------------------------------------------------------------------------------------------------------------------------------------------

@echo off

NOTEPAD /P E:\\Practice\PrintDemo\PrintDemo\test.txt

NOTEPAD /P E:\\Practice\PrintDemo\PrintDemo\test.txt

------------------------------------------------------------------------------------------------------------------------------------------

@echo off

print /d:\\inhydprintsrv01\PrinterZone\_1 "E:\\Practice\PrintDemo\PrintDemo\AFE Find Default Behavior.docx"

------------------------------------------------------------------------------------------------------------------------------------------

@echo off

RUNDLL32 PRINTUI.DLL,PrintUIEntry /y /n \\inhydprintsrv01\PrinterZone\_1

pushd "E:\Practice\PrintFilesDi"

for %%a in (\*.pdf \*.doc) do (

call E:\Practice\PrintDemo\PrintDemo\printjs.bat "%%~fa"

)

for %%a in (\*.html) do (

RUNDLL32.EXE MSHTML.DLL,PrintHTML "%%~fa"

)

Pause

----------------------------------------------------PRINT JS BATCH CODE--------------------------------------------------------

@if (@X)==(@Y) @end /\* JScript comment

@echo off

cscript //E:JScript //nologo "%~f0" %\*

exit /b %errorlevel%

@if (@X)==(@Y) @end JScript comment \*/

//gets an information that normally is acquired by right click-details

// can get image dimensions , media file play time and etc.

//////

FSOObj = new ActiveXObject("Scripting.FileSystemObject");

var ARGS = WScript.Arguments;

if (ARGS.Length < 1 ) {

WScript.Echo("No file passed");

WScript.Echo("Usage (prints with default printer a file if possible)");

WScript.Echo(WScript.ScriptName + " file");

WScript.Quit(1);

}

var filename=ARGS.Item(0);

var objShell=new ActiveXObject("Shell.Application");

/////

//fso

ExistsItem = function (path) {

return FSOObj.FolderExists(path)||FSOObj.FileExists(path);

}

getFullPath = function (path) {

return FSOObj.GetAbsolutePathName(path);

}

//

//paths

getParent = function(path){

var splitted=path.split("\\");

var result="";

for (var s=0;s<splitted.length-1;s++){

if (s==0) {

result=splitted[s];

} else {

result=result+"\\"+splitted[s];

}

}

return result;

}

getName = function(path){

var splitted=path.split("\\");

return splitted[splitted.length-1];

}

//

function main(){

if (!ExistsItem(filename)) {

WScript.Echo(filename + " does not exist");

WScript.Quit(2);

}

var fullFilename=getFullPath(filename);

var namespace=getParent(fullFilename);

var name=getName(fullFilename);

var objFolder=objShell.NameSpace(namespace);

var objItem=objFolder.ParseName(name);

//https://msdn.microsoft.com/en-us/library/windows/desktop/bb787870(v=vs.85).aspx

//https://msdn.microsoft.com/en-us/library/windows/desktop/bb774170(v=vs.85).aspx

//WScript.Echo(fullFilename + " : ");

//WScript.Echo(objFolder.GetDetailsOf(objItem,-1));

var verbs=objItem.Verbs();

objItem.InvokeVerb("Print");

/\*

Language dependent version commented bellow.

InovkeVerb("Print") cannot check if the print action is supported, but works independently of language settings

\*/

/\*

if (verbs != null) {

//WScript.Echo(verbs.Count);

for (var i=0;i<verbs.Count;i++){

if(verbs.Item(i).Name === "&Print"){

verbs.Item(i).DoIt();

return;

}

//WScript.Echo(verbs.Item(i).Name);

}

WScript.Echo("the item " + filename + "does not support print verb");

return;

}

\*/

}

main();