## Java Use API

* Java use API enables a HTL file to initialize a java class to access helper methods in that class. This is useful for manipulation to be done on java side which are not doable in HTL
* The above said java class can be any of the below
  + A java class available at same level as htl file
  + Bundled java file(built and deployed as OSGi container)

A **local Java use-class**

is recommended when the use-class is specific to the component in question.

A **bundle Java use-class**

is recommended when the Java code implements a service that is accessed from multiple HTL components.

<div data-sly-use.info="Info">

info is the identifier through which we can utilize ‘Info’ Java class.

The scope of info is global after being declared and not limited to the

Element that has *data-sly-use* statement

<div data-sly-use.info="${'Info' @ text='Some text'}">

Info – identifier for Info class

Text – parameter for Info class

* Java use API can take html files for initialization
* A html file can have multiple templates in it.

Consider below ‘somefile.html’ it has 2 templates , where extra – local identifier for template 1 &

extra1 – is the local identifier for template 2

<template data-sly-template.extra="${@ text}"

data-sly-use.extraHelper="${'ExtraHelper' @ text=text}">

<p>${extraHelper.reversedText}</p>

</template>

<template data-sly-template.extra1="${@ text}"

data-sly-use.extraHelper="${'ExtraHelper' @ text=text}">

<p>${extraHelper.reversedText}</p>

</template>

Now below Info.html uses above templates available in somefile.html as below

<div data-sly-use.info="Info"

data-sly-use.someIdentifier="extra.html">

<h1>${info.lowerCaseTitle}</h1>

<p>${info.lowerCaseDescription}</p>

<div data-sly-call="${ someIdentifier.extra @ text=properties.description}"></div?

<div data-sly-call="${ someIdentifier.extra1 @ text=properties.description}"></div>

</div>