

Editing JSON Messages

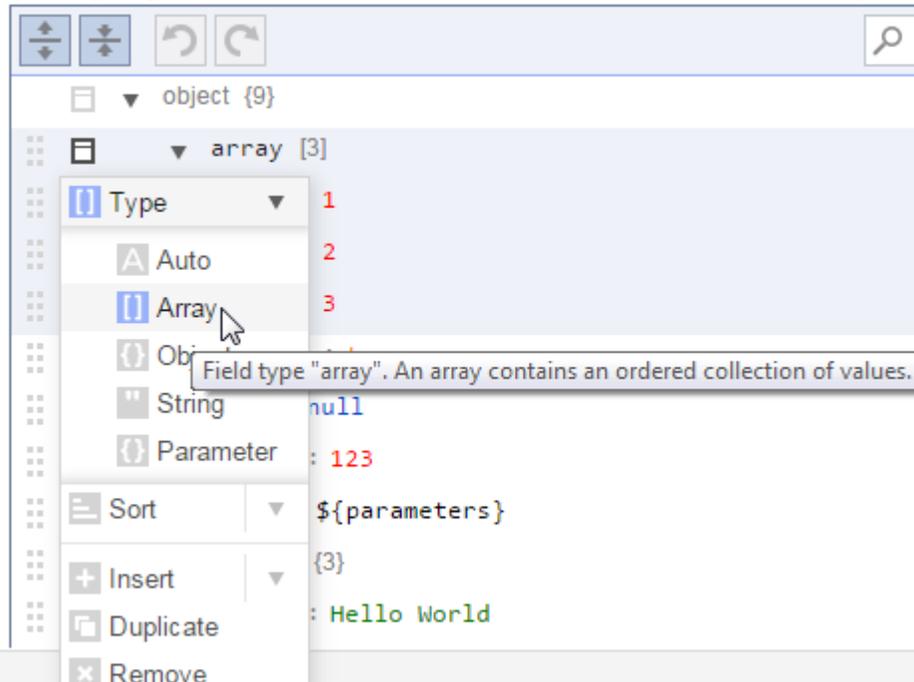
The JSON Editor provides an editable tree that makes it easy to create, duplicate, and remove fields, as well as to edit field content.

The bar at the top of the editor provides the following functionality:



- **Expand all:** Expands all fields. You can also expand and collapse individual fields by clicking the arrow icons within the editor area.
- **Collapse all:** Collapses all fields. You can also expand and collapse individual fields by clicking the arrow icons within the editor area.
- **Undo:** Reverts the last action.
- **Redo:** Repeats an action that you undid.
- **Search:** Highlights the designated search term.

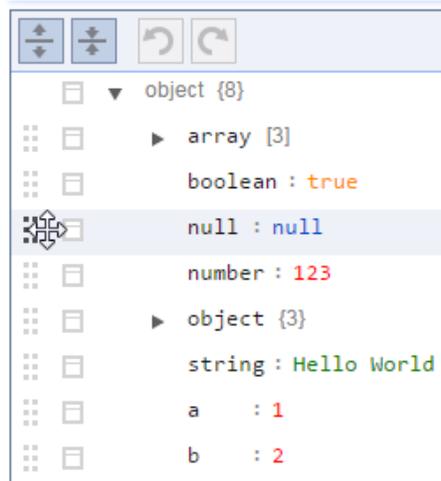
You can extend and modify fields by using the commands available from the command icons. The available commands depend on the field type.



Available commands could include:

- **Type:** Changes the field's type. Options are:
 - **Auto:** The field type is automatically determined from the value; it can be a string, number, boolean, or null.
 - **Array:** An ordered collection of values.
 - **Object:** An unordered set of key/value pairs.
 - **String:** Field type is not determined from the value, but always returned as a string.
- **Sort:** Sorts the children of an array or object. For an array, the values of the children will be sorted. For an object, the children will be sorted by key. Arrays and objects can be sorted ascending or descending.
- **Insert:** Inserts a new field before the current field. Available types are auto (default), object, array, and string.
- **Append:** Inserts a new field after the current field. Available types are auto (default), object, array, and string.
- **Duplicate:** Duplicates the field, including all children.
- **Remove:** Removes the field, including all children.

Additionally, you can rearrange fields by dragging and dropping a field's drag icon.



Note that when you are working with JSON, a special notation is used for parameterizing a number or boolean field within a JSON message: `$(number:<value>)` or `$(boolean:<value>)`. For example, to parameterize a number field with the column Count, you would use `$(number:Count)`.