

Code Builder

Introduction

The **Code Builder** allow app designers to create and reuse code snippets in a Joget app.

To retrieve code snippets created in this builder, use the [Code Builder Hash Variable](#).

The benefits of this addon builder are:

1. Enhance maintainability. Changes done to a code snippet will be reflected to all of its usages in the app.
2. Access to predefined code snippet templates to get started on scripting quickly.
3. Code indentation and syntax highlighting via the code editor.
4. Ability to [find all usages](#) of current code snippet template via [Builder Advanced Tools](#).
5. If required, able to create highly customized code snippet templates via plugin.

You can try out this builder at <https://archives.joget.org/addons/>.

New Feature

The Code Builder is a plugin created for the [Addon Builders](#). This is a new feature in Joget DX Enterprise and Professional editions.



This plugin is still in **PREVIEW** stage and undergoing development. **NOT** recommended for production use.

The feature set found in the current plugin is not finalized and may be expanded in the future.

The screenshot displays the Code Builder interface within a Joget application. The title bar shows 'CODE SNIPPET BUILDER Expenses Claim v3: Test (Published)'. The interface includes tabs for 'DESIGN', 'PROPERTIES', and 'SAVE', along with 'Undo', 'Redo', and 'Advanced Tools' options. The 'Type' dropdown is set to 'Bean Shell' and the 'Template' dropdown is set to 'Form Options Binder'. The code editor contains the following Java code:

```
1 import org.joget.apps.app.service.AppUtil;
2 import org.joget.commons.util.LogUtil;
3 import org.joget.apps.form.model.Element;
4 import org.joget.apps.form.model.FormData;
5 import org.joget.apps.form.model.FormRowSet;
6 import org.joget.apps.form.model.FormRow;
7 import org.joget.apps.form.service.FormUtil;
8
9 /**
10  * @param element - Element that this binder is tie to
11  * @param formData - The data holder of the whole form
12  *
13  * @return FormRowSet object which contains one or more FormRow object.
14  * All FormRow objects are expected to have "value" and "label" property.
15  */
16 public FormRowSet load(Element element, FormData formData) {
17     FormRowSet rows = new FormRowSet();
18
19     // Continue your code here
20
21     /* Sample code:
22      *   FormRow option = new FormRow();
23      *   option.setProperty(FormUtil.PROPERTY_VALUE, "sample-value");
24      *   option.setProperty(FormUtil.PROPERTY_LABEL, "sample-label");
25      *   rows.add(option);
26      */
27
28     return rows;
29 }
30
31 return load(element, formData);
```

The bottom status bar shows navigation icons and the text 'Forms & UI | Processes | Properties | Users | Monitor | Settings | Quick Edit: Off'.