

가

Last update: 2024/07/15

가	1
.....	1
.....	1

가

FileUpload

가 .

: /HTML5/COMPONENT/FILEUPLOADER/fileuploader_imageresize

- [fileuploader_imageresize.xml](#)
- [fileuploader_imageresize.js](#)
- 

```
//          \technet\project\template\ext\java\fileuploader.jsp.txt
fileuploader.jsp
//          WAS          ,          URL          base_url

var base_url = "http://127.0.0.1:8080/xframe5/fileuploader.jsp";

function screen_on_load()
{
    //          URL
    uploader_basic.seturl(base_url);
}

//          가
function btn_addfile_on_mouseup(objInst)
{
    uploader_basic.addfile();
}

//
function btn_deleteallfile_on_mouseup(objInst)
{
    //
    uploader_basic.deleteallfile();
}

//
```

```
function btn_clearallfilestatus_on_mouseup(objInst)
{
    //
    uploader_basic.clearallfilestatus();

    //
    UpdateFileStauts(-1);
}

//
function btn_startupload_on_mouseup(objInst)
{
    var i, file_count, file_object;

    //
    //
    //          (setfiledata)
    file_count = uploader_basic.getfilecount();
    for (i = 0; i < file_count; i++) {
        file_object = uploader_basic.getfileobject(i);

        //
        uploader_basic.setfileuserdata(i, "1");

        //
        if (factory.isimagefileobject(file_object)) {
            // TODO:          4K
            if (file_object.size > 4096) {
                //
                uploader_basic.setfileuserdata(i, "0");
            }
        }
    }
}

//
ReduceImageFileSize();
}

//
function ReduceImageFileSize()
{
    var i, file_count;

    file_count = uploader_basic.getfilecount();
    for (i = 0; i < file_count; i++) {
        //
        if (uploader_basic.getfileuserdata(i) == "1") {
            continue;
        }

        //
        img_resize.setimageobject(uploader_basic.getfileobject(i));
        img_resize.setuserdata(uploader_basic.getfilename(i));
    }
}
```

```
        return;
    }

    //
    ReloadFileStatus();

    //
    uploader_basic.startupload();
}

function img_resize_on_load(objInst, nImageWidth, nImageHeight)
{
    var i, count, reduce_width, reduce_height, file_name, file_object;

    //
    file_name = this.img_resize.getuserdata();
    if (!file_name) { return; }

    file_object = this.img_resize.getimagefileobject(file_name, 1);
    factory.consoleprint("before> " + file_object.name + ", size = " +
file_object.size);

    // 50%          (TODO:          )
    reduce_width = parseInt(nImageWidth * 0.5);
    reduce_height = parseInt(nImageHeight * 0.5);
    objInst.setsize(reduce_width, reduce_height);

    //
    file_object = this.img_resize.getimagefileobject(file_name, 2,
reduce_width, reduce_height);
    factory.consoleprint("after> " + file_object.name + ", size = " +
file_object.size);

    file_count = uploader_basic.getfilecount();
    for (i = 0; i < file_count; i++) {
        //
        if (uploader_basic.getfileuserdata(i) == "1") {
            continue;
        }

        //
        uploader_basic.setfileobject(i, file_object);

        //
        uploader_basic.setfileuserdata(i, "1");

        //
        ReduceImageFileSize();
    }
}

//
```

```
function btn_stopupload_on_mouseup(objInst)
{
    uploader_basic.stopupload();
}

//
function UpdateFileStauts(nFileIndex)
{
    var count, i;

    //      가 -1      ,
    if (nFileIndex == -1) {
        count = uploader_basic.getfilecount();
        for (i = 0; i < count; i++) {
            UpdateOneFileStauts(i);
        }
    }
    //      가 -1      ,
    else {
        UpdateOneFileStauts(nFileIndex);
    }
}

//
function ReloadFileStatus()
{
    var nFileIndex, count;

    //
    grdList.deleteall();

    //      Loop      가
    count = uploader_basic.getfilecount();
    for (nFileIndex = 0; nFileIndex < count; nFileIndex++) {
        grdList.additem();

        grdList.setitemtext(nFileIndex, 0,
uploader_basic.getfilename(nFileIndex));
        grdList.setitemtext(nFileIndex, 1,
uploader_basic.getfilesize(nFileIndex));
        grdList.setitemtext(nFileIndex, 2,
uploader_basic.getfilebriefsize(nFileIndex));
        grdList.setitemtext(nFileIndex, 3,
uploader_basic.getfiledate(nFileIndex));
        grdList.setitemtext(nFileIndex, 4,
uploader_basic.getfiletime(nFileIndex));

        UpdateOneFileStauts(nFileIndex);
    }
}

//
```

```

function UpdateOneFileStauts(nFileIndex) {
    grdList.setitemtext(nFileIndex, 5,
uploader_basic.getfilestatus(nFileIndex));
    grdList.setitemtext(nFileIndex, 6,
uploader_basic.getfileprogress(nFileIndex));
    grdList.setitemtext(nFileIndex, 7,
uploader_basic.getfileresult(nFileIndex));
    grdList.setitemtext(nFileIndex, 8,
uploader_basic.getfileresultmsg(nFileIndex));
    grdList.setitemtext(nFileIndex, 9,
uploader_basic.getfileresultfilename(nFileIndex));
}

////////////////////////////////////
////////////////////////////////////
// EVENT
////////////////////////////////////
////////////////////////////////////

//          (          Drag&Drop          )
function grdList_on_dropfiles(objInst, arrayDropFiles, nDropFileCount)
{
    var    i, fileObj;

    //
    factory.consoleprint("nDropFileCount = " + nDropFileCount);
    for (i = 0; i < nDropFileCount; i++) {
        fileObj = arrayDropFiles[i];
        factory.consoleprint(i + " : fileObj.name = " + fileObj.name);
        factory.consoleprint(i + " : fileObj.size = " + fileObj.size);
    }

    //          가
    uploader_basic.addfileobjectarray(arrayDropFiles);
}

//
function uploader_basic_on_fileprogress(objInst, nFileIndex, strFileName,
nPos)
{
    //
    grdList.setitemtext(nFileIndex, 6, nPos);
}

//
function uploader_basic_on_filecomplete(objInst, nFileIndex, strFileName)
{
    //
    UpdateOneFileStauts(nFileIndex);
}

//

```

```
function uploader_basic_on_listupdate(objInst)
{
    //
    ReloadFileStatus();
}
```

From:

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

Permanent link:

https://technet.softbase.co.kr/wiki/guide/component/fileuploader/fileuploader_imageresize 

Last update: **2024/07/15 17:42**