “Save” will save all changes made to the artifact.
    c. Clicking “Publish” will launch the publishing workflow, detailed below in section 3.2
3. The Builder page tabs, which divide the Builder page into workflow sections for building an artifact. This helps keep sections of the artifact organized for the user (more on each tab below in sections 3.8-3.13).
    a. Clicking will switch to the “Inclusions” section.
    b. Clicking will switch to the “Exclusions” section.
    c. Clicking will switch to the “Subpopulations” section.
    d. Clicking will switch to the “Recommendations” section.
    e. Clicking will switch to the “Parameters” section.
    f. Clicking will switch to the “Handle Errors” section.
4. The tab content area, which in this figure shows a blank “Inclusions” area (more on each tab below in sections 3.8-3.13).

3.2 Publish modal

The publish modal will walk the user through the process of publishing the current Artifact to the CDS Connect Repository. The user can upload Artifacts authored in the tool

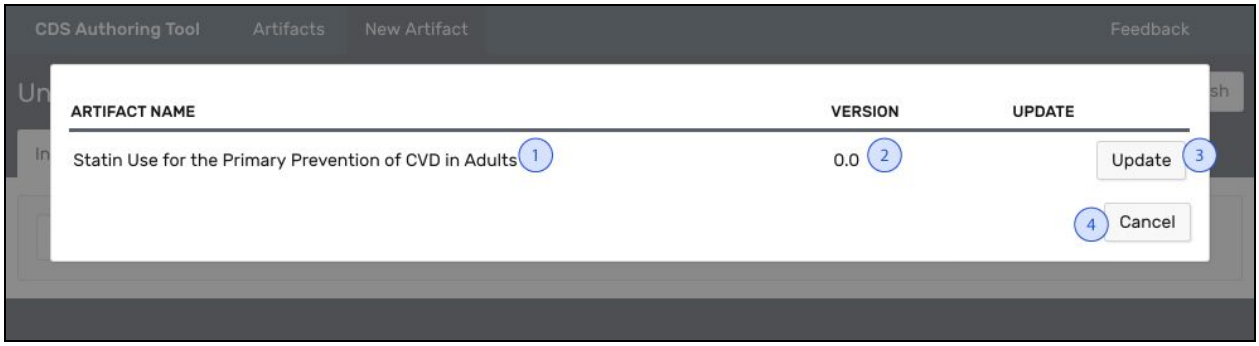
to corresponding Artifacts in the repository that they have access to. The following walks through the publishing process.

First, the user must authenticate with the CDS Connect repository.



1. The “X” button can be used to cancel the login/publishing process.
2. Enter the CDS Connect username.
3. Enter the CDS Connect password.
4. Click “Login” button to authenticate with the CDS Connect repository.

Upon authenticating with the CDS Connect repository, the publish modal will present a list of Artifacts pertaining to the authenticated user. The user can select which artifact to upload the newly authored material for.



1. The name of the artifact in the CDS Connect repository.
2. The current version of the artifact in the CDS Connect repository.
3. The “Update” button is used to select the artifact to be uploaded to. This will begin the upload.
4. The “Cancel” button is used to cancel the publishing process.

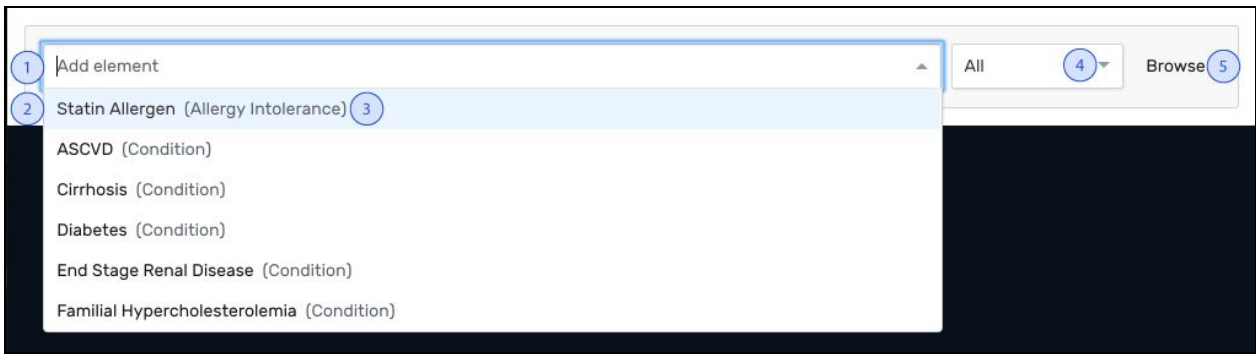
While the newly authored material is being uploaded to the corresponding Artifact in the CDS Connect repository, the modal will display an “uploading” message, seen below. The Publish modal will automatically close when the upload has successfully completed.

In the case of an error in logging in or uploading, the Publish modal will present an error message like below.

1. Upon receiving the error message, the user can close the modal using the “X” button and, if they wish, retry the process by clicking “Publish” button again.

3.3 Element Picker

The element picker is used to find and add elements into various sections of the artifact. This same module is used across the “Inclusions”, “Exclusions”, and “Subpopulations” sections of the application.

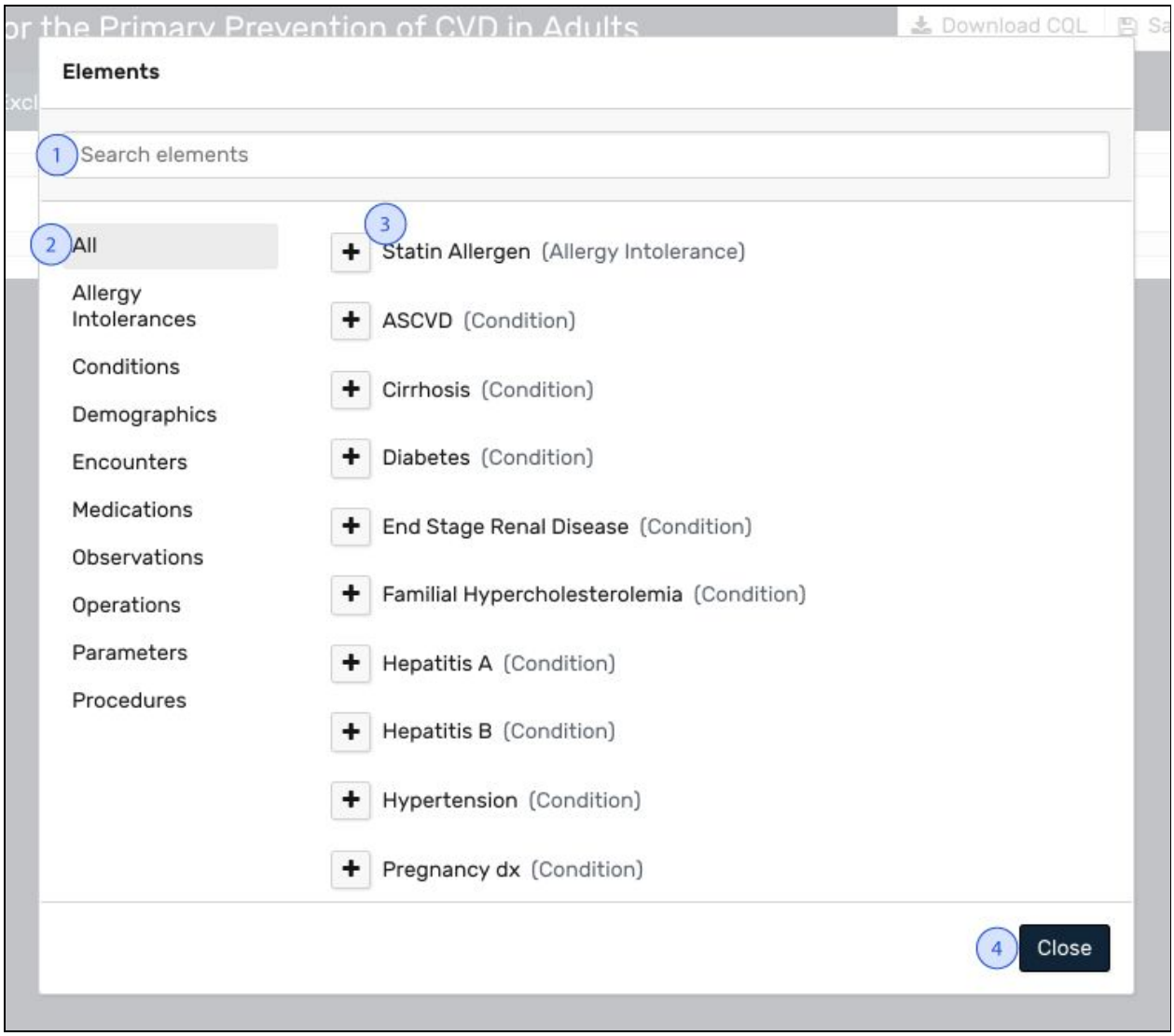


1. The element search field. Click in this field to open the element dropdown (2). Type in this field to narrow the results in the dropdown by the term typed.
2. The element dropdown, which shows relevant elements filtered by category selected (4) and search term (1).
3. An individual element which can be added to the artifact. The element name is displayed, and if looking in “All” categories, will be followed by the category type in parenthesis. Clicking on an element will add it to the artifact.

4. The category dropdown, which allows the user to select which category of elements they wish to filter results by.
5. Clicking “Browse” will launch the “Element Browse” modal.

3.4 Element Browse

Similar to the Element Picker, the Element Browse modal enables the user to have a more spacious area to search, view, and add elements to an artifact.

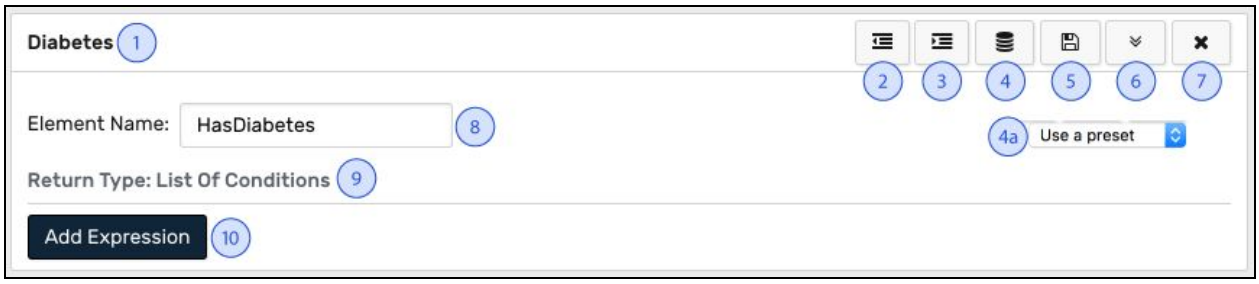


1. The element search field. Type in this field to narrow the results in the element list (3) by the term typed.
2. The element category list. Clicking on a category will filter the results in the element list (3) by the selected category. The selected category is highlighted with a gray background.

3. The element list is the result of the filtering by category and search term within the modal. Click on an element (name or "+" button) to add it to the artifact and close the modal.
4. Clicking "Close" will close the modal without adding any elements to the artifact.

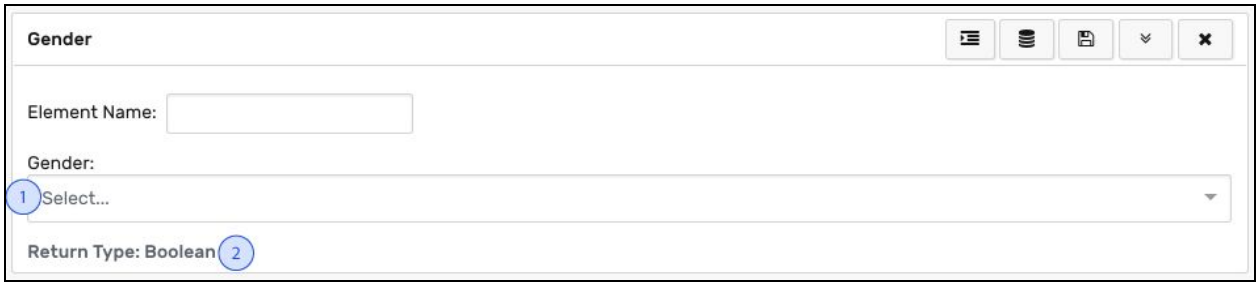
3.5 Elements

Elements are the main building blocks for an artifact. Each element represents different conditions, medications, demographics, etc. Using a combination of elements together in groups (covered below in "Element Groups") helps the user define different populations for the artifact.



1. The name of the type of element, in this case the "Diabetes" condition.
2. Move this element out of the current group it belongs to. (Outdent)
3. Move this element inside a new group. (Indent)
4. Clicking this button allows the user to select a saved preset to use for this particular element type.
    a. Upon clicking the "Presets" button (4), a dropdown menu will appear to pick the appropriate preset to apply.
5. Save the current element configuration as a preset.
6. Expand or collapse the current element (helps preserve space and keep the workplace tidy).
7. Clicking "X" on an element will delete it.
8. The name of this specific element which the user can specify in the field.
9. The current "Return Type" of this element based on the Expressions (more in section 3.6 Expressions).
10. Add an expression to this element (more in section 3.6 Expressions).

Some elements require additional fields to be filled in, or don't support adding Expressions.

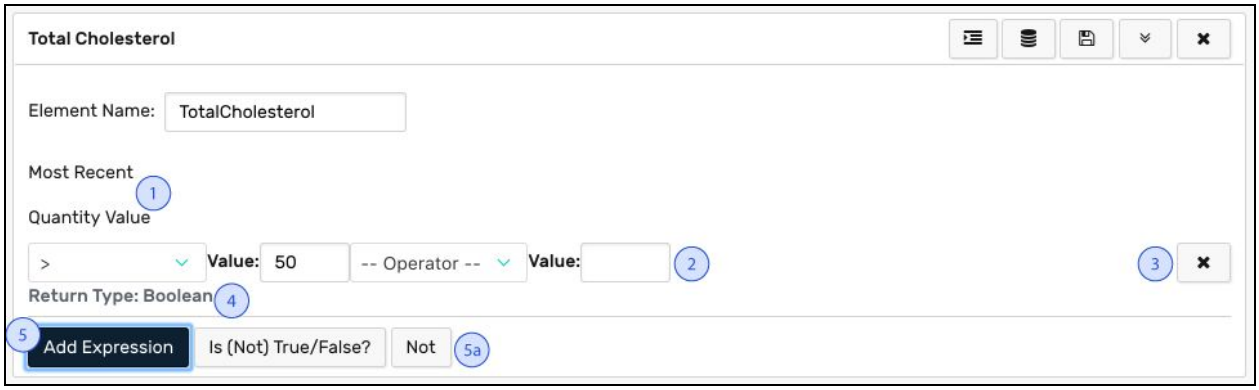


1. Some elements have more fields to fill in beyond just the “Element Name”. This element (“Gender”) requires the user to select which gender is desired, using the select menu. Fill out every field in an element to ensure proper CQL code is generated.
2. Some elements do not support “Expressions”. This element (“Gender”) does not have any Expressions that can be applied to it, and automatically returns a Boolean value.

3.6 Expressions

Expressions modify an Element to define or narrow its intent. Many Elements will start as a list, which if left as such, *will generate invalid CQL*. The user must ensure that the Return Type of every Element returns a “Boolean” value. To achieve this, the user can apply Expressions to narrow or filter the previous Expression further. For example, one could start with a list of conditions and then apply the “Most Recent” Expression to find the most recent condition in the list, then apply “Quantity Value” and “Value Comparison” to achieve a Boolean Return Type (pictured below).

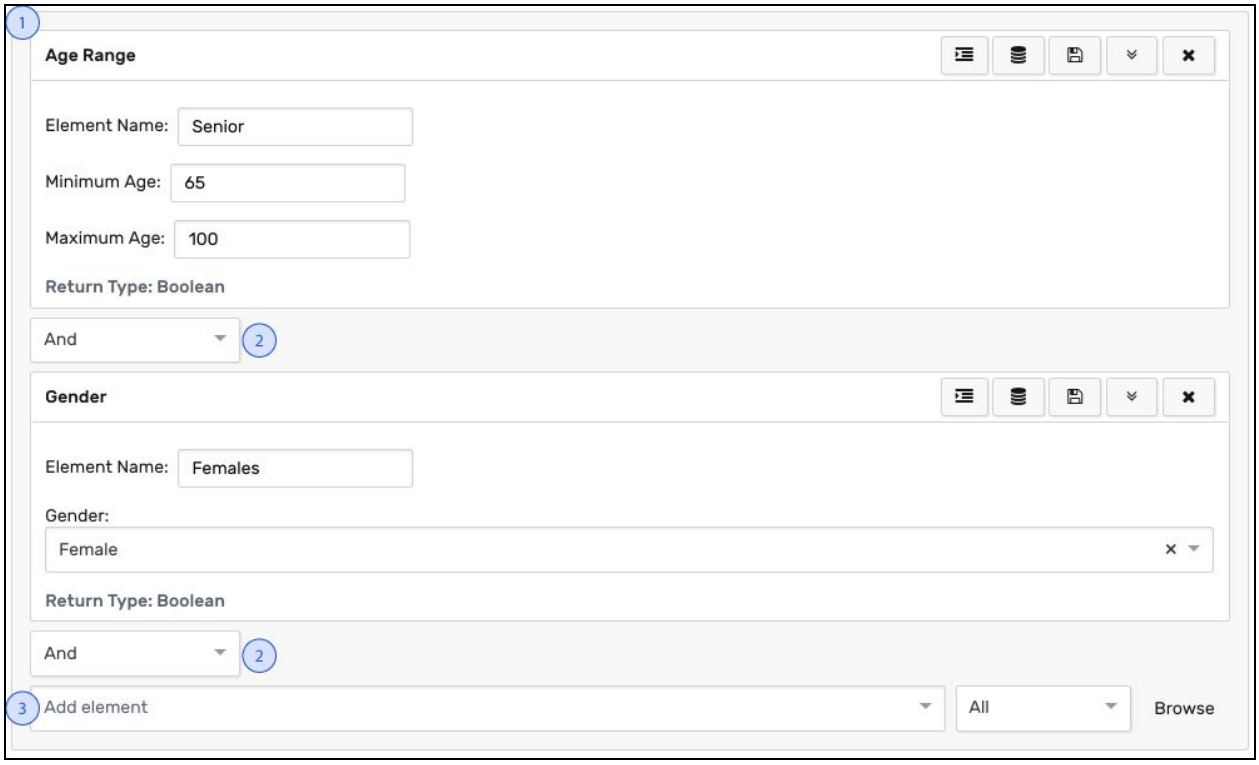
Expressions chain onto one another in succession. The return type from the first Expression applied will narrow the types of Expressions that can be applied as the second, and so on. The CDS Authoring Tool performs this filtering for the user automatically.



1. The list of Expressions applied to the element so far (in this example “Most Recent”, “Quantity Value”, and a comparison).
2. The last Expression that has been applied will appear at the bottom of the Expressions list, directly above the “Return Type” label. In this example the last Expression happens to be a comparison. Note that the comparison Expression has additional fields that need to be filled in. With comparisons, it is acceptable to fill in one or both sides of the comparison.
3. The last Expression can be removed by clicking the “X” button on the far right side of the Expression item. Because of the nature of the Expression chaining, only the last most Expression can be deleted. If the user wishes to delete an Expression higher up in the list, they must first delete all the ones below it.
4. Once again, the “Return Type” of the Element will always be listed at the end of the Expressions list.
5. Clicking the “Add Expression” button will reveal a list (to the right of the button) of relevant Expressions that can be applied on the Element.
    a. Clicking a revealed Expression button will add that Expression to the Element.

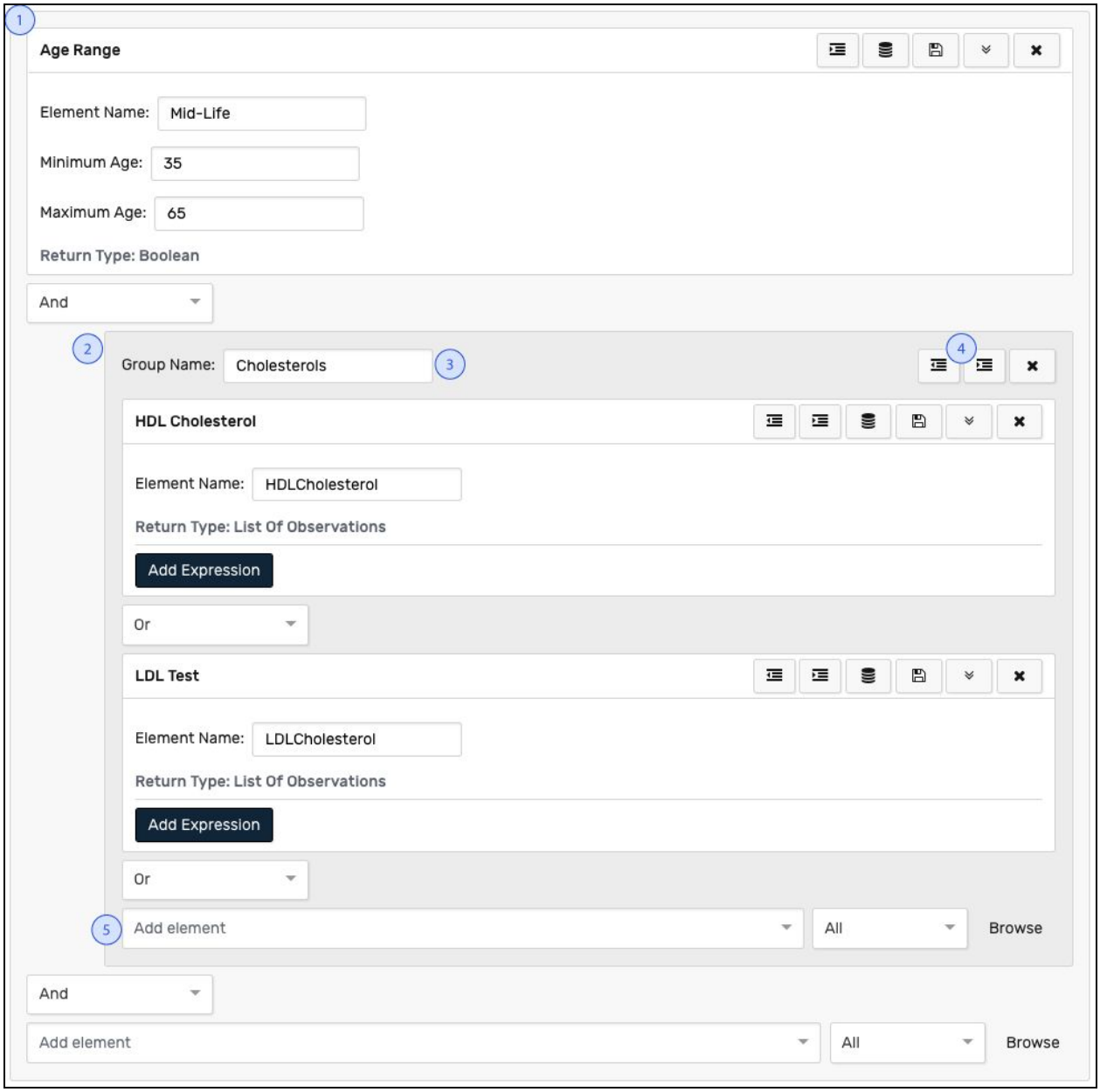
3.7 Logic Elements

Logic Elements are groups of Elements tied together by a particular conjunction, “And” or “Or”. By stringing Elements together with conjunctions, a set of logic can be created to define a population.



1. The outermost light gray box is the first level (“root”, or “main”) Logic Element, which houses all other Elements and ties them together with a conjunction (2).
2. Between every Element inside a Logic Element group, there will be a dropdown denoting the conjunction used to tie them together. The options for conjunctions are “And” or “Or”. Note that within any particular group, the same conjunction must be used. For instance, in the example above, if one changes the first occurrence of the conjunction (the first (2) marker) to “Or”, the second conjunction will also update to “Or”. This avoids creating ambiguous logic for the system to interpret. The user can think of “And” as meaning every Element must be true, while “Or” means at least one of the Elements must be true.
3. Every Logic Element will have an Element Picker (Section 3.3) at the bottom, to allow the user to add new Elements to the group.

Logic Elements can also be “nested”, which is to say, Logic Elements can have other Logic Elements inside them. Logic Elements can be nested as much as the user desires. Using the Indent/Outdent Buttons helps the user quickly group and ungroup individual Elements and entire Logic Element groups.



1. Once again, the outermost Logic Element here is represented as the light gray box.
2. A nested Logic Element, represented by a level of indentation as well as a darker colored gray background.
3. Nested Logic Elements can be named using the “Group Name” field, similar to Elements.
4. Entire Logic Element groups can be indented or outdented, similar to individual Elements. This helps the user move entire groups rather than just one Element at a time.
5. Again, note that every Logic Element group will have its own Element Picker, allowing the user to add more Elements or nested Logic Elements to the group.

3.8 Inclusions

The Inclusions section uses Elements, Expressions, and Logic Elements to create a target population which are qualified to receive a Recommendation from the Artifact. The Inclusions populations with the Exclusions population filtered out creates the general population for the Artifact. Every interaction required to build Inclusions is covered in above sections.

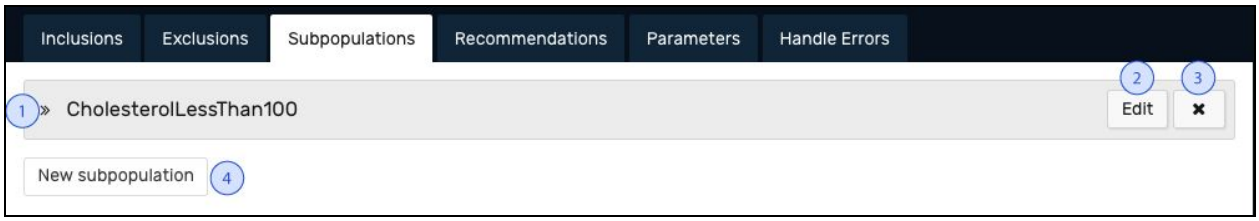
3.9 Exclusions

The Exclusions section uses Elements, Expressions, and Logic Elements to create a target population which is generally excluded from receiving a Recommendation from the Artifact. The population matching Exclusions are filtered out of the Inclusions population, which creates the general population for the Artifact. Every interaction required to build Exclusions is covered in above Sections.

3.10 Subpopulations

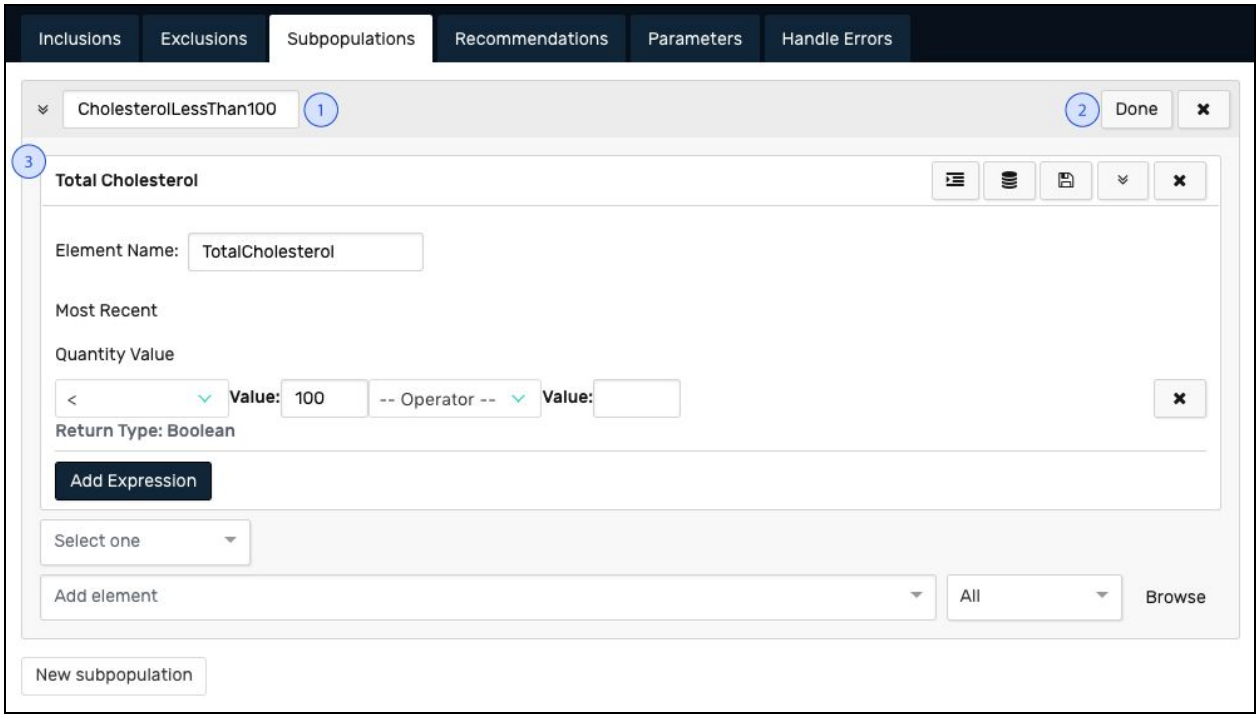
The Subpopulations section uses Elements, Expressions, and Logic Elements to create named target populations which can then be applied to a Recommendation. This helps the user further filter the general population created from the combination of Inclusions and Exclusions. There are two default "Subpopulations" which can additionally be applied to a recommendation, "Doesn't Meet Inclusion Criteria" and "Meets Exclusion Criteria" (more in section 3.11 Recommendations). Most interactions required to build Subpopulations are covered in above sections, but Subpopulations has a few differences.

Subpopulations are presented as a list of named populations, which can be expanded or collapsed. The following shows a collapsed Subpopulation.



1. In this example, the Subpopulation "CholesterolLessThan100" is collapsed. Clicking anywhere in the gray bar will expand the Subpopulation for editing.
2. The "Edit" button is another way to expand the Subpopulation for editing.
3. The "X" button is used to delete the Subpopulation.
4. "New Subpopulation" will add a new Subpopulation at the bottom of the list, ready for editing.

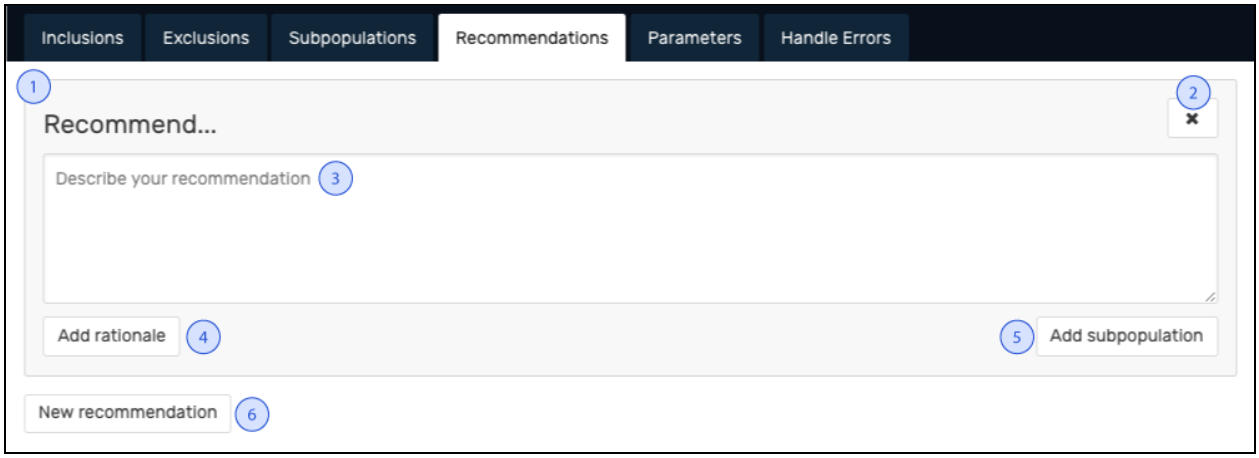
The following demonstrates an expanded Subpopulation, ready for editing.



1. The name of the Subpopulation can be edited when the Subpopulation is expanded.
2. Clicking “Done” will save changes to and collapse the Subpopulation.
3. The content of a Subpopulation is built the same as Inclusions or Exclusions, and uses items covered in above sections (Elements, Logic Elements, Element Picker, etc).

3.11 Recommendations

Recommendations are the resulting notices that should be delivered to the clinician after the CDS Artifact is executed. Recommendations are written as free text and can have an accompanying Rationale. Most Recommendations will apply to the population which meet Inclusions logic and doesn’t meet Exclusions logic. There are two exceptions to this, where alternate default Subpopulation options can be applied to a Recommendation, which is covered below.

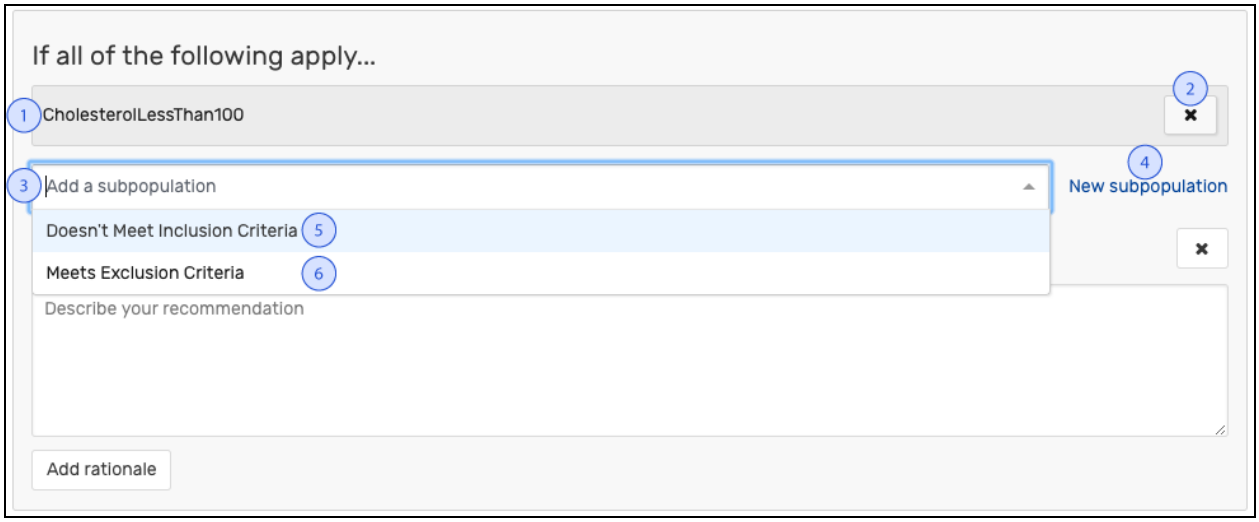


1. A blank Recommendation shown as the light gray box in this image.
2. The “X” button is used to delete a Recommendation.
3. The Recommendation’s content is written in free text using this field. This is the message that the clinician will read in the EHR system if the Recommendation is triggered.
4. Clicking “Add rationale” will append an additional free text field where the user can enter the supporting evidence or reasoning for the Recommendation. This is covered below
5. Recommendations can be further filtered by Subpopulations, which is performed by clicking “Add subpopulation”. This will prepend an area above the Recommendation to add Subpopulations, covered below.
6. “New recommendation” adds a new Recommendation to the list of Recommendations.

Any Recommendation supports having an optional accompanying Rationale, pictured below.

1. A free text field to enter the Rationale for the Recommendation.

Recommendations can be further filtered by Subpopulations, to target different Recommendations for different groups within the general target population, shown in the image below.

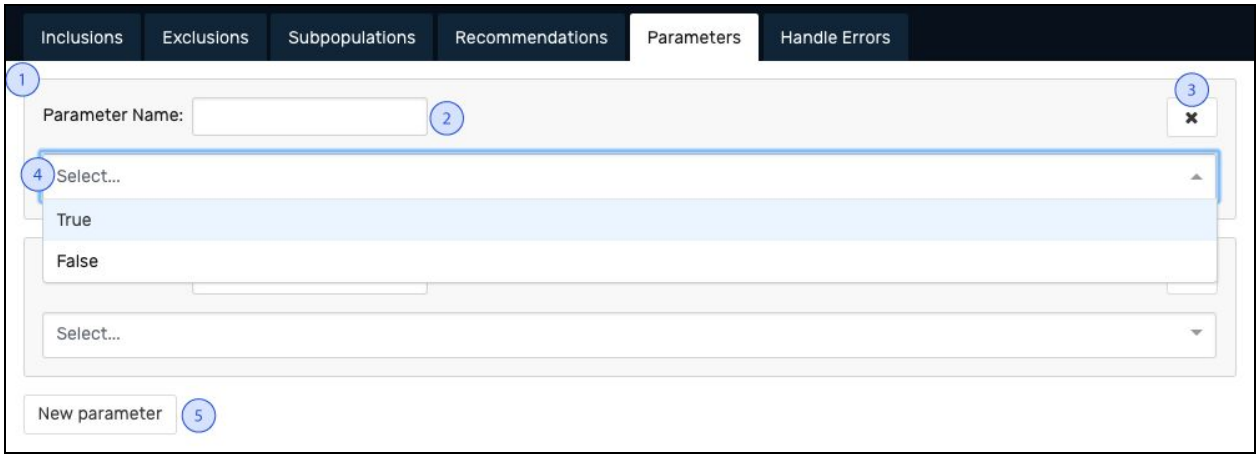


1. An applied Subpopulation on this Recommendation. This means this Subpopulation's logic will have to evaluate to true for a given patient in order for the Recommendation to be delivered.
2. The "X" button removes the Subpopulation from the Recommendation.
3. A field to search for and select the Subpopulations to apply to the Recommendation. Search for Subpopulations by typing here. Click a Subpopulation in the dropdown list below to add it to the Recommendation.
4. A link to add a new Subpopulation in the Subpopulations tab.
5. One of the default "Subpopulation" options. This default option is supplied to allow the user to add Recommendations for patients who did not meet the Inclusion criteria and thus were not part of the general population for this CDS Artifact.
6. One of the default "Subpopulation" options. This default option is supplied to allow the user to add Recommendations for patients who met the Exclusion criteria and thus were not part of the general population for this CDS Artifact.

3.12 Parameters

Parameters allow the user to create named, reusable Boolean values which help readability and communicate intent in the resulting CQL code. An example of this might be a Parameter called "GradeCRecommendationEnabled". One user might choose to

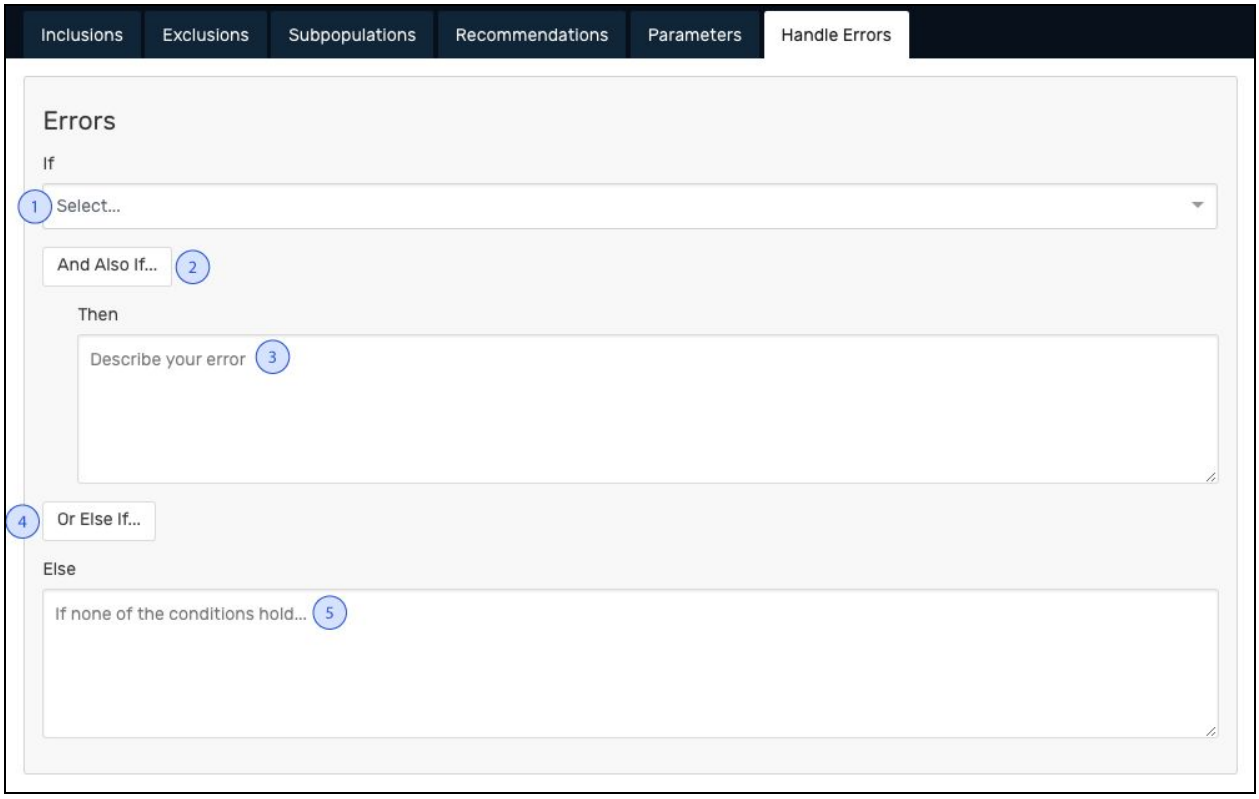
accept this value as true, while another may prefer to set the GradeCRecommendation to false. Parameters are optional additions to the artifact, and can be used in building Inclusions, Exclusions, and Subpopulations, as well as Error Handling.



1. Each light gray box is an individual Parameter object.
2. Parameters should be aptly named using this field.
3. The “X” button deletes a Parameter.
4. Parameters can have a Boolean value (“True” or “False”), selected by the user with this dropdown.
5. The “New parameter” button adds a new Parameter to the list.

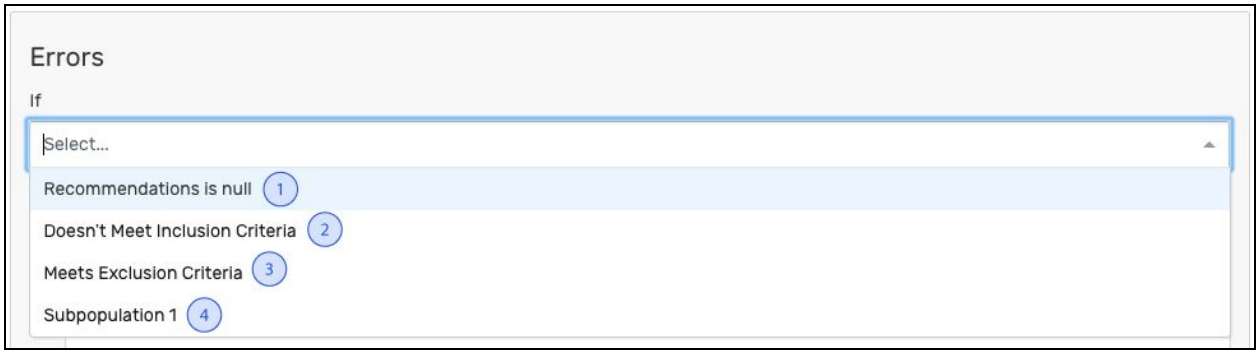
3.13 Handle Errors

The “Handle Errors” tab is an area to optionally direct the system how to handle various errors encountered when running the CDS Artifact. This helps the user define what error messages to display when certain data is missing, for instance. Error handling is built by chaining together “If” statements, which say “if this condition is met, then deliver this error message”.



1. Each “If” statement will require a condition, which is selected by the user with this dropdown. Conditions can include user-defined Subpopulations and Parameters, as well as a few default options. More below.
2. The user can opt to have a second “If” statement tied to the first, meaning both conditions must be met in order to deliver the error.
3. This free-text field is used to enter the error message associated with the “If” condition.
4. The user can add as many “If” statements to the error handling as desired. Clicking “Or Else If…” will add another “If” statement to the list.
5. The final free-text area is used to define the error message that will be displayed if none of the “If” conditions are met.

Similar to Recommendation’s Subpopulations, “If” statement conditions for errors support a few default options, detailed below.



1. This default condition will be met if none of the user-defined recommendation conditions are met.
2. The same as Recommendation's Subpopulations, the "Doesn't Meet Inclusion Criteria" condition will be met for patients who did not meet the Inclusion criteria and thus were not part of the general population for this CDS Artifact.
3. The same as Recommendation's Subpopulations, the "Meets Exclusion Criteria" condition will be met for patients who meet the Exclusion criteria and thus were not part of the general population for this CDS Artifact.
4. The user-defined Subpopulations and Parameters will be displayed after the three default options.