

com.fr.stable.fun.JavaScriptPlaceholder

-
-
-
-
-
-
-
-

[JSjs.JavaScriptPlaceholderjsUI](#)

JavaScriptPlaceholder.java

```
package com.fr.stable.fun;

import com.fr.stable.StringUtils;
import com.fr.stable.fun.mark.Mutable;

/**
 * Created by richie on 16/4/26.
 * ,page.htmlform.html
 */
public interface JavaScriptPlaceholder extends Mutable {

    String MARK_STRING = "JavaScriptPlaceholder";

    int CURRENT_LEVEL = 2;

    /**
     *
     * @return
     */
    String placeHolderContent();

    /**
     *
     * @return
     */
    ScriptTag[] holderScripts();

    class ScriptTag {

        private String type;
        private String src;
        private String text;

        public static ScriptTag build() {
            return new ScriptTag();
        }

        private ScriptTag() {

        }

        public ScriptTag type(String type) {
            this.type = type;
        }
    }
}
```

```
        return this;
    }

    public ScriptTag src(String src) {
        this.src = src;
        return this;
    }

    public ScriptTag text(String text) {
        this.text = text;
        return this;
    }

    public String toTag() {
        StringBuilder sb = new StringBuilder();
        sb.append("<script");
        if (StringUtils.isNotEmpty(type)) {
            sb.append(" ");
            sb.append("type=");
            sb.append("\"");
            sb.append(type);
            sb.append("\"");
        }
        if (StringUtils.isNotEmpty(src)) {
            sb.append(" ");
            sb.append("src=");
            sb.append("\"");
            sb.append(src);
            sb.append("\"");
        }
        sb.append(">");
        if (StringUtils.isNotEmpty(text)) {
            sb.append(text);
        }
        sb.append("</script>");
        return sb.toString();
    }
}
}
```

AbstractJavaScriptPlaceholder.java

```
package com.fr.stable.fun.impl;

import com.fr.stable.fun.JavaScriptPlaceholder;
import com.fr.stable.fun.mark.API;

/**
 * Created by richie on 16/4/26.
 */
@API(level = JavaScriptPlaceholder.CURRENT_LEVEL)
public abstract class AbstractJavaScriptPlaceholder extends AbstractProvider implements JavaScriptPlaceholder {

    public int currentAPILevel() {
        return CURRENT_LEVEL;
    }

    public String mark4Provider() {
        return this.getClass().getName();
    }

    public String placeholderContent() {
        StringBuilder sb = new StringBuilder();
        ScriptTag[] scripts = holderScripts();
        if (scripts != null) {
            for (ScriptTag script : scripts) {
                sb.append(script.toTag());
            }
        }
        return sb.toString();
    }

    public ScriptTag[] holderScripts() {
        return new ScriptTag[0];
    }
}
```

FR	8.0		
FR	9.0		
FR	10.0		
BI	3.6		
BI	4.0		
BI	5.1		
BI	5.1.2		
BI	5.1.3		

plugin.xml

```
<extra-report>
  <JavaScriptPlaceholder class="your class name" />
</extra-report>
```

JS

plugin.xml

```
<extra-form>
  <JavaScriptPlaceholder class="your class name" />
</extra-form>
```

[cptReportletDealWith#dealWithHtml](#) [htmlReportletDealWith#createScriptPlaceholderString](#)

[AbstractFormActor#initMap4Form](#) [htmlAbstractFormActor#createScriptPlaceholderString](#)

[com.fr.stable.fun.JavaScriptFileHandler](#) [com.fr.decision.fun.WebResourceProvider](#)

JSCSS

[demodemo-java-script-place](#)

[com.fr.stable.fun.StylePlaceholder](#)