

Response



Apr 09, 2024 04:01

Supported Clients

SmartClient WebClient NGClient

Methods Summary

Boolean	close()	Needs to be called when not reading content via <code>getResponseBody</code> or <code>getMediaData</code> to be able to reuse the client.
String	getCharset()	Get the charset of the response body.
String	getException()	Getter for the exception message.
Array	getMediaData()	Get the content of response as binary data.
String	getResponseBody()	Get the content of the response as String.
Object	getResponseHeaders()	Gets the headers of the response as name/value arrays.
Object	getResponseHeaders(headerName)	Gets the headers of the response as name/value arrays.
Number	getStatusCode()	Gets the status code of the response, the list of the possible values is in <code>HTTP_STATUS</code> constants.
String	getStatusReasonPhrase()	Gets the status code's reason phrase.

Methods Details

close()

Needs to be called when not reading content via `getResponseBody` or `getMediaData` to be able to reuse the client.

Returns

[Boolean](#) true if the entity content is consumed and content stream (if exists) is closed

Supported Clients

SmartClient,WebClient,NGClient

Sample

getCharset()

Get the charset of the response body.

Returns

[String](#)

Supported Clients

SmartClient,WebClient,NGClient

Sample

```
var charset = response.getCharset();
```

getException()

Getter for the exception message.

Returns

[String](#) the exception message

Supported Clients

SmartClient,WebClient,NGClient

Sample

```
var exception = response.getException();
```

getMediaData()

Get the content of response as binary data. It also supports gzip-ed content.

Returns

[Array](#)

Supported Clients

SmartClient, WebClient, NGClient

Sample

```
var mediaData = response.getMediaData();
```

getResponseBody()

Get the content of the response as String.

Returns[String](#)**Supported Clients**

SmartClient, WebClient, NGClient

Sample

```
var pageData = response.getResponseBody();
```

getResponseHeaders()

Gets the headers of the response as name/value arrays.

Returns[Object](#)**Supported Clients**

SmartClient, WebClient, NGClient

Sample

```
var allHeaders = response.getResponseHeaders();
var header;

for (header in allHeaders) application.output(header + ': ' + allHeaders[header]);
```

getResponseHeaders(headerName)

Gets the headers of the response as name/value arrays.

Parameters[String](#) headerName;**Returns**[Object](#)**Supported Clients**

SmartClient, WebClient, NGClient

Sample

```
var contentLength = response.getResponseHeaders("Content-Length");
```

getStatusCode()

Gets the status code of the response, the list of the possible values is in HTTP_STATUS constants.

In case there was an exception executing the request, please ignore/do not use this value (it will be 0).
You can check that situation using response.getException().

Returns[Number](#)**Supported Clients**

SmartClient, WebClient, NGClient

Sample

```
var status = response.getStatusCode();// compare with HTTP_STATUS constants
```

getStatusReasonPhrase()

Gets the status code's reason phrase. For example if a response contains status code 403 (Forbidden) it might be useful to know why.

For example a Jenkins API req. could answer with "403 No valid crumb was included in the request" which will let you know that you simply have to request a crumb and then put that in the request headers as "Jenkins-Crumb". But you could not know that from 403 status alone...

Returns

[String](#)

Supported Clients

SmartClient, WebClient, NGClient

Sample

```
var statusReasonPhrase = response.getStatusReasonPhrase();
```