

- 
- JavaScript
  - JavaScript
  - JS
  - JS
  - HTML
  - CSS
- 

## JavaScript

### JavaScript

AbstractCodeTemplate.getJavascriptMainPath()JavaScript YellowfinJavaScript Yellowfin

Java

my\_widget.js mainJavascriptPath()

```
setupResources() {
  addResource("my_widget.js", "text/javascript");
}

getMainJavascriptPath() {
  return "my_widget.js";
}
```

JavaScript

```
my_widget.js
define(function() {

  class myWidget {
    constructor(options) {
      //This will be called by YF when the code widget is rendered.
    }
  }

  return myWidget;
});
```

```
my_widget.js
define(['my_example_dependency.js'], function(MyExampleDependencyObject) {
  class myWidget {
    constructor(options) {
      //This will be called by YF when the code widget is rendered.
    }
  }

  return myWidget;
});
```

- API
- Element
- ResourceLoader
- Messenger

		JavaScript
API	<ul style="list-style-type: none"> <li>• DashboardAPI</li> <li>• WidgetAPI</li> <li>• CanvasAPI</li> <li>• Filters</li> </ul>	<pre> constructor(options) {   options.apis.dashboards.name;    options.apis.filters; } </pre>
Element	YellowfinDOMHTML	<pre> constructor(options) {   options.element.innerHTML = 'Hello World'; } </pre>
ResourceLoader	AbstractCodeTemplate.setupResources <a href="#">isLibrary</a> CodeWidget	<pre> options.resourceLoader.load('my_second_file.js', function(MySecondFile) {   //Do something with MySecondFile }); </pre>
Messenger	<div>edit</div> <div>getOptionValue(optionName)</div> <div>setOptionValue(optionName, optionValue)</div> <div>optionNameoptionValue</div>	<pre> <b>edit</b>  if(options.messenger.edit) {   //Custom Edit Code } else {   //Published Code }  <b>getOptionValue(optionName)</b>  console.log(options.messenger.getOptionValue('myOption'));  <b>setOptionValue(optionName, optionValue)</b>  options.messenger.setOptionValue('myOption', 'myOptionValue');  console.log(options.messenger.getOptionValue('myOption')); //Value will be 'myOptionValue' </pre>

JS

JS JavaScript JS

JS

YellowfinJS

Java

```
addResource("my_second_file.js", "text/javascript");
```

JavaScript

```
my_widget.js
define(['my_second_file.js'], function(SecondFile) {

});

my_second_file.js
define(function() {
    return class SecondFile {
        //My Class
    }
});
```



YellowfinsetupResourceisLibrary true

## JS

AbstractCodeTemplateisLibrary

Java

```
addResource(new Resource("jquery.js", "text/javascript", true));
```

JavaScript

```
define(['jquery.js'], function($) {

});
```

## HTML

HTMLStringJavaScript

Java

```
addResource("my_html.html", "text/html");
```

JavaScript

```
define(['my_html.html'], function(MyHTML) {
    console.log(MyHTML); //HTML String
});
```

## CSS

CSS

Java

```
addResource("my_css.css", "text/css");
```

## JavaScript

```
define(['resource/my_css.css'], function(css) {  
    //The CSS variable will be undefined, however you should still include it in your function header to ensure  
    other inclusions aren't ignored  
})
```

---

[jarYellowfin](#)

API