system.util.sendRequest

This feature is new in Ignition version **7.9.3**
Click here to check out the other new features

This function is used in **Python Scripting.**

Description

This function sends a message to the Gateway, working in a similar manner to the sendMessage function, except sendRequest expects a response to the message. To handle received messages, you must set up Gateway Event Script message handlers within a project. These message handlers run Jython code when a message is received. You can then place a return at the end of the code to return something to where the sendRequest was originally called from. You can add message handlers under the "Message" section of the Gateway Event Script configuration dialog.

Client Permission Restrictions

This scripting function has no Client Permission restrictions.

Syntax

system.util.sendRequest(project, messageHandler, payload, remoteServer, timeoutSec)

- Parameters

  String project - The name of the project containing the message handler.

  String messageHandler - The name of the message handler that will fire upon receiving a message.

  PyDictionary payload - Optional. A PyDictionary which will get passed to the message handler. Use "payload" in the message handler to access dictionary variables.

  String hostName - Optional. Limits the message delivery to the client that has the specified network host name.

  String remoteServer - Optional. A string representing a target Gateway Server name. The message will be delivered to the remote Gateway over the Gateway Network. Upon delivery, the message is distributed to the local Gateway and clients as per the other parameters.

  String timeoutSec - Optional. The number of seconds before the sendRequest call times out.

- Returns

  Object - The return from the message handler.

- Scope

  All

Code Examples

Code Snippet

```
# Request sent to the message handler 'test' which then saves the return value to
returnValue and prints it.
returnValue = system.util.sendRequest(project='ACME', messageHandler='test',
payload={'hoursOn':15})
print returnValue
```