

RenderTemplateAction

The `RenderTemplateAction` performs the job of taking a template - that is, a string of code, in a templating language, such as `Velocity`, and merging it with a set of data (known as a context), to produce some using-facing output, such as a `String` of HTML.

Examples

Canonical Form

```
RenderTemplateInput input = new RenderTemplateInput(qInstance);
input.setSession(session);
input.setCode("Hello, ${name}");
input.setTemplateType(TemplateType.VELLOCITY);
input.setContext(Map.of("name", "Darin"));
RenderTemplateOutput output = new RenderTemplateAction.execute(input);
String result = output.getResult();
assertEquals("Hello, Darin", result);
```

Convenient Form

```
String result = RenderTemplateAction.renderVelocity(input, Map.of("name", "Darin"),
"Hello, ${name}");
assertEquals("Hello, Darin", result);
```

RenderTemplateInput

- `code` - **String, Required** - String of template code to be rendered, in the templating language specified by the `type` parameter.
- `type` - **Enum of VELOCITY, Required** - Specifies the language of the template code.
- `context` - **Map of String → Object** - Data to be made available to the template during rendering.

RenderTemplateOutput

- `result` - **String** - Result of rendering the input template and context.