

UDPCOMM 가

Last update: 2023/11/08

www.softbase.co.kr
Copyright © SOFTBase Inc. All rights reserved.

UDPCOMM 가 1
..... 1
..... 1

UDPCOMM 가

UDP

UDP

UDP /

: /HTML5/COMPONENT/UDPCOMM/udpcomm_basic

- [udpcomm_basic.xml](#)
- [udpcomm_basic.js](#)
- [udpcomm_basic.css](#)

```
function screen_on_load()
{
    fldSendData1.setText("1234ABCD가      1");
    fldSendData2.setText("1234ABCD가      2");
}

function btnInitXPlusUdpSvrComm_on_mouseup(objInst)
{
    //
    udpsvr01.init();
}

function btnInitXPlusUdpCltComm_on_mouseup(objInst)
{
    //
    udpclt01.init();
}

function btnOpenUdpSvrSocket_on_mouseup(objInst)
{
    //
    udpsvr01.open(fldUdpSvrPortNo.getText());
}

function btnOpenUdpCltSocket_on_mouseup(objInst)
{
    //
```

```
    udpclt01.open(fldUdpCltPortNo.getText());
}

function btnCloseUdpSvrSocket_on_mouseup(objInst)
{
    //
    udpsvr01.close();
}

function btnCloseUdpCltSocket_on_mouseup(objInst)
{
    //
    udpclt01.close();
}

function btnSendData1_on_mouseup(objInst)
{
    //
    udpsvr01.send(fldReceiverIpAddr1.getText(),
fldReceiverPortNo1.getText(), fldSendData1.getText());
}

function btnSendData2_on_mouseup(objInst)
{
    //
    udpclt01.send(fldReceiverIpAddr2.getText(),
fldReceiverPortNo2.getText(), fldSendData2.getText());
}

function btnClearData1_on_mouseup(objInst)
{
    //
    fldRecvDataLength1.setText("");
    fldSenderId1.setText("");
    fldSenderPortNo1.setText("");
    fldRecvData1.setText("");
}

function btnClearData2_on_mouseup(objInst)
{
    //
    fldRecvDataLength2.setText("");
    fldSenderId2.setText("");
    fldSenderPortNo2.setText("");
    fldRecvData2.setText("");
}

////////////////////////////////////
////////////////////////////////////
// EVENT
////////////////////////////////////
////////////////////////////////////
```

```
/**
 *
 * @param objInst UDP
 */
function udpsvr01_on_init(objInst)
{
    factory.consoleprint(objInst.getname() + "_on_init>");
    console.log(objInst.getname() + "_on_init>");
}

/**
 * UDP
 * @param objInst UDP
 * @param nLocalPort
 */
function udpsvr01_on_open(objInst, nLocalPort)
{
    factory.consoleprint(objInst.getname() + "_on_open : " + nLocalPort);
    console.log(objInst.getname() + "_on_open : " + nLocalPort);
}

/**
 *
 * @param objInst UDP
 * @param nDataLen
 * @param strData
 * @param strRemoteIP          IP
 * @param nRemotePort          UDP
 */
function udpsvr01_on_recv(objInst, nDataLen, strData, strRemoteIP,
nRemotePort)
{
    factory.consoleprint("nDataLen = " + nDataLen);
    factory.consoleprint("strData = " + strData);

    fldRecvDataLength1.settext(nDataLen);
    fldSenderIpAddr1.settext(strRemoteIP);
    fldSenderPortNo1.settext(nRemotePort);
    fldRecvData1.settext(strData);
}

/**
 *
 * @param objInst UDP
 * @param nErrorCode
 * @param strErrorMsg
 * @param strRemoteIP          IP      (          )
 * @param nRemotePort          UDP      (          0)
 */
function udpsvr01_on_error(objInst, nErrorCode, strErrorMsg, strRemoteIP,
nRemotePort)
```

```
{
    if (strRemoteIP == "") {
        screen.alert(objInst.getname() + "_on_error : " + nErrorCode + ", "
+ strErrorMsg);
    } else {
        screen.alert(objInst.getname() + "_on_error : " + nErrorCode + ", "
+ strErrorMsg + ", " + strRemoteIP + ", " + nRemotePort);
    }
}

/**
 *
 * @param objInst UDP
 */
function udpclt01_on_init(objInst)
{
    factory.consoleprint(objInst.getname() + "_on_init>");
    console.log(objInst.getname() + "_on_init>");
}

/**
 * UDP
 * @param objInst UDP
 * @param nLocalPort
 */
function udpclt01_on_open(objInst, nLocalPort)
{
    factory.consoleprint(objInst.getname() + "_on_open : " + nLocalPort);
    console.log(objInst.getname() + "_on_open : " + nLocalPort);

    fldUdpClntPortNo.setText(nLocalPort);
    fldReceiverPortNo1.setText(nLocalPort);
}

/**
 *
 * @param objInst UDP
 * @param nDataLen
 * @param strData
 * @param strRemoteIP      IP
 * @param nRemotePort      UDP
 */
function udpclt01_on_rcv(objInst, nDataLen, strData, strRemoteIP,
nRemotePort)
{
    factory.consoleprint("nDataLen = " + nDataLen);
    factory.consoleprint("strData = " + strData);

    fldRecvDataLength2.setText(nDataLen);
    fldSenderIdAddr2.setText(strRemoteIP);
    fldSenderIdPortNo2.setText(nRemotePort);
}
```

```
fldRecvData2.setText(strData);
}

/**
 *
 * @param objInst UDP
 * @param nErrorCode
 * @param strErrorMsg
 * @param strRemoteIP IP ( )
 * @param nRemotePort UDP ( 0)
 */
function udpclt01_on_error(objInst, nErrorCode, strErrorMsg, strRemoteIP,
nRemotePort)
{
    if (strRemoteIP != "") {
        //
        screen.alert(objInst.getname() + "_on_error : " + nErrorCode + ", "
+ strErrorMsg + ", " + strRemoteIP + ", " + nRemotePort);
    } else {
        // ( , API )
        screen.alert(objInst.getname() + "_on_error : " + nErrorCode + ", "
+ strErrorMsg);
    }
}
}
```

From:

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

Permanent link:

http://technet.softbase.co.kr/wiki/guide/component/udpcomm/udpcomm_basic 

Last update: **2023/11/08 10:13**