

# WS Extensions

## 7.05 connectHTML()

### Purpose

With connectHTML(), you can work with the values from HTML pages rendered using Control() or View() commands.

Using connectHTML() command, you can read the values from the HTML pages, change values, click a link on the HTML page, integrate the display via Control() or View() with the SAP screen or create a separate "floating" window. This command works with your own HTML pages, internet and intranet pages.

### Syntax

```
connectHTML( {options} )
```

For example -

```
connectHTML(click:"submit_login")
```

```
connectHTML({shownnames:true})
```

### Available Options

You can use the following options with the connectHTML:

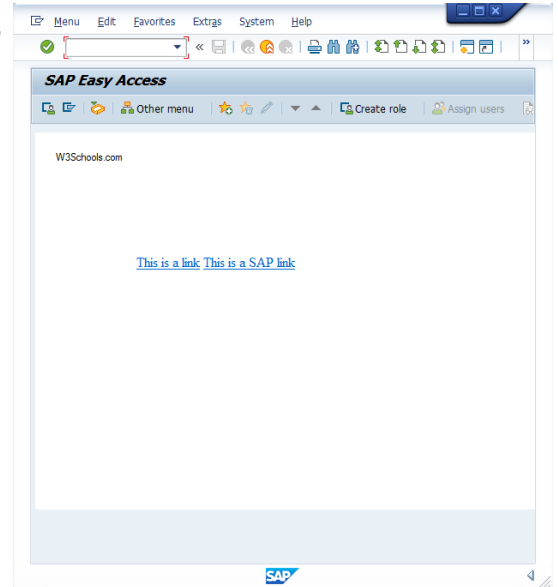
<a href="#">"name": ".."</a>	{control.. name= } - Specification of the control name.
<a href="#">"window": ".."</a>	{View.. returnwindow= statement} - Specification of the window number returned by.
<a href="#">"click": ".."</a>	(click:"Element") - To set the name of the HTML element (e.g. a button, an image or a link) to click.
<a href="#">"seturl": ".."</a>	Displays the HTML page with the given URL.
<a href="#">"gettext": ".."</a>	Return the page title.
<a href="#">"settitle": ".."</a>	Set the new page title.
<a href="#">"listelements": ".."</a>	This option shows all HTML elements of the current page in the given text variable.
<a href="#">"shownames": ".."</a>	This option displays name of each HTML element as a 'tooltip' on the HTML page on mouse hover

### Scenario

# WS Extensions

You can perform click on the HTML page using connectHTML() in the SAP as shown below.

1. Logon to SAP and navigate to SAP Easy Access screen.
2. You can view a message box and click Ok to open the GUIXT website.
3. You can view the GUIXT website opened in SAP with auto-click or auto-submit of a link and the tooltip for the HTML elements on mouse hover using connectHTML().



## Script

### SAPLSMTR\_NAVIGATION.E0100.SJS

```
clearscreen();
if((z_guixtcrm_user == "Username" || z_guixtcrm_user == "USERNAME") &&
z_guixtcrm_pass == "Password")
{
set("V[link_icwc]", "http://mamba/guixtcrm/");
view([0,0],[20,120], "&V[link_icwc]", {name: "z_guixt_html"});
if(htmlExists("text_text"))
{
// tests if "text_text" exists in the HTML page now.
//textbox([0,130],[30,180], {name: "z_longtext"
});
// limitation, longtext cannot display more than 2048 characters at a
time.
set("html[text_text]", z_guixtcrm_user
);
// sets USERNAME z_guixtcrm_user
set("html[password_user_password]", z_guixtcrm_pass
);
// sets PASSWORD z_guixtcrm_pass
set("V[z_test_get_html]", "&html[text_text
]");
```

# WS Extensions

```
// gets USERNAME into variable z_test_get_html
    println("\n\n-----\n\n"+"z_test_get_html is:"+z_test_ge
t_html+"\n\n-----\n\n");
// check if get was successful
    //connectHTML("z_guixt_html",{
click:"submit_login",listelements:"z_longtext"});
        // click on the button and also lists elements inside
the longtext
        connectHTML({click:"submit_login"
});
// click on the button "submit_login"
    }
}
else
{
    set("V[crmuser]","sarvani");
    set("V[crmpass]","guixt100");
    view([0, 0], [20, 120], "test_WS.html", {name:"z_guixt_html", exec
uteJS:"externalFLogon('&V[crmpass]', '&V[crmpass]')", s1ologon:true});

// will execute JAVASCRIPT in html document test_WS.html with params
}
connectHTML({shownames:true});
function single_signon(){
    println("Executing single_signon");
    view([15.5,20.6],[19.5,28.6],"
http://www.guixt.com/",{
"floating":true,"maximize":true, name:"thisisbozopopup"});
    onscreen 'ZGUIXT.0100'
        println("\nExecuting onscreen 1 for single_signon");
        enter();
    onscreen 'ZGUIXT.0100'
        println("\nExecuting onscreen 2 for single_signon");
        enter();
}
}
```

## ELOGON.SJS

```
set("F[User]","Username");
set("F[Password]","Password");
enter();
```

## test\_ws.html

# WS Extensions

//

THE TEST PAGE

[This is a link](#) [This is a SAP link](#)

Unique solution ID: #1680

Author: madhubabu.koti@guixt.com

Last update: 2022-01-06 07:20