

RichText Control

RichText control is used to display properties of datatype FormattedText from the associated ItemType. In edit mode, RichText control shows toolbar, using it you can format the text with bold, italic, font, color etc. It also allows you to insert items like images, hyperlinks and videos, upon saving all these items are embedded inside FormattedText on the property.

The screenshot shows the 'RichText Settings' dialog box. It includes the following fields and options:

- Name (maximum 32 characters):** custom_notes
- Label:** Seller Notes
- Data Type:** Formatted Text
- Tooltip:** (empty)
- Help Text:** (empty)
- Show Label:** Yes No
- Inline Toolbar:** Yes No
- Max Height (pixels):** 350
- Access:**
 - Show:** (empty) ✓ ✗
 - Edit:** (empty) ✓ ✗

Buttons: Ok, Cancel

- **Name:** It is used to uniquely identify control inside the template.
- **Label:** It is used to show label on the control. It can be localized by defined Languages and Locales inside innovator.
- **Data Type:** It shows the type of the property as defined in the ItemType definition. In case of RichText control, it will always be FormattedText, and it is shown as disabled.
- **Tooltip:** Tooltip will be shown when you hover the mouse on the information icon on the form field. It is derived from the property bound to the control.



- **Help Text:** Help Text flyout will be shown when you click on the information icon on the form field label. It is derived from the property bound to the control.
- **Show label:** If show label is selected as Yes, RichText control will be shown with a label at top-left corner.
- **Inline Toolbar:** If an inline toolbar is selected as Yes, toolbar will be shown within RichText control. Otherwise it will be shown in the popup after selecting the tick icon.
- **Max Height:** It is an optional setting, if you don't set any value for this, it will be shown with the height that fits the content to avoid scrollbar. By setting max height explicitly, RichText control will auto grow up to this height based on the content, if the content goes out of bounds, then it shows scrollbar.
- **Access:** It allows you to define roles which can access the control. Roles (in the form group identities) can be at Show and Edit level. When control is hidden because you do not have defined role, property associated with this control is not even fetched from server.
- **Read Only:** Based on the option selected for Read Only, control can become read-only even if you can edit the properties used inside the page with lock. You can select one of the options like 'Yes', 'No', 'Expression', or 'Script'. Expression allows you to define a Boolean expression using properties from the current and previous pages (in case of multi-page application) to conditionally make the control read-only if expression is evaluated to true. Expression will be evaluated based on the cached property values on the client. Script allows you to write custom JavaScript code that should return Boolean value. If returned value is true, the control will be read-only.
- **Display Condition:** Based on the option selected for Display Condition, the control will be shown or hidden in the wizard at runtime. You can select one of the options like 'Always Show', 'Always Hide', 'Expression', or 'Script'. Expression allows you to define a Boolean expression using properties from the current and previous pages (in case of multi-page application) to conditionally display the control if expression is evaluated to true. Expression will be evaluated based on the cached property values on the client. Script allows you to write custom JavaScript code that should return Boolean value. If returned value is false, the control will be hidden.
- **Reset Dependency:** It allows you reset value of the control to the default if any of the controls defined in "Reset Dependency" have their value changed at runtime.
- **Margin:** It allows you to set required top, right, bottom, left margins with respect to adjacent controls.
- **On Load:** OnLoad event allows you to write custom logic that will be executed after the value in the control is loaded. Please refer to the Template Customization section for more details.
- **On Change:** OnChange event allows you to write custom logic that will be executed after the value in the control is updated. Please refer to the Template Customization section for more details.

