

가

Last update: 2023/11/10

가	1
.....	1
.....	1

가

WebSocket / Promise async, await (ECMAScript (ECMA-262))

API async function , await API

API IE

API opensync, sendrecv가

on_open, on_message, on_error, on_close가

: /HTML5/COMPONENT/WEBSOCKET/websocket_sync

- [websocket_sync.xml](#)
- [websocket_sync.js](#)
- [websocket_sync.html](#)

```
// "
function ws_on_open(objInst)
{
    screen.alert("    on_open>    ");
}

// "
function ws_on_close(objInst, nCode, strReason, wasClean)
{
    screen.alert("    on_close>    " + "\n" +
        "code = [" + nCode + "]\n" +
        "reason = [" + strReason + "]\n" +
        "wasClean = " + wasClean);
}

// "
function ws_on_error(objInst)
{
```

```
    screen.alert("        on_error>        ");
}

// "        "
function ws_on_message(objInst, strMessage)
{
    this.fldRecvData.setText(strMessage);
}

// "        "
function btnConnect_on_mouseup(objInst)
{
    this.ws.open(this.fld_websocket_server_addr.getText());
}

// "        "
async function btnConnectSync_on_mouseup(objInst)
{
    var ret;

    // 가        , try/catch
    try {
        // opensync API        "on_open", "on_error", "on_close"
        // opensync API
        //        , true
        factory.consoleprint("opensync> before opensync");
        ret = await
this.ws.opensync(this.fld_websocket_server_addr.getText(), 5000);
        factory.consoleprint("opensync> after opensync, ret = " + ret);

        if (ret == true) {
            factory.consoleprint("opensync> websocket open success");
        }
        else {
            factory.consoleprint("opensync> websocket open fail");
        }
    }
    catch (e) {
        factory.consoleprint("opensync> after opensync, error = " +
e.message);
        screen.alert("opensync> after opensync, error = " + e.message);
    }
}

// "        "
async function btnSendRecv_on_mouseup(objInst)
{
    var recv_data;

    // 가        , try/catch
    try {
```

```
// sendrecv API      "on_message", "on_error"
// ("on_close"      .)
// sendrecv API
factory.consoleprint("sendrecv> before sendrecv");
recv_data = await this.ws.sendrecv(this.fldSendData.getText(),
5000);
factory.consoleprint("sendrecv> after sendrecv");

//
factory.consoleprint("sendrecv> data length = " + recv_data.length);
this.fldRecvData.setText(recv_data);
}
catch (e) {
factory.consoleprint("sendrecv> after sendrecv, error = " +
e.message);
screen.alert("sendrecv> after sendrecv, error = " + e.message);
}
}

// " "
function btnClose_on_mouseup(objInst)
{
this.ws.close();
}

// " "
function btnClear_on_mouseup(objInst)
{
this.fldRecvData.setText("");
}

// " "
function btnSend_on_mouseup(objInst)
{
var ret;

ret = this.ws.send(this.fldSendData.getText());
if (ret == false) {
screen.alert("send fail");
}
}
}
```

From:

<http://technet.softbase.co.kr/wiki/> - **xFrame5 TechNet**

Permanent link:

http://technet.softbase.co.kr/wiki/guide/component/websocket/websocket_sync



Last update: **2023/11/10 14:22**