ExtCmmon NexaCro17 / API Guide

**Nexacro 17**

**Runtime 에서 만 사용 가능 합니다**



| 파일버전 | 작성자 | 작성일 | | 검토 및 승인 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PM | 일시 | | 검수인 | | 일시 | |
| **2017.5.11.1** | **안성현** | **2017.05.11** | |  |  | |  | |  | |
|  | | * ExtCommonV17 작성 및 배포. | | | | | | | | |
| 2017.12.04.1 | 안성현 | 2017.12.04 | |  |  | |  | |  | |
|  | | * 2017년 4월 엔진부터 v8 엔진이 Version 5 로 업데이트 됨으로 인해, 2017년 4월 엔진 부터, 해당 [2017.12.04.1] ExtCommon을 사용하여야 합니다. | | | | | | | | |
| 2018년 02 월 정기 버전부터 V8 java script 엔진( V8.DLL ) 버전이  Version 5 >>> Version 6 으로 업데이트 | | | | | | | | | | |
| V8.dll 이 업데이트 됨 / 개발 Tool VS2017 변경. ( 하위버전 호환 안됨. ) | | | | | | | | | | |
| 2018.03.15.1 | 안성현 | 2018.03.15 | |  |  | |  | |  | |
|  | | * Nexacro 17 Platform v8 엔진이 업데이트 되어 ExtCommon17.dll 업데이트 됨으로 각각의 프로세스 버전에 맞게 사용 하여야 함. * 32bit / 64bit / xp 용으로 각각의 모듈 출력 * DLL 파일명이 변경 되었습니다.   + **XP** 용 : **ExtCommonV17\_XP**.dll   + **32Bit** 용 : **ExtCommonV17\_32**.dll   + **64Bit** 용 : **ExtCommonV17\_64**.dll | | | | | | | | |
| 2018.04.09.01 | 안성현 | 2018.04.09 | |  |  | |  | |  | |
|  | | * \_ext\_isWow64 함수 사용 시,   + 64bit 컴포넌트에서 체크 하지 못하는 현상 수정. | | | | | | | | |
| 2018.05.10.1 | 안성현 | 2018.05.10 | |  |  | |  | |  | |
|  | | * \_ext\_excuteProcess 함수 수정.   + 64bit Nexacro 프로세스 사용 시 실행되지 않는 문제 수정.   + 32bit Nexacro 프로세스 가 64bit OS 에서 사용하기 위해, Redirection 기능을 추가혔는데, 이 부분이 충돌나서 발생. | | | | | | | | |
| 2018.06.11 | 안성현 | 2018.06.11 | |  |  | |  | |  | |
|  | | * \_ext\_getIPAdapterInfo 함수 추가. | | | | | | | | |
| 2018.10.08 | 안성현 | 2018.10.08 | |  |  | |  | |  | |
|  | | * \_ext\_excuteProcess 함수 수정.   특정 경로에 있는 프로세스를 실행 할 때 사용.  (프로세스의 Path)  Default : “” ( empty ) | | | | | | | | |
| 2018.10.15.3 | 안성현 | 2018.10.15 | |  |  | |  | |  | |
|  | | * \_ext\_getIPAdapterInfo 함수 수정.   + OS : Windows XP 에서 WS2\_32.DLL 버전 호환되지 않아.   오류 내용:  [ 프로시저 시작 지점 InetNtopW을(를) DLL WS2\_32.dll에서 찾을 수 없습니다. ]   * Windows XP 에서는 IPV6 값을 취득 할 수 없습니다.   출력 되는 값은 “Windows XP Not Support”   * \_ext\_EmptyWorkingSet   + Windows XP 일 경우 프로시저 오류 발생.   [ 컴파일 옵션 추가 : 전처리기 PSAPI\_VERSION=1 ] | | | | | | | | |
| 2018.10.25.1 | 안성현 | 2018.10.25 | |  |  | |  | |  | |
|  | | * \_ext\_excuteProcess, \_ext\_excuteAppPostion   OS 64bit 에서 프로세스 실행 false 리턴 된 후 windows media player를 웹 페이지에서 실행 시 대기 상태 빠짐 현상. 수정. (컴포넌트 내부에서 redirect 부분 수정.) | | | | | | | | |
| 2018.10.30.1 | 안성현 | 2018.10.30 |  | | |  | |  | |  |
|  | | * \_ext\_getPrimaryHDDSerial 물리 하드 디스크 정보 취득 함수 추가. | | | | | | | | |
| 2018.12.13.01 | 권성근 | **2018.12.13.01** |  | | |  | |  | |  |
|  | | * \_ext\_excuteProcess, \_ext\_excuteAppPostion, \_ext\_winExec   + **2018.11.20일 이후로 보안정책에 의해서 지원안함.** | | | | | | | | |
|  |  |  |  | | |  | |  | |  |
|  | |  | | | | | | | | |

목 차

[ExtCmmon NexaCro17 / API Guide 1](#_Toc528690448)

[파일버전 2](#_Toc528690449)

[작성자 2](#_Toc528690450)

[작성일 2](#_Toc528690451)

[검토 및 승인 2](#_Toc528690452)

[목 차 4](#_Toc528690453)

[NexaCro17 / Project Script 에서 ExtCommonV17.dll 등록 9](#_Toc528690454)

[**Method Name : \_ext\_messagebox(message);** 11](#_Toc528690455)

[**Method Name : \_ext\_sum** 12](#_Toc528690456)

[**Method Name : \_ext\_strcat** 13](#_Toc528690457)

[**Method Name : \_ext\_setRegValue** 14](#_Toc528690458)

[**Method Name : \_ext\_getRegValue** 15](#_Toc528690459)

[**Method Name : \_ext\_getRegNameList** 16](#_Toc528690460)

[**Method Name : \_ext\_getRegValueList** 17](#_Toc528690461)

[**Method Name : \_ext\_setTopWindows** 18](#_Toc528690462)

[**Method Name : \_ext\_getFindWindows** 19](#_Toc528690463)

[**Method Name :** **\_ext\_windowSendMSG** 20](#_Toc528690464)

[**Method Name :** **\_ext\_delDirectory** 21](#_Toc528690465)

[**Method Name : \_ext\_readFile** 22](#_Toc528690466)

[**Method Name : \_ext\_writeFile** 23](#_Toc528690467)

[**Method Name : \_ext\_reNameFile** 25](#_Toc528690468)

[**Method Name : \_ext\_copyFile** 26](#_Toc528690469)

[**Method Name : \_ext\_excuteAppPostion** 27](#_Toc528690470)

[**Method Name : \_ext\_moveFile** 28](#_Toc528690471)

[**Method Name : \_ext\_deleteFile** 29](#_Toc528690472)

[**Method Name : \_ext\_excuteProcess** 30](#_Toc528690473)

[**Method Name : \_ext\_killProcessAll** 32](#_Toc528690474)

[**Method Name : \_ext\_killProcessID** 33](#_Toc528690475)

[**Method Name : \_ext\_getCurrentProcessID** 34](#_Toc528690476)

[**Method Name : \_ext\_getProcessNameHandle** 35](#_Toc528690477)

[**Method Name : \_ext\_getSystemDir** 36](#_Toc528690478)

[**Method Name : \_ext\_setTime** 43](#_Toc528690479)

[**Method Name : \_ext\_getTime** 44](#_Toc528690480)

[**Method Name : \_ext\_getIPAddress** 45](#_Toc528690481)

[**Method Name : \_ext\_getMACAddress** 46](#_Toc528690482)

[**Method Name : \_ext\_getPrintList** 47](#_Toc528690483)

[**Method Name : \_ext\_setDefaultPrinter** 48](#_Toc528690484)

[**Method Name : \_ext\_getDefaultPrinter** 49](#_Toc528690485)

[**Method Name : \_ext\_setPageOption** 50](#_Toc528690486)

[**Method Name : \_ext\_setPrintOption** 51](#_Toc528690487)

[**Method Name : .\_ext\_selectFolderPath** 52](#_Toc528690488)

[**Method Name : \_ext\_getFileList** 53](#_Toc528690489)

[**Method Name : \_ext\_getKeyCheck** 54](#_Toc528690490)

[**Method Name : \_ext\_getHostName** 55](#_Toc528690491)

[**Method Name : \_ext\_getUserName** 56](#_Toc528690492)

[**Method Name : \_ext\_getFileDateInfo** 57](#_Toc528690493)

[**Method Name : \_ext\_getCheckDir** 58](#_Toc528690494)

[**Method Name : \_ext\_getSystemEnv** 59](#_Toc528690495)

[**Method Name : \_ext\_getDriveInfo** 60](#_Toc528690496)

[**Method Name : \_ext\_getDriveInfo2** 61](#_Toc528690497)

[**Method Name : \_ext\_getFontInfo** 62](#_Toc528690498)

[**Method Name : \_ext\_writeUProfile** 63](#_Toc528690499)

[**Method Name : \_ext\_readUProfile** 64](#_Toc528690500)

[**Method Name : \_ext\_getAppPath** 65](#_Toc528690501)

[**Method Name : \_ext\_getTraySizeInfo** 66](#_Toc528690502)

[**Method Name : \_ext\_getDesktopWorkArea** 67](#_Toc528690503)

[**Method Name : \_ext\_setSecuritySite(strStie)** 68](#_Toc528690504)

[**Method Name : \_ext\_EmptyWorkingSet** 69](#_Toc528690505)

[**Method Name : \_ext\_setBeep** 70](#_Toc528690506)

[**Method Name : .\_ext\_isWow64** 71](#_Toc528690507)

[**Method Name : \_ext\_CreateDirectory** 72](#_Toc528690508)

[**Method Name : \_ext\_setInternetSetCookie** 73](#_Toc528690509)

[**Method Name : \_ext\_getBitCalc** 74](#_Toc528690510)

[**Method Name : MMFMaping.\_ext\_MmfFileWrite** 75](#_Toc528690511)

[**Method Name : MMFMaping.\_ext\_MmfFileClose()** 77](#_Toc528690512)

[**Method Name : \_ext\_getFileInfo (strPath)** 78](#_Toc528690513)

[**Method Name : \_ext\_getIMEmode()** 80](#_Toc528690514)

[**Method Name : \_ext\_setIMEmode(bImeFlag)** 81](#_Toc528690515)

[**Method Name : \_ext\_msgBox (strTitle, strMessage, strStyle)** 82](#_Toc528690516)

[**Method Name : \_ext\_getInternetConnectState()** 85](#_Toc528690517)

[**Method Name : \_ext\_setOleServerBusyTimeOut(strReg, iTime, bEnable, bBusyDialog )** 86](#_Toc528690518)

[**Method Name : \_ext\_getInstallProg(strProgidORclsid))** 88](#_Toc528690519)

[**Method Name : \_ext\_setShowCursor(bCursor)** 89](#_Toc528690520)

[**Method Name :** **\_ext\_getNumFormatLoc(strNumberData, intLCID/LCID, intFmt);** 90](#_Toc528690521)

[**Method Name : \_ext\_getDateUTC (intYear, intMon, intDay, intHour, intMin, intSec)** 92](#_Toc528690522)

[**Method Name : \_ext\_setOutputdebugstring (strMsg)** 93](#_Toc528690523)

[**Method Name : \_ext\_getMaxMonitorRect ()** 94](#_Toc528690524)

[**Method Name : \_ext\_getMonitorInfo()** 95](#_Toc528690525)

[**Method Name : \_ext\_getGetKeyState (strKey(intKey))** 97](#_Toc528690526)

[**Method Name : \_ext\_getFindProcessInfo(strProcessName)** 103](#_Toc528690527)

[**Method Name : \_ext\_getFindAllWinHwnd(strTitle);** 105](#_Toc528690528)

[**Method Name : \_ext\_getNumLocaleInfo(strLCID/intLCID,intFmt)** 106](#_Toc528690529)

[**Method Name : \_ext\_winExec(strCommand, intStateFlg)** 108](#_Toc528690530)

[**Method Name : \_ext\_getLocaleFormat(strTitle);** 110](#_Toc528690531)

[**Method Name : \_ext\_getDefaultLocaleInfo(strType)** 133](#_Toc528690532)

[**Method Name : \_ext\_setURLencode(strEncode, strType)** 134](#_Toc528690533)

[**Method Name : \_ext\_getURLdecode (strDecode, strType)** 136](#_Toc528690534)

[**Method Name : \_ext\_setBase64encode(strEncode, strType)** 138](#_Toc528690535)

[**Method Name : \_ext\_getBase64decode (strEncode, strType)** 139](#_Toc528690536)

[**Method Name : \_ext\_existFile (string filepath)** 140](#_Toc528690537)

[**Method Name : \_ext\_setPrintOrientation (int opt)** 141](#_Toc528690538)

[**Method Name : \_ext\_getFindMiplatformOnlyOne(string Pojectkey, string sitekey, string MiplatformProcName)** 142](#_Toc528690539)

[**Method Name : \_ext\_getLengthB( strData, strCodePage)** 144](#_Toc528690540)

[**Method Name : \_ext\_getGlobalMemoryStatusEx()** 145](#_Toc528690541)

[**Method Name : \_ext\_getProcessMemoryInfo ()** 147](#_Toc528690542)

[**Method Name : \_ext\_getUseFont(szFont)** 149](#_Toc528690543)

[**Method Name : \_ext\_getComPortList()** 151](#_Toc528690544)

[**Method Name : \_ext\_getCurrentProcessCPU()** 153](#_Toc528690545)

[**Method Name : \_ext\_getProcessModule(nProcessID)** 154](#_Toc528690546)

[**Method Name : \_ext\_getIPAdapterInfo** 156](#_Toc528690547)

[**Method Name : \_ext\_getPrimaryHDDSerial(nNumberDisk)** 158](#_Toc528690548)

# NexaCro17 / Project Script 에서 ExtCommonV17.dll 등록

* **Runtime 에서 만 사용 가능 합니다 -**

ExtCommonV17 API 모듈을 사용하려면 아래의 파일을 필요로 합니다.

| 파일명 | 설명 |
| --- | --- |
| **ExtCommonV17.dll** | **ExtCommonV17 API 모듈** |

[표 1‑2] 모듈 구성파일

* + **등록 및 파일 로드.**

**this**.**objExtCommon** = {}; //object 명.

**this**.**strPATH** = **"D://ExtCommonV17/ ExtCommonV17.dll";**

* **NexaCro 의 경우. ( Nexacro14 와 등록 방법이 다릅니다.)**

**this.objExtCommon = nexacro.\_addExtensionModule**(**this.strPATH**);

확장 모듈 로딩 시 사용하는 API 입니다.

| **Argument** | **설명** |
| --- | --- |
| **strPATH** | 확장 모듈의 전체 경로를 지정합니다. |

* + **등록 해제**
* **NexaCro 의 경우**

**nexacro.\_clearExtensionModule**(**this.strPATH**);

확장 모듈을 더 이상 사용하지 않고 로딩된 모듈을 내려야 할 때 사용하는 API 입니다.

| **Argument** | **설명** |
| --- | --- |
| modulepath | UnLoad 하려는 확장 모듈의 전체 경로를 지정합니다.  지원하는 Path Alias를 사용할 수 있습니다.   * 경로를 입력 하지 않으면, 확장 모듈로 등록 되어 진 모듈들을 모두들을 모두 Unload 합니다. |

**Method Name : \_ext\_messagebox(message);**

▶ **Description**

* + String data(message) 를 Win32Api MessageBox 로 뛰운다.

▶ **Syntax**

var message = "HELLO Nexacro V17";

**this.objExtCommon.\_ext\_messagebox**(message);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| message | string | MessageBox 로 발생 할 string Data.. |

▶ **Return.**

없음.

**Method Name : \_ext\_sum**

▶ **Description**

* + 입력 된 Prameter 의 합을 구하는 함수..

▶ **Syntax**

var arg1 = 1;

var arg2 = 2;

var sum = **this.objExtCommon.\_ext\_sum**(arg1, arg2);

trace(sum);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| arg1 | integer | 합을 구하려는 integer 형 |
| Arg2 | Integer | 합을 구하려는 integer 형 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Integer | 입력된 두수의 합. |

**Method Name : \_ext\_strcat**

▶ **Description**

* + 입력 된 String Parameter 들을 연결하는 합수.
  + [ **API** Object 에 등록 되어 있음. ]

▶ **Syntax**

var arg1 = "안녕";

var arg2 = "HELLO";

var sum = **this.objExtCommon.\_ext\_strcat**(arg1, arg2);

trace(sum);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| arg1 | String | 연결하려는 string 형 문자. |
| arg2 | String | 연결하려는 string 형 문자. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | 두 parameter 의 연결된 문자열. |

**Method Name : \_ext\_setRegValue**

▶ **Description**

* + 해당 위치의 레지스트리 값을 등록 합니다.

▶ **Syntax**

var ret;

var strKey = "HKEY\_CURRENT\_USER";

var str Path = "Software\\AppDataLow\\Software\\TOBESOFT\\";

var strName = "setVal";

var strValue = "HELLO\_안녕/"; //[ integer / string ]

ret = **this.objExtCommon.\_ext\_setRegValue**(strKey, strPath, strName, strValue, "String");

trace("\_ext\_setRegValue : "+ret);

ret = **this.objExtCommon.\_ext\_setRegValue**(strKey, strPath, "IntegerValue", 1234, "DWORD");

trace("\_ext\_setRegValue : "+ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| strKey | string | String 메인 key.  HKEY\_CLASSES\_ROOT  HKEY\_CURRENT\_USER  HKEY\_LOCAL\_MACHINE  HKEY\_USERS |
| strPath | string | Key 이하의 경로. |
| strName | string | 기록할 name |
| strValue | String / Integer | String : 기록 할 값.  Integer : 기록하려는 값 |
| valueType | String | 기록하려는 값의 Type 지정.  String : “String”  Integer : “String” 이외의 문자열은 모두 integer type |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | TRUE : 성공.  FALSE : 실패 |

**Method Name : \_ext\_getRegValue**

▶ **Description**

* + 해당 위치의 레지스트리 값을 읽어 옵니다.

▶ **Syntax**

var ret;

var strKey = "HKEY\_CURRENT\_USER";

var strPath = "Software\\AppDataLow\\Software\\TOBESOFT\\";

var strName = "setVal";

ret = **this.objExtCommon.\_ext\_getRegValue**(strKey, strPath, strName, "String");

trace("\_ext\_setRegValue : "+ret);

ret = **this.objExtCommon.\_ext\_getRegValue**(strKey, strPath, "IntegerValue", "DWORD");

trace("\_ext\_setRegValue : "+ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| strKey | string | String 메인 key.  HKEY\_CLASSES\_ROOT  HKEY\_CURRENT\_USER  HKEY\_LOCAL\_MACHINE  HKEY\_USERS |
| strPath | string | Key 이하의 경로. |
| strName | string | 읽어오려는 name |
| valueType | String | 읽어 오려는 값의 Type 지정.  String : “String”  Integer : “String” 이외의 문자열은 모두 integer type |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| String / Integer | 레지스트리에서 해당 위치의 값.. |

**Method Name : \_ext\_getRegNameList**

▶ **Description**

* + 해당 위치의 레지스트리 하위 폴더 이름들을 읽어 옵니다.

▶ **Syntax**

var strKey = "HKEY\_CURRENT\_USER";

var strPath = "Software\\AppDataLow\\Software\\TOBESOFT\\";

ret = **this.objExtCommon.\_ext\_getRegNameList**(strKey, strPath);

trace("\_ext\_getRegNameList : "+ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| strKey | string | String 메인 key.  HKEY\_CLASSES\_ROOT  HKEY\_CURRENT\_USER  HKEY\_LOCAL\_MACHINE  HKEY\_USERS |
| strPath | string | Key 이하의 경로. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| arrray | 레지스트리에서 해당 위치의 하위 폴더 이름들을 리턴 합니다... |

**Method Name : \_ext\_getRegValueList**

▶ **Description**

* + 해당 위치의 레지스트리 하위 폴더 value 이름 들을 읽어 옵니다.

▶ **Syntax**

var strKey = "HKEY\_CURRENT\_USER";

var strPath = "Software\\AppDataLow\\Software\\TOBESOFT\\";

ret = **this.objExtCommon.\_ext\_getRegValueList**(strKey, strPath);

trace("\_ext\_getRegValueList : "+ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| strKey | string | String 메인 key.  HKEY\_CLASSES\_ROOT  HKEY\_CURRENT\_USER  HKEY\_LOCAL\_MACHINE  HKEY\_USERS |
| strPath | string | Key 이하의 경로. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| arrray | 레지스트리에서 해당 위치의 하위 value 이름들을 리턴 합니다... |

**Method Name : \_ext\_setTopWindows**

▶ **Description**

* + 현대 Form을 topmost 에 위치합니다. (최상위 설정.).

▶ **Syntax**

var ret = false;

ret = **this.objExtCommon.\_ext\_setTopWindows(true);**

trace( "\_ext\_setTopWindows : " + ret);

ret = **this.objExtCommon.\_ext\_setTopWindows(false);**

trace( "\_ext\_setTopWindows : " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| **true** | BOOL | TRUE : 최상위 설정.  FALSE : 설정 해제. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | TRUE : 설정 성공.  FALSE : 설정 실패.( 해당 윈도우 헨들을 찾을 수 없습니다. ) |

**Method Name : \_ext\_getFindWindows**

▶ **Description**

* + 검색 되어진 프로그램의 핸들을 ( DWORD 형) 리턴 합니다.

▶ **Syntax**

var ret = false;

ret = **this.objExtCommon.\_ext\_getFindWindows**("","Q-DIR 5.60",false);

trace( "\_ext\_setTopWindows : " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| classnaem | string | 찾으려는 프로그램의 class name. |
| Titlename | string | 찾으려는 프로그램의 title name |
| bTop | bool | 프로그램을 검색 되었다면, window top 으로 설정유무.  True : top 으로 설정.  False : top 으로 설정 하지 않음. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| integer | 검색 되어진 프로그램의 Handle ( 핸들 )을 리턴 합니다. |

**Method Name :** **\_ext\_windowSendMSG**

▶ **Description**

* + 검색 되어진 프로그램으로 copy data sendmessage 로 메시지를 보냅니다.

▶ **Syntax**

var ret = false;

ret = **this.objExtCommon.\_ext\_windowSendMSG**("","Q-DIR 5.60",123,"check");

trace( "\_ext\_windowSendMSG : " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| classnaem | string | 찾으려는 프로그램의 class name. |
| Titlename | string | 찾으려는 프로그램의 title name |
| flag | integer | 보내려는 윈도우 에서 의 식별값 |
| copyData | string | 전달 되어야 할 copydata |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| integer | 검색 되어진 프로그램의 Handle ( 핸들 )을 리턴 합니다. |

**Method Name :** **\_ext\_delDirectory**

▶ **Description**

* + 디렉터리 의 모든 파일을 삭제 할 수 있습니다.
  + 해당 피시에 보안 프로그램 및 폴더 및 파일 의 보안설정이 있을 시 삭제 되지 않을 수 있습니다.

▶ **Syntax**

var ret = false;

ret = **this.objExtCommon.\_ext\_delDirectory**("C:/TEMP/test",0);

trace( "\_ext\_delDirectory : " + ret);

ret = **this.objExtCommon.\_ext\_delDirectory**("C:/TEMP/test",1);

trace( "\_ext\_delDirectory : " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| path | string | directory path: 삭제하려는 디렉 토리 경로. |
| flag | integer | integer :  0 파일만 삭제 ,  1 디렉터리 모두 삭제 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| bool | true 성공  false : 실패 |

**Method Name : \_ext\_readFile**

▶ **Description**

* + 파일을 읽어 string 으로 리턴 합니다.
  + **주의**
    - 50mb 이상 데이터를 한번에 처리 시 오류 발생.

( 대용량 파일 사용 시 Nexacro api “VirtualFile.readLine” 함수 이용.)

▶ **Syntax**

ret = **this.objExtCommon**.\_ext\_readFile("D:/TEMP/ UTF8.txt");

trace("\_ext\_readFile : " + ret);

trace("-------------------------------------------");

ret = **this.objExtCommon**.\_ext\_readFile("D:/TEMP/ansi.txt");

trace("\_ext\_readFile : " + ret);

trace("-------------------------------------------");

ret = **this.objExtCommon**.\_ext\_readFile("D:/TEMP/NotBomUtf8.txt");

trace("\_ext\_readFile : " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| path | string | directory path: 읽으려는 파일 경로. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | 읽은 파일의 data |

* + **오류 발생 Return Data**

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | File open 시 오류 메시지. //2015.11.24 추가  [ Failed \_ext\_readFile file open : + 오류 발생 한 파일 명 ] |

**Method Name : \_ext\_writeFile**

▶ **Description**

* + Parameter 로 전달된 string 데이터를 이용하여 파일을 생성 합니다..

▶ **Syntax**

var ret;

ret = **this.objExtCommon**.\_ext\_readFile("D:/TEMP/ANSI.txt");

trace("\_ext\_readFile : " + ret);

var strFormat = “euc-kr”;

ret = **this.objExtCommon.\_ext\_writeFile**("C:/TEMP/test/euc\_kr.txt",ret, strFormat);

trace("\_ext\_writeFile : " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| path | string | directory path: 읽으려는 파일 경로. |
| data | string | 생성하려는 파일의 data. |
| strFormat | string | [2015.2.1 추가]  Default : BOM 없는 utf-8 format 으로 저장.  [2015.11.11. 추가]  “ansi” or “ANSI” argument 추가.  국가별 Chartset 로 변경  Ex:>  “euc-kr” or “shift-jis” or “utf-8” 등…… |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | 성공 시   * [ Success ] 문자열. //2015.11.24 추가. |

* + **오류 발생 Return Data** //2015.11.24 추가.

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | File open 시 오류 메시지.  [ Failed \_ext\_writeFile file open : + 오류 발생 한 파일 명 ] |

**Method Name : \_ext\_reNameFile**

▶ **Description**

* + 파일이름을 변경합니다.

▶ **Syntax**

var ret;

ret = **this.objExtCommon.\_ext\_reNameFile("C:/TEMP/test/utf-8.txt","C:/TEMP/test/utf-83.txt");**

trace("\_ext\_reNameFile : " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| Oldpath | string | 원본 파일경로. |
| Newpath | string | 새로운 파일명. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| bool | True : 성공.  False : 실패. |

**Method Name : \_ext\_copyFile**

▶ **Description**

* + 파일을 복사 합니다..

▶ **Syntax**

var ret;

ret = **this.objExtCommon**.\_**ext\_copyFile**("C:/TEMP/test/utf-83.txt","C:/TEMP/test/utf-88.txt", true);

trace("\_ext\_copyFile : " + ret);

ret = **this.objExtCommon.\_ext\_copyFile**("C:/TEMP/test/utf-83.txt","C:/TEMP/test/utf-89.txt", false);

trace("\_ext\_copyFile : " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| Oldpath | string | 원본 파일경로. |
| Newpath | string | 새로운 파일 명. |
| flag | bool | True : 같은 이름의 파일이 있으면 복사 취소.  FALSE : 같은 이름의 파일이 있으면, 덮어 쓰기. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| bool | True : 성공.  False : 실패. |

**~~Method Name : \_ext\_excuteAppPostion~~(2018.11.21버전부터 지원안함)**

~~▶~~ **~~Description~~**

* + ~~외부 프로그램을 특정 위치에 실행 합니다.~~

~~▶~~ **~~Syntax~~**

~~var ret;~~

~~ret =~~ **~~this.objExtCommon.\_ext\_excuteAppPostion~~**~~("notepad.exe" ,10,23,800,600);~~

~~trace("\_ext\_excuteAppPostion : " +ret);~~

~~ret =~~ **~~this.objExtCommon.\_ext\_excuteAppPostion~~**~~("notepad.exe c:/TEMP/test.txt" ,800,600,800,600);~~

~~trace("\_ext\_excuteAppPostion : " +ret);~~

~~▶~~ **~~Parameter~~**

|  |  |  |
| --- | --- | --- |
| **~~Arg~~** | **~~Type~~** | **~~Description~~** |
| ~~path~~ | ~~string~~ | ~~실행 하려는 프로그램 경로.~~  ~~참고.~~  ~~System 에 등록 되어 있는 프로그램의 경우 파일명.~~  ~~해당 프로그램에 arg 를 전달 하려면 스트링을 추가.~~  ~~sample~~  **~~"notepad.exe c:/TEMP/test.txt"~~** |
| ~~Int\_x~~ | ~~integer~~ | ~~X 좌표.~~ |
| ~~Int\_y~~ | ~~Integer~~ | ~~Y 좌표.~~ |
| ~~Int\_w~~ | ~~Integer~~ | ~~Weigth~~ |
| ~~Int\_h~~ | ~~integer~~ | ~~Height~~ |

~~▶~~ **~~Return.~~**

|  |  |
| --- | --- |
| **~~Type~~** | **~~Description~~** |
| ~~bool~~ | ~~True : 성공.~~  ~~False : 실패.~~ |

**Method Name : \_ext\_moveFile**

▶ **Description**

* + 파일을 이동합니다.

▶ **Syntax**

var ret;

ret = **this.objExtCommon**.\_ext\_moveFile("C:/TEMP/test/utf-83.txt","C:/TEMP/test/test32/utf-88.txt");

trace("\_ext\_moveFile : " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| Oldpath | string | 원본 파일경로. |
| Newpath | string | 새로운 파일경로. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| bool | True : 성공.  False : 실패. |

**Method Name : \_ext\_deleteFile**

▶ **Description**

* + 파일을 삭제 합니다.

▶ **Syntax**

var ret;

ret = **this.objExtCommon**.\_ext\_deleteFile("C:/TEMP/test/test32/utf-88.txt");

trace("\_ext\_deleteFile : " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| Oldpath | string | 원본 파일경로. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| bool | True : 성공.  False : 실패. |

**~~Method Name : \_ext\_excuteProcess~~(2018.11.21버전부터 지원안함)**

~~▶~~ **~~Description~~**

* + ~~외부 프로그램을 특정 위치에 실행 합니다.~~
  + ~~WorktPath 추가 (2018.10.08)~~

~~▶~~ **~~Syntax~~**

~~var ret;~~

~~ret =~~ **~~this.objExtCommon.\_ext\_excuteProcess~~** ~~("notepad.exe" );~~

~~ret =~~ **~~this.objExtCommon.\_ext\_excuteProcess~~** ~~("notepad.exe”, ”c:/TEMP/test.txt" );~~

~~trace("\_ext\_excuteAppPostion : " +ret);~~

~~ret =~~ **~~this.objExtCommon.\_ext\_excuteProcess~~** ~~("notepad.exe”, ”c:/TEMP/test.txt", “C:\\Windows\\System32\\” );~~

~~▶~~ **~~Parameter~~**

|  |  |  |
| --- | --- | --- |
| **~~Arg~~** | **~~Type~~** | **~~Description~~** |
| ~~path~~ | ~~string~~ | ~~실행 하려는 프로그램 경로.~~  ~~참고.~~  ~~System 에 등록 되어 있는 프로그램의 경우 파일명.~~  ~~해당 프로그램에 arg 를 전달 하려면 스트링을 추가.~~  ~~sample~~  **~~"notepad.exe"~~** |
| ~~Prameter~~ | ~~string~~ | ~~실행 하려는 프로그램의 Parameter~~  ~~“~~**~~c:/TEMP/test.txt~~**~~”~~ |
| ~~WorkPath~~ | ~~string~~ | ~~특정 경로에 있는 프로세스를 실행 할 때 사용.~~  ~~(프로세스의 Path)~~  ~~Default : “” ( empty )~~ |

~~▶~~ **~~Return.~~**

|  |  |
| --- | --- |
| **~~Type~~** | **~~Description~~** |
| ~~bool~~ | ~~True : 성공.~~  ~~False : 실패.~~ |

**Method Name : \_ext\_killProcessAll**

▶ **Description**

* + 프로그램을 강제로 종료 합니다.

▶ **Syntax**

var strFileName = "notepad.exe"

var nCloseOption = 1; // 0 : 첫번째 프로세스 / 1 : 모든 프로세스

**this.objExtCommon.\_ext\_killProcessAll**(strFileName, nCloseOption);

trace("\_ext\_killProcessAll : " );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| strFileName | string | 프로세스 명. |
| nCloseOption | integer | 0 : 검색 되어진 첫 번째 프로세스만 종료  1 : 모든 프로세스 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| bool | 성공 : true / 실패 / false. |

**Method Name : \_ext\_killProcessID**

▶ **Description**

* + 프로세스 아이디를 이용하여 프로그램을 강제로 종료 합니다.

▶ **Syntax**

**this.objExtCommon.\_ext\_killProcessID**(6944);

trace("\_ext\_killProcessID : " );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| INT\_PID | integer | 프로세스 아이디. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
|  | 없음. |

**Method Name : \_ext\_getCurrentProcessID**

▶ **Description**

* + 현재 프로세스의 아이디 리턴 합니다.

▶ **Syntax**

var ret\_ProcessID;

ret\_ProcessID = **this.objExtCommon.\_ext\_getCurrentProcessID**();

trace("\_ext\_getcurrentProcessID : " + ret\_ProcessID );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| integer | 프로세스 아이디. |

**Method Name : \_ext\_getProcessNameHandle**

▶ **Description**

* + 현재 프로세스의 헨들 리턴 합니다.

▶ **Syntax**

var ret\_ProcessHwnd;

ret\_ProcessHwnd = **this.objExtCommon.\_ext\_getProcessNameHandle**("XPlatform.exe");

trace("\_ext\_getcurrentProcessID : " + ret\_ProcessHwnd );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| processname | string | 프로세스 명. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| integer | 프로세스 헨들을 리턴 합니다.. |

**Method Name : \_ext\_getSystemDir**

▶ **Description**

* + 입력된 flag 값을 이용하여 system path 를 리턴 한다..

▶ **Syntax**

var ret\_SysPath;

ret\_SysPath = **this.objExtCommon.\_ext\_getSystemDir**("SYSTEM");

trace("SYSTEM : " + ret\_SysPath );

ret\_SysPath = **this.objExtCommon.\_ext\_getSystemDir**("PROGRAMS");

trace("PROGRAMS : " + ret\_SysPath );

ret\_SysPath = **this.objExtCommon.\_ext\_getSystemDir**("PROGRAM\_COMMON");

trace("PROGRAM\_COMMON : " + ret\_SysPath );

ret\_SysPath = **this.objExtCommon.\_ext\_getSystemDir**("WINDOWS");

trace("WINDOWS : " + ret\_SysPath );

ret\_SysPath = **this.objExtCommon.\_ext\_getSystemDir**("CSIDL\_COMMON\_DOCUMENTS");

trace("CSIDL\_COMMON\_DOCUMENTS : " + ret\_SysPath );

ret\_SysPath = **this.objExtCommon.\_ext\_getSystemDir**("CSIDL\_DESKTOP");

trace("DESKTOP : " + ret\_SysPath );

ret\_SysPath = **this.objExtCommon.\_ext\_getSystemDir**("TEMP");

trace("TEMP : " + ret\_SysPath );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| flag | string | SYSETM: 컴퓨터의 SYSTEM 폴더  PROGRAMS: 컴퓨터의 program files 폴더 (32bit)  PROGRAM\_COMMON: 컴퓨터의 program files 폴더 (32bit)  WINDOWS: 컴퓨터의 window 폴더  CSIDL\_COMMON\_DOCUMENTS : 사용자 문서 폴더 ( 이전에 바탕화면이라고 작성된 내용은 잘못 된 내용 입니다. )  TEMP : Templates 폴더.  2015.5.22 추가된 Parameter  **CSIDL\_ADMINTOOLS** Version 5.0. File system directory that is used to store administrative tools for an individual user. The Microsoft Management Console will save customized consoles to this directory and it will roam with the user.    **CSIDL\_ALTSTARTUP**  File system directory that corresponds to the user's nonlocalized Startup program group.  CSIDL\_APPDATA  File system directory that serves as a common repository for application-specific data. A common path is C:\WINNT\Profiles\username\Application Data.    **CSIDL\_BITBUCKET**  Virtual folder containing the objects in the user's Recycle Bin.  CSIDL\_COMMON\_ADMINTOOLS  Version 5.0. File system directory containing containing administrative tools for all users of the computer.    **CSIDL\_COMMON\_ALTSTARTUP**  File system directory that corresponds to the nonlocalized Startup program group for all users. Valid only for Windows NT? systems.    **CSIDL\_COMMON\_APPDATA**  Version 5.0. Application data for all users. A common path is C:\WINNT\Profiles\All Users\Application Data.    **CSIDL\_COMMON\_DESKTOPDIRECTORY**  File system directory that contains files and folders that appear on the desktop for all users. A common path is C:\WINNT\Profiles\All Users\Desktop. Valid only for Windows NT? systems.    **CSIDL\_COMMON\_DOCUMENTS** File system directory that contains documents that are common to all users. A common path is C:\WINNT\Profiles\All Users\Documents. Valid only for Windows NT? systems.    **CSIDL\_COMMON\_FAVORITES** File system directory that serves as a common repository for all users' favorite items. Valid only for Windows NT? systems.    **CSIDL\_COMMON\_PROGRAMS** File system directory that contains the directories for the common program groups that appear on the Start menu for all users. A common path is c:\WINNT\Profiles\All Users\Start Menu\Programs. Valid only for Windows NT? systems.    **CSIDL\_COMMON\_STARTMENU** File system directory that contains the programs and folders that appear on the Start menu for all users. A common path is C:\WINNT\Profiles\All Users\Start Menu. Valid only for Windows NT? systems.    **CSIDL\_COMMON\_STARTUP** File system directory that contains the programs that appear in the Startup folder for all users. A common path is C:\WINNT\Profiles\All Users\Start Menu\Programs\Startup. Valid only for Windows NT? systems.    **CSIDL\_COMMON\_TEMPLATES** File system directory that contains the templates that are available to all users. A common path is C:\WINNT\Profiles\All Users\Templates. Valid only for Windows NT? systems.    **CSIDL\_CONTROLS** Virtual folder containing icons for the Control Panel applications.    **CSIDL\_COOKIES** File system directory that serves as a common repository for Internet cookies. A common path is C:\WINNT\Profiles\username\Cookies.    **CSIDL\_DESKTOP** Windows Desktop—virtual folder that is the root of the namespace.    **CSIDL\_DESKTOPDIRECTORY** File system directory used to physically store file objects on the desktop (not to be confused with the desktop folder itself). A common path is C:\WINNT\Profiles\username\Desktop    **CSIDL\_DRIVES** My Computer—virtual folder containing everything on the local computer: storage devices, printers, and Control Panel. The folder may also contain mapped network drives.    **CSIDL\_FAVORITES** File system directory that serves as a common repository for the user's favorite items. A common path is C:\WINNT\Profiles\username\Favorites.    **CSIDL\_FONTS** Virtual folder containing fonts. A common path is C:\WINNT\Fonts.    **CSIDL\_HISTORY** File system directory that serves as a common repository for Internet history items.    **CSIDL\_INTERNET** Virtual folder representing the Internet.    **CSIDL\_INTERNET\_CACHE** File system directory that serves as a common repository for temporary Internet files. A common path is C:\WINNT\Profiles\username\Temporary Internet Files.    **CSIDL\_LOCAL\_APPDATA** Version 5.0. File system directory that serves as a data repository for local (non-roaming) applications. A common path is C:\WINNT\Profiles\username\Local Settings\Application Data.    **CSIDL\_MYPICTURES** Version 5.0. My Pictures folder. A common path is C:\WINNT\Profiles\username\My Documents\My Pictures.    **CSIDL\_NETHOOD** File system directory containing objects that appear in the network neighborhood. A common path is  C:\WINNT\Profiles\username\NetHood.    **CSIDL\_NETWORK** Network Neighborhood—virtual folder representing the top level of the network hierarchy.    **CSIDL\_PERSONAL** File system directory that serves as a common repository for documents. A common path is C:\WINNT\Profiles\username\My Documents.    **CSIDL\_PRINTERS** Virtual folder containing installed printers.    **CSIDL\_PRINTHOOD** File system directory that serves as a common repository for printer links. A common path is C:\WINNT\Profiles\username\PrintHood.    **CSIDL\_PROFILE** Version 5.0. User's profile folder.    **CSIDL\_PROGRAM\_FILES**  Version 5.0. Program Files folder. A common path is C:\Program Files.    **CSIDL\_PROGRAM\_FILES\_COMMON** Version 5.0. Program Files folder that is common to all users. A common path is C:\Program Files\Common. Valid only for Windows NT? systems.    **CSIDL\_PROGRAM\_FILES\_COMMONX86** Version 5.0. Program Files folder that is common to all users for x86 applications on RISC systems. A common path is C:\Program Files (x86)\Common.    **CSIDL\_PROGRAM\_FILESX86** Version 5.0. Program Files folder for x86 applications on RISC systems. Corresponds to the %PROGRAMFILES(X86)% environment variable. A common path is C:\Program Files (x86).    **CSIDL\_PROGRAMS** File system directory that contains the user's program groups (which are also file system directories). A common path is C:\WINNT\Profiles\username\Start Menu\Programs.    **CSIDL\_RECENT** File system directory that contains the user's most recently used documents. A common path is C:\WINNT\Profiles\username\Recent. To create a shortcut in this folder, use SHAddToRecentDocs. In addition to creating the shortcut, this function updates the shell's list of recent documents and adds the shortcut to the Documents submenu of the Start menu.    **CSIDL\_SENDTO** File system directory that contains Send To menu items. A common path is c:\WINNT\Profiles\username\SendTo.    **CSIDL\_STARTMENU** File system directory containing Start menu items. A common path is c:\WINNT\Profiles\username\Start Menu.    **CSIDL\_STARTUP** File system directory that corresponds to the user's Startup program group. The system starts these programs whenever any user logs onto Windows NT? or starts Windows? 95. A common path is C:\WINNT\Profiles\username\Start Menu\Programs\Startup.    **CSIDL\_SYSTEM** Version 5.0. System folder. A common path is C:\WINNT\SYSTEM32.    **CSIDL\_SYSTEMX86**  Version 5.0. System folder for x86 applications on RISC systems. A common path is C:\WINNT\SYS32X86.    **CSIDL\_TEMPLATES** File system directory that serves as a common repository for document templates.    **CSIDL\_WINDOWS** Version 5.0. Windows directory or SYSROOT. This corresponds to the %windir% or %SYSTEMROOT% environment variables. A common path is C:\WINNT. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | 디렉터리 경로를 리턴 합니다. |

**Method Name : \_ext\_setTime**

▶ **Description**

* + 입력된 값을 이용하여 system 시간을 설정한다.

▶ **Syntax**

var ret\_SysTime;

ret\_SysTime = **this.objExtCommon.\_ext\_setTime**("20130225000000"); //년,월,일,시간,분,초 : [ len : 14 ]

ret\_SysTime = **this.objExtCommon.\_ext\_setTime**("122432 //시간,분,초 : [ len : 6 ]

if( ret\_SysTime )

alert("OK");

trace( ret\_SysTime );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| strTime | string | Length 가 14 : //년,월,일,시간,분,초 : [ len : 14 ]  Length 가 6 : //시간,분,초 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| bool | True : 성공.  False : 실패. |

**Method Name : \_ext\_getTime**

▶ **Description**

* + 현재 시간을 읽어온다..

▶ **Syntax**

var ret\_SysTime;

ret\_SysTime = **this.objExtCommon.\_ext\_getTime**(); //년,월,일,시간,분,초 : [ len : 14 ]

trace( ret\_SysTime );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | 현재시간. [년,월,일,시간,분,초]  Len : 14 |

**Method Name : \_ext\_getIPAddress**

▶ **Description**

* + 현재 피시의 IP 리스트를 읽어온다..

▶ **Syntax**

//배열로 리턴 합니다.

var ret\_SysipAddress;

ret\_SysipAddress = **this.objExtCommon.\_ext\_getIPAddress**();

trace("IP Address :: " + ret\_SysipAddress )

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| array | String 현재 피시에 등록 되어 있는 아이피 리스트 리턴. |

**Method Name : \_ext\_getMACAddress**

▶ **Description**

* + 현재 피시의 어뎁터의 리스트를 읽어온다.

▶ **Syntax**

//배열로 리턴 합니다.

var ret\_SysMACAddress;

ret\_SysMACAddress = **this.objExtCommon.\_ext\_getMACAddress**(1); //

trace("MAC Address ALL:: " + ret\_SysMACAddress);

ret\_SysMACAddress = **this.objExtCommon.\_ext\_getMACAddress**(); //

trace("MAC Address ONLY:: " + ret\_SysMACAddress);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| flag | integer | 0 : macaddress 만 읽어 옵니다.  1 :adapter 의 name , dec, type, geteway, ip , mask, mac  의 정보를 리턴 합니다. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| array | String 현재 피시에 등록 되어 있는 adapter 의 정보를 리턴 합니다. |

**Method Name : \_ext\_getPrintList**

▶ **Description**

* + 현재 피시의 프린터 리스트를 읽어온다.

▶ **Syntax**

//배열로 리턴 합니다.

var ret\_SysPrintLists;

ret\_SysPrintLists = **this.objExtCommon**.\_ext\_getPrintList(); //

trace("Print List : " + ret\_SysPrintLists);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| array | String 현재 피시에 등록 되어 있는 [프린터 이름]을 string 으로 리턴 합니다. |

**Method Name : \_ext\_setDefaultPrinter**

▶ **Description**

* + 현재 피시의 디폴트 프린터를 설정한다..

▶ **Syntax**

var ret\_SysPrintLists;

ret\_SysPrintLists = **this.objExtCommon**.\_ext\_getPrintList(); //

trace(ret\_SysPrintLists);

var settingRet;

settingRet = **this.objExtCommon.\_ext\_setDefaultPrinter(ret\_SysPrintLists[0]);** //

trace("\_ext\_setDefaultPrinter : " + settingRet);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| printname | string | 설정하려는 프린터의 이름. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| bool | True : 성공.  False : 실패. |

**Method Name : \_ext\_getDefaultPrinter**

▶ **Description**

현재 피시의 디폴트 프린터를 읽어온다.

▶ **Syntax**

var settingRet;

settingRet = **this.objExtCommon.\_ext\_getDefaultPrinte**r(); //

trace("\_ext\_getDefaultPrinter : " + settingRet);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | Default 로 설정 되어 있는 프린터 이름. |

**Method Name : \_ext\_setPageOption**

▶ **Description**

현재 피시의 프린터 페이지 옵션 창을 활성화 합니다.

▶ **Syntax**

var settingRet;

settingRet = **this.objExtCommon**.**\_ext\_setPageOption**()

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
|  | 없음. |

**Method Name : \_ext\_setPrintOption**

▶ **Description**

현재 피시의 프린트 옵션 창을 활성화 합니다

▶ **Syntax**

ar settingRet;

settingRet = **this.objExtCommon.\_ext\_setPrintOption**();

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
|  | 없음. |

**Method Name : .\_ext\_selectFolderPath**

▶ **Description**

입력된 패스를 이용하여 파일 다이얼로그 창을 활성화 하여

선택 완료 후 폴더 명을 리턴 합니다

▶ **Syntax**

var settingRet;

settingRet = **this.objExtCommon**.\_**ext\_selectFolderPath**(); //

trace("\_ext\_selectFolderPath default 설정 시 : " + settingRet);

var setFolder = "D://TEMP";

settingRet = **this.objExtCommon.\_ext\_selectFolderPath**(setFolder); //

trace("\_ext\_selectFolderPath "+ setFolder +" [ 설정 : ] " + settingRet);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| path | sring | Default : null  ( c:/ 를 디폴트 드라이버로 선택 다이얼로그 창을 오픈.)  Path 입력 시 입력된 폴더를 default 폴더로 설정 하여 선택 다이얼로그 창을 오픈 한다. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | 선택 되어진 폴더 경로를 리턴합니다. |

**Method Name : \_ext\_getFileList**

▶ **Description**

입력 된 폴더 경로에서 파일 리스트를 스트링으로 리턴 합니다.

▶ **Syntax**

var settingRet;

var getFilePATH = "C://TEMP";

settingRet = **this.objExtCommon.\_ext\_getFileList**(getFilePATH,"D"); //D

trace("\_ext\_getFileList : " + settingRet );

settingRet = **this.objExtCommon.\_ext\_getFileList**(getFilePATH); //D

trace("\_ext\_getFileList : " + settingRet );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| path | string | 파일경로.. |
| flag | string | Option :  “F”(default) 파일 명을 array로 리턴 한다.  “D”일 때, 디렉터리 명을 리스트를 리턴한다. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| array | 선택된 폴더에서 파일명들을 string array로 리턴 합니다. |

**Method Name : \_ext\_getKeyCheck**

▶ **Description**

현재 피시의 key 활성화 여부를 리턴 합니다.

▶ **Syntax**

var ret = "D://TEMP";

ret = **this.objExtCommon.\_ext\_getKeyCheck**("CAPSLOCK"); //

trace(" CAPSLOCK : " + ret );

ret = **this.objExtCommon.\_ext\_getKeyCheck**("NUMLOCK"); //

trace(" NUMLOCK : " + ret );

ret = **this.objExtCommon.\_ext\_getKeyCheck**("SCROLLOCK"); //

trace(" SCROLLOCK : " + ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| Str KEY | string | CAPSLOCK  NUMLOCK  SCROLLOCK |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| bool | 해당 키의 활성화 여부를 리턴 합니다.. |

**Method Name : \_ext\_getHostName**

▶ **Description**

피시의 호스트명을 리턴합니다..

▶ **Syntax**

var ret = "";

ret = **this.objExtCommon.\_ext\_getHostName**(); //

trace("\_ext\_getHostName : " + ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | 피시의 호스트 명. |

**Method Name : \_ext\_getUserName**

▶ **Description**

로그인 되어 있는 사용자 계정 명.

▶ **Syntax**

var ret = "";

ret = **this.objExtCommon.\_ext\_getUserName** (); //

trace (“\_ext\_getUserName: " + ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | 피시의 로그인 되어 있는 사용자 명. |

**Method Name : \_ext\_getFileDateInfo**

▶ **Description**

해당 파일의 버전 정보를 리턴 합니다.

▶ **Syntax**

var ret;

ret = **this.objExtCommon.\_ext\_getFileDateInfo**("C://TEMP//test.txt"); //

trace("\_ext\_getFileDateInfo: " + ret );

trace("\_ext\_getFileDateInfo 만든날짜[ 0 ]: " + ret[0] );

trace("\_ext\_getFileDateInfo 수정날짜[ 1 ]: " + ret[1]);

trace("\_ext\_getFileDateInfo 저장날자[ 2 ]: " + ret[2]);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| Path | string | 버전정보를 읽으려는 파일 경로. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| array | [0] : 만든 날자 : 년,월,일,시분초  [1] : 수정 날자 : 년,월,일,시분초  [2] : 저장 날자 : 년,월,일,시분초 |

**Method Name : \_ext\_getCheckDir**

▶ **Description**

지정된 경로에 해당 폴더 유무 확인 하여 true, false 리턴.

▶ **Syntax**

var ret;

ret = **this.objExtCommon.\_ext\_getCheckDir**("C://TEMP//test"); //

trace( "\_ext\_getCheckDir : " + ret);

if(ret == false)

{

ret = **this.objExtCommon.\_ext\_getCheckDir**("C://TEMP//test", **1**); //

trace( "\_ext\_getCheckDir : " + ret);

}

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| Path | string | 체크하려는 폴더의 경로. |
| flag | integer | Default :  Null : 체크만 합니다.  1 : 해당 폴더가 없으면, 폴더를 생성합니다. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| bool | true: 있음  false :없음. |

**Method Name : \_ext\_getSystemEnv**

▶ **Description**

system 환경변수에 등록 되어 있는 값을 읽어 옵니다.

▶ **Syntax**

var ret;

ret = **this.objExtCommon.\_ext\_getSystemEnv**("path"); //

trace( " \_ext\_setSystemEnv : " + ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| env | string | 환경변수에 등록 되어진, 환경변수 name. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | 환경변수 명으로 등록 되어진 값들을 string 으로 리턴 합니다. |

**Method Name : \_ext\_getDriveInfo**

▶ **Description**

현재 피시에 등록 되어진 드라이버 정보를 등을 array 로 리턴 합니다..

▶ **Syntax**

var ret;

ret = **this.objExtCommon.\_ext\_getDriveInfo**(); //

trace( " \_ext\_getDriveInfo : " + ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| array | 드라이브 정보를 string array 로 리턴합니다. |

**Method Name : \_ext\_getDriveInfo2**

▶ **Description**

현재 피시에 등록 되어진 드라이버 정보를 등을 array 로 리턴 합니다..

▶ **Syntax**

var ret;

ret = **this.objExtCommon.\_ext\_getDriveInfo2**(); //

trace( " \_ext\_getDriveInfo : " + ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| array | 드라이브 정보를 string array 로 리턴 합니다. |

**참고:**

**\_ext\_getDriveInfo 함수와의 차이점 각각의 drive 의 Type을 알 수 있습니다.**

**Ex:**

C:\ HARD DISK DRIVER,

D:\ HARD DISK DRIVER,

E:\ HARD DISK DRIVER,

F:\ CD-ROM DRIVER,

G:\ HARD DISK DRIVER,

H:\ NETWORK DISK DRIVER

**Method Name : \_ext\_getFontInfo**

▶ **Description**

입력 된 폰트 영역을 이용하여 폰트정보를 array 로 리턴 합니다..

▶ **Syntax**

var ret;

ret = **this.objExtCommon.\_ext\_getFontInfo**("Caption"); //

trace( " \_ext\_getDriveInfo Caption : " + ret );

ret = **this.objExtCommon**.\_ext\_getFontInfo("SmCaption"); //

trace( " \_ext\_getDriveInfo SmCaption : " + ret );

ret = **this.objExtCommon**.\_ext\_getFontInfo("Menu"); //

trace( " \_ext\_getDriveInfo Menu : " + ret );

ret = **this.objExtCommon**.\_ext\_getFontInfo("Status"); //

trace( " \_ext\_getDriveInfo Status : " + ret );

ret = **this.objExtCommon**.\_ext\_getFontInfo("Message"); //

trace( " \_ext\_getDriveInfo Message : " + ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| strFontinfo | string | Caption : 폰트정보.  SmCaption: 폰트정보.  Menu: 폰트정보.  Status: 폰트정보.  Message: 폰트정보. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| array | 드라이브 정보를 string array 로 리턴합니다.  [0] 폰트명.  [1] height.  [2] width.  [3] lfEscapement  [4] lfOrientation.  [5] lfWeight. |

**Method Name : \_ext\_writeUProfile**

▶ **Description**

ini 파일 작성. 성공 유무에 따라. true, false 리턴.

▶ **Syntax**

var groupVal = "English";

var keyVal = "key";

var strVal = "한글";

var ret;;

ret = this.objExtCommon**.\_ext\_writeUProfile**(groupVal,keyVal,strVal,"D:\\TEMP\\iniFileUTF16.ini","utf-16");

trace( " \_ext\_writeUProfile UTF-16 : " + ret );

ret = this.objExtCommon**.\_ext\_writeUProfile**(groupVal,keyVal,strVal,"D:\\TEMP\\iniFile.ini");

trace( " \_ext\_writeUProfile : " + ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| groupVal | string | 그룹명 |
| keyVal | string | 그룹내의 Key명 |
| strVal | string | 지정할 값 |
| strFilename | string | 경로가 포함된 파일명 |
| strUnicode | string | Type : String [ “utf-16” ]  Description : INI File 생성 시 Unicode format 형식으로 파일 생성.  INI file format 은 Unicode format 중 “utf-16 little engdian“ format 형식 사용.  Format 형식을 지정하지 않으면, default “ANSI” format 형식으로 파일 생성 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| bool | True 성공 / False 실패 |

**Method Name : \_ext\_readUProfile**

▶ **Description**

ini 파일 의 키 값을 읽어 들입니다.

▶ **Syntax**

var groupVal = "English"// "한글";

var keyVal = "key";

var strVal = "value";

var ret;

ret = this.objExtCommon**.\_ext\_readUProfile(**groupVal,keyVal,strVal,"D:\\TEMP\\iniFileUTF16.ini");

trace( " \_ext\_readUProfile : " + ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| groupVal | string | 그룹명 |
| keyVal | string | 그룹내의 Key명 |
| strVal | string | 지정된 값을 찾을 수 없을 때 반환되는 디폴트 값 |
| strFilename | string | 경로가 포함된 파일명 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | 해당 위치의 key 값을 읽어 들입니다. |

**Method Name : \_ext\_getAppPath**

▶ **Description**

현재 프로그램의 경로를 출력합니다.

▶ **Syntax**

var ret;

ret = **this.objExtCommon**.\_ext\_getAppPath();

trace("\_ext\_getAppPath : " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  |  |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| string | 현재 프로그램의 폴더 path 를 리턴 합니다. |

**Method Name : \_ext\_getTraySizeInfo**

▶ **Description**

시작 표시줄 및 트레이 영역 위치를 리턴 한다..

▶ **Syntax**

var ret;

ret = **this.objExtCommon.\_ext\_getTraySizeInfo**();

trace("\_ext\_getTraySizeInfo " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| array | rect 좌표.  [ 0 ] 상태표시줄 rect.left  [ 1 ] 상태표시줄 rect.top  [ 2 ] 상태표시줄 rect.right  [ 3 ] 상태표시줄 rect.bottom  [ 4 ] 트레이영역 rect.left  [ 5 ] 트레이영역 rect.top  [ 6 ] 트레이영역 rect.right  [ 7 ] 트레이영역 rect.bottom |

**Method Name : \_ext\_getDesktopWorkArea**

▶ **Description**

메인 화면의 작업 영역을 리턴 합니다.

▶ **Syntax**

var ret;

ret = **this.objExtCommon.\_ext\_getDesktopWorkArea**();

trace("\_ext\_getDesktopWorkArea : "+ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| array | rect 좌표.  [ 0 ] 작업 영역 rect.left  [ 1 ] 작업 영역 rect.top  [ 2 ] 작업 영역 rect.right  [ 3 ] 작업 영역 rect.bottom |

**Method Name : \_ext\_setSecuritySite(strStie)**

▶ **Description**

신뢰할 수 있는 사이트 등록

▶ **Syntax**

var site = "www.miplalform.com";

var ret;

ret = **this.objExtCommon**.\_ext\_setSecuritySite(site);

trace("\_ext\_setSecuritySite : " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| stie | string | 등록 하려는 site string 데이터. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| integer | 1 성공  0 실패  2 등록하려는 영역이 서버 확인이 필요.  3 등록하려는 영역이 이미 설정되어 있음. |

**Method Name : \_ext\_EmptyWorkingSet**

▶ **Description**

Nexacro 프로세스의 작업 집합에서 가능한 한 많은 페이지(workingset memory)를 제거합니다.

PSAPI 기능 중 [ BOOL WINAPI EmptyWorkingSet ] 함수 추가.

▶ **Syntax**

var ret;

ret = **this.objExtCommon.\_ext\_EmptyWorkingSet**();

trace("\_ext\_EmptyWorkingSet: " + ret);

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| bool | true 성공  false 실패 |

**Method Name : \_ext\_setBeep**

▶ **Description**

DESC \_ext\_setBeep(frequency, length)

INPUT Beep 음 발생.

frequency 는 주파수 (높을수록 높은음) <vc : Beep() 함수 참고.>

length 는 1/1000초 단위.

frequency

경고음의 주파수이며 해당 범위는 37 - 32767헤르츠입니다.

length

경고음의 지속 시간(밀리초)입니다.

OUPUT 없음.

▶ **Syntax**

**this.objExtCommon.\_ext\_setBeep**(1000,500);

▶ **Return.**

없음.

**Method Name : .\_ext\_isWow64**

▶ **Description**

Windows OS BIT 수 출력.

▶ **Syntax**

var ret = **this.objExtCommon.\_ext\_isWow64()**;

if( ret === true)

trace("Windows OS 64BIT ");

else

trace("Windows OS 32BIT ");

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | TRUE : 64BIT OS  FALSE : 32BIT OS |

**Method Name : \_ext\_CreateDirectory**

▶ **Description**

특정 경로에 폴더를 생성 합니다.

예외:>

생성 DRM 보안 및 Windows 보안 설정의 영향으로 폴더 생성이 안될 수도 있습니다.

▶ **Syntax**

var strCreatePath = "D:/TEMP/\_\_Encoding/日本語Test/日本語Test2**/"**;

// OR

strCreatePath = "D:\\TEMP\\\_\_Encoding\\日本語Test\\日本語Test2**\\**";

ret = this.objExtCommon.\_ext\_CreateDirectory(strCreatePath);

trace ( "return : " +ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
| strCreatePath | STRING | 생성하려는 폴더 경로.  하위 폴더 경로 까지 모두 생성이 됨.  중요:  경로 마지막 폴더 인식 데이터를 입력 하여야 함. **[ / or \\ ]** |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | TRUE : 성공.  FALSE : 실패. |

**Method Name : \_ext\_setInternetSetCookie**

▶ **Description**

URL 에 해당 하는 쿠키 정보를 넣어주는 기능.

참고:MSDN : [ InternetSetCookie 함수 호출. ]

https://msdn.microsoft.com/ko-kr/library/windows/desktop/aa385107(v=vs.85).aspx

▶ **Syntax**

var url = "http://www.전송하려는 URL.COM";

var name = "cookie";

var value = "123";

var ret = this.objExtCommon.\_ext\_setInternetSetCookie(url,name,value );

trace ( "return : " +ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| strURL | **String** | 쿠키 설정하려는 해당 URL 입력. |
| strCookieName | **String** | 쿠키 이름 |
| strCookieValue | **String** | 쿠키 값. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | TRUE : 성공.  FALSE : 실패. |

**Method Name : \_ext\_getBitCalc**

▶ **Description**

Bit연산

▶ **Syntax**

var strTag = ">>"; //&,|,>>,<<,~ 중 하나

var value1 = 40;

var value2 = 1;

var ret = this.objExtCommon.\_ext\_getBitCalc(strTag,value1,value2 );

trace ( "return : " +ret );

ret = this.objExtCommon.\_ext\_getBitCalc(">>",-65,1 );

trace ( "return : " +ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| strTag | **String** | 문자열로 &,|,>>,<<,~ 중 하나 |
| value1 | **integer** | bit연산할 첫번째값 |
| value2 | **integer** | bit연산할 2번째 값(shift연산은 shift수) |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| INTEGER | BIT 연산한 결과 값 |

**Method Name : MMFMaping.\_ext\_MmfFileWrite**

▶ **Description**

입력 된 경로에 파일을 생성

(파일 이 없으면 : 파일을 생성. / 있으면 : 해당 파일에 데이터 추가)

▶ **Syntax**

var ret = this.objExtCommon.**MMFMaping.\_ext\_MmfFileWrite**("D:\\Temp\\\_MMFileWrite.txt",sData)

if(ret == false) //실패.

{

trace(“실패”);

}else {

trace(“성공”);

}

this.objExtCommon.**MMFMaping.\_ext\_MmfFileClose**();

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| strPath | **String** | 저장하려는 파일의 경로 |
| strData | **String** | 파일에 저장할 데이터 |
| strFormat | **integer** | 저장하려는 데이터의 Foramt  Default : BOM 없는 utf-8 format 으로 저장.  국가별 Chartset 로 변경  Ex:>  “euc-kr” or “shift-jis” or “utf-8” 등…… |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | TRUE : 성공.  FALSE : 실패. |

**Method Name : MMFMaping.\_ext\_MmfFileClose()**

▶ **Description**

MMFMaping.\_ext\_MmfFileWrite 함수를 이용하여 File Write 후 Close 호출.

▶ **Syntax**

var ret = this.objExtCommon.MMFMaping.\_ext\_MmfFileWrite("D:\\Temp\\\_MMFileWrite.txt",sData)

if(ret == false) //실패.

{

trace(“실패”);

}else {

trace(“성공”);

}

this.objExtCommon.**MMFMaping.\_ext\_MmfFileClose**();

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
|  |  | 없음. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | TRUE : 성공.  FALSE : 실패. |

**Method Name : \_ext\_getFileInfo (strPath)**

▶ **Description**

**경로에 위치해 있는 파일의 정보를 취득 합니다.**

▶ **Syntax**

var tmpPath = "D:/\_WORK\_Project\_V8/\_ExtCommonApi\_V13/\_src/ExtCommonV13/release/ExtCommonV13.dll";

var ret;

ret = this.objExtCommon.\_ext\_getFileInfo(tmpPath);

var len = ret.length

trace("File 설명 :: " + ret[0] );

trace("File 버전 :: " + ret[1] );

trace("File interalname :: " + ret[2] );

trace("file 회사 명 :: " + ret[3] );

trace("file Copyright :: " + ret[4] );

trace("file orignalfilename :: " + ret[5] );

trace("file productname :: " + ret[6] );

trace("file productversion :: " + ret[7] );

trace("file Size :: " + ret[8] ); // **[ Byte ]**

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **strPath** | **String** | 파일의 경로. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| ARRAY | trace("File 설명 :: " + ret[0] );  trace("File 버전 :: " + ret[1] );  trace("File interalname :: " + ret[2] );  trace("file 회사 명 :: " + ret[3] );  trace("file Copyright :: " + ret[4] );  trace("file orignalfilename :: " + ret[5] );  trace("file productname :: " + ret[6] );  trace("file productversion :: " + ret[7] );  trace("file Size :: " + ret[8] ); **//2015.11.16.1 추가 [ Byte ]** |

**Method Name : \_ext\_getIMEmode()**

▶ **Description**

**현재 활성화 된 최상위 윈도우의 IME Mode 를 읽어 옵니다.**

▶ **Syntax**

var ret;

ret = this.objExtCommon.\_ext\_getIMEmode();

trace("IME : " + ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
|  |  | 없음. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Integer | 0 : English  1 : Native  3 : Language |

**Method Name : \_ext\_setIMEmode(bImeFlag)**

▶ **Description**

**현재 활성화 된 최상위 윈도우의 IME Mode 를 설정 합니다.**

▶ **Syntax**

var ret;

ret = this.objExtCommon.\_ext\_getIMEmode();

if( ret == 0 ) //English

{

Extension.\_ext\_setIMEmode(true); //native

}

else if( ret == 1 ) //native

{

Extension.\_ext\_setIMEmode(false); // English

}

trace("IME : " + ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **bImeFlag** | **BOOL** | True : Native  False : .English |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
|  | 없음 |

**Method Name : \_ext\_msgBox (strTitle, strMessage, strStyle)**

▶ **Description**

**Windows Api 함수 인 MessageBox 함수를 호출 하여 사용.**

▶ **Syntax**

var ret;

ret = this.objExtCommon.\_ext\_msgBox("MSG BOX", "MSG HOHOHO" , "MB\_YESNO |MB\_ICONSTOP");

trace( "\_ext\_msgBox :: " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **strTitle** | **String** | MessageBox 의 Title 명 |
| **strMessage** | **String** | MessageBox 의 내용 |
| **strStyle** | **String** | MessageBox Button Style 설정.  Type   |  | | --- | | **MB\_ABORTRETRYIGNORE** | | **MB\_OK** | | **MB\_OKCANCEL** | | **MB\_RETRYCANCEL** | | **MB\_YESNO** | | **MB\_YESNOCANCEL** | | **IDCANCEL** |   Modality   |  | | --- | | **MB\_APPLMODAL** | | **MB\_SYSTEMMODAL** | | **MB\_TASKMODAL** |   Icon   |  | | --- | | **MB\_ICONEXCLAMATION** | | **MB\_ICONINFORMATION** | | **MB\_ICONQUESTION** | | **MB\_ICONSTOP** | | **MB\_ICONHAND** |   Default button   |  | | --- | | **MB\_DEFBUTTON1** | | **MB\_DEFBUTTON2** | | **MB\_DEFBUTTON3** | |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| integer | |  |  | | --- | --- | | **Return code/value** | **Description** | | **IDABORT**  3 | The **Abort** button was selected. | | **IDCANCEL**  2 | The **Cancel** button was selected. | | **IDCONTINUE**  11 | The **Continue** button was selected. | | **IDIGNORE**  5 | The **Ignore** button was selected. | | **IDNO**  7 | The **No** button was selected. | | **IDOK**  1 | The **OK** button was selected. | | **IDRETRY**  4 | The **Retry** button was selected. | | **IDTRYAGAIN**  10 | The **Try Again** button was selected. | | **IDYES**  6 | The **Yes** button was selected. | |

**Remark**

|  |  |
| --- | --- |
| **Icon** | **Flag values** |
| Icon for MB_ICONHAND, MB_ICONSTOP, and MB_ICONERROR | **MB\_ICONHAND**, **MB\_ICONSTOP**, or **MB\_ICONERROR** |
| Icon for MB_ICONQUESTION | **MB\_ICONQUESTION** |
| Icon for MB_ICONEXCLAMATION and MB_ICONWARNING | **MB\_ICONEXCLAMATION** or **MB\_ICONWARNING** |
| Icon for MB_ICONASTERISK and MB_ICONINFORMATION | **MB\_ICONASTERISK** or **MB\_ICONINFORMATION** |

**Method Name : \_ext\_getInternetConnectState()**

▶ **Description**

**인터넷 연결 상태를 확인 합니다.**

▶ **Syntax**

var ret;

ret = this.objExtCommon.\_ext\_getInternetConnectState();

trace( "\_ext\_msgBox :: " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
|  |  | 없음. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | TRUE : 연결 됨.  FALSE : 연결 되어 있지 않음. |

**Method Name : \_ext\_setOleServerBusyTimeOut(strReg, iTime, bEnable, bBusyDialog )**

▶ **Description**

사용하고자 하는 ActiveX 컨트롤 또는 DCOM 클라이언트가 어떤 이유로 인해 응답하지 못하고 있는 상태일 때 사용.(다음과 같은 이유에서 발생할 수 있습니다.)

1. ActiveX 컨트롤이 초기화되기 전에 클라이언트 애플리케이션이 이를 사용하려고 시도할 때

2. 시스템이 느리거나 동시에 많은 애플리케이션이 동작 중이어서 ActiveX 컨트롤이 응답하지 못하는 경우

3. ActiveX 컨트롤의 설계에 문제가 있어서 응답이 늦도록 제작되어 있을 때

▶ **Syntax**

var ret;

var strReg = true

var iTime = 5000;

var bEnable = false;

var bBusyDialog = false;

ret = this.objExtCommon.\_ext\_setOleServerBusyTimeOut(strReg, iTime, bEnable, bBusyDialog);

//------------------------------//

// OLE ActiveX 작업..

ret = this.objExtCommon.\_ext\_setOleServerBusyTimeOut(false);

//------------------------------//

trace( "ext\_setOleServerBusyTimeOut :: " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **strReg** | **String** | TRUE : 설정.  FALSE : 해제: (default) |
| **iTime** | **Integer** | 딜레이 시간. (밀리세컨드)  Default 10000 |
| **bEnable** | **String** | (키보드 또는 마우스 메시지는 보류 중인 호출 및 호출 하는 동안 초과 되었습니다 (초과 메시지 보류 중인 지연), "응답 없음" 대화 상자가 표시 됩니다.) 메시지 박스 활성화 유무  TRUE : 활성화  FALSE : 비활성화 ( Default ) |
| **bBusyDialog** | **String** | 메시지 박스 활성화 유무  TRUE : 활성화  FALSE : 비활성화 ( Default ) |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
|  | 없음 |

**Method Name : \_ext\_getInstallProg(strProgidORclsid))**

▶ **Description**

**PC 에 입력 한 ProgID 또는 CLSID 가 등록 되어 있는지 유무..**

▶ **Syntax**

var bInstall = this.objExtCommon.\_ext\_getInstallProg("Excel.Application");

if( bInstall )

{

trace("Prog Install.");

}

else

{

trace("Not Prog Install");

}

bInstall = this.objExtCommon.\_ext\_getInstallProg("{00024500-0000-0000-C000-000000000046}");

if( bInstall )

{

trace("CLSID Install");

}

else

{

trace("Not CLSID Install");

}

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **strProgidORclsid** | **STRING** | Progid 또는 Clsid 입력. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | TRUE : 설치됨.  FALSE : 설치되어 있지 않음. |

**Method Name : \_ext\_setShowCursor(bCursor)**

▶ **Description**

마우스 커서 보이기 / 숨기기 기능.

▶ **Syntax**

var ret = this.objExtCommon.\_ext\_setShowCursor(false);

alert("\_ext\_setShowCursor ");

ret = this.objExtCommon.\_ext\_setShowCursor(true);

trace(" showcursor "+ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| bCursort | **BOOL** | TRUE : 보이기 [커서의 디스플레이 카운터를 하나 증가시키고 ]  FALSE: : 숨기기 [ 디스플레이 카운터를 하나 감소 ] |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| integer | 커서의 디스플레이 카운트 |

**Method Name :** **\_ext\_getNumFormatLoc(strNumberData, intLCID/LCID, intFmt);**

▶ **Description**

국가별 숫자 표현 변환 함수.

▶ **Syntax**

var str = this.objExtCommon.\_ext\_getNumFormatLoc("123456789","ko",4);

trace(str);

str = this.objExtCommon.\_ext\_getNumFormatLoc("123456789",0x0412,4);

trace(str);

str = this.objExtCommon.\_ext\_getNumFormatLoc("123456789", 1062 ,4);

trace(str);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **strNumberData** | **String** | 출력하려는 숫자 |
| **intLCID** | **Integer/String** | 국가 코드 LCID  **StrLCID** : 다국어 언어 코드 사용.(ISO Language Code Table)  참고:  <https://msdn.microsoft.com/ko-kr/library/windows/apps/jj657969.aspx>  **IntLCID** 의 경우 HEX 코드 및 DEC 사용가능  참고  <https://msdn.microsoft.com/en-us/goglobal/bb964664.aspx> |
| **IntFmt** | **integer** | 소수점 자릿수 표현. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| **BOOL** | **실패 시 : False 리턴.** |
| **String** | 성공 시 변환된 숫자 데이터 출력. |

참고:

* + Nexacro Studio 에서 Output 로그에 국가별 숫자 출력 시 [12?345?678?009,0000 ]과 같이 ? 로 표현 되는 것은 Output 로그 출력 창에서 공백 표현을 하지 못하여 발생 한 것일 뿐 text 컴포넌트에 출력하여 보면 정상 출력 되는 것을 확인할 수 있다..

**Method Name : \_ext\_getDateUTC (intYear, intMon, intDay, intHour, intMin, intSec)**

▶ **Description**

* + (년,월,일,시,분,초) 를 입력하게 되면 UTC 시간을 1970년도 부터 초단위로 계산된 타임스탬프 형식으로 값을 반환

▶ **Syntax**

var gap1= this.objExtCommon.\_ext\_getDateUTC(2005,4,12,0,0,0);

trace(gap1);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| intYear | **Integer** | 숫자로 된 년 |
| intMon | **Integer** | 숫자로 된 월 |
| intDay | **Integer** | 숫자로 된 일 |
| intHour | **Integer** | 숫자로 된 시간 |
| intMin | **Integer** | 숫자로 된 분 |
| intSec | **Integer** | 숫자로 된 초 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| integer | 초 단위의 숫자. |

**Method Name : \_ext\_setOutputdebugstring (strMsg)**

▶ **Description**

* + **Windows Api 함수 인 “OutputDebugString” 을 사용하는 함수.**
  + **debugView 등으로 해당 메시지를 확인할 수 있습니다.**

▶ **Syntax**

this.objExtCommon.\_ext\_setOutputdebugstring("OutPutDebugString.한글");

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| strMsg | **String** | 출력하려는 내용. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
|  | 없음. |

**Method Name : \_ext\_getMaxMonitorRect ()**

▶ **Description**

**활성화 된 Window 가 위치해 있는 모니터의 최대 해상도.**

▶ **Syntax**

var rect = this.objExtCommon.\_ext\_getMaxMonitorRect();

trace(rect);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| strMsg | **String** | 출력하려는 내용. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Array | 해상도 구하기.  ect 좌표.  [ 0 ] rect.left  [ 1 ] rect.top  [ 2 ] rect.right  [ 3 ] rect.bottom |

**Method Name : \_ext\_getMonitorInfo()**

▶ **Description**

**PC 의 모니터 정보를 취득 합니다.**

**모니터 개수, 각각의 해상도, 주 모니터 여부 등...**

▶ **Syntax**

var ret = this.objExtCommon.\_ext\_getMonitorInfo();

trace("리턴 데이터 " + ret);

//모니터 갯수 [ 배열로 리턴. integer ]

var count = ret[0];

trace("모니터의 갯수 : " + count);

//가상 모니터 [ integer ,영역 2차원 배열로 리턴. ]

trace("가상 모니터 영역 : ");

trace("left : " + ret[1][0]); //left

trace("top : " + ret[1][1]); //top

trace("rigth : " + ret[1][2]); //right

trace("bottom : " + ret[1][3]); //bottom

for(var i = 2; i<count+2;i++)

{

// 모니터의 좌표,

// 마지막은 데이터는 주 모니터 여부(1이면주모니터,0이면서브 모니터)

// integer , 영역 2차원 배열로 리턴.

trace("각 모니터 좌표" + ret[i] );

trace("left : " + ret[i][0]); //left

trace("top : " + ret[i][1]); //top

trace("right : " + ret[i][2]); //right

trace("bottom : " + ret[i][3]); //bottom

trace("주 모니터 여부 1이면 주모니터 : " + ret[i][4]); //주 모니터 여부

}

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
|  |  | 없음 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Array | 샘플 Script 참고.  각각의 정보 들이 설명 되어 있습니다. |

**Method Name : \_ext\_getGetKeyState (strKey(intKey))**

▶ **Description**

**키보드의 눌림 상태를 검사**

▶ **Syntax**

var keyCode = this. objExtCommon.\_ext\_getGetKeyState("CTRL");

trace("string :: " + keyCode);

//or

keyCode = this.Ext objExtCommon.\_ext\_getGetKeyState(17);

trace("int :: " + keyCode);

;▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| strMsg | **String**  **또는**  **Integer** | **STRING** :   |  |  | | --- | --- | | **입력값** | **의미** | | Control | Control Key | | Ctrl | Control Key | | Alt | Alt Key | | Shift | Shift Key | | MouseLeft | Mouse의 Left Button | | MouseMiddle | Mouse의 Middle Button | | MouseRight | Mouse의 Right Button | | 0~9 | 0~9번 숫자키 | | a-z | a-z키. | |  |  | | BackSpace | Back Space 키 | | Delete | Delete키 | | F1~F12 | F1~F12키 | | NUMPAD0~NUMPAD9 | 키보드 오른쪽의 숫자패드의 0번에서 0번사이의 값. | | Home | Home키 | | End | END키 | | PageUp | PageUp키 | | PageDown | PageDown키 |   **INTEGER** 키보드 키 값   |  |  |  | | --- | --- | --- | | **Virtual Key - Codes** | | **Description** | | **Dec** | **Hex** | | 1 | 1 | Left Mouse Button | | 2 | 2 | Right Mouse Button | | 3 | 3 | Control-Brake Processing | | 4 | 4 | Middle Mouse Button | | 8 | 8 | BackSpace Key | | 9 | 9 | Tab Key | | 12 | C | Clear Key | | 13 | D | Enter Key | | 16 | 10 | Shift Key | | 17 | 11 | Ctrl Key | | 18 | 12 | Alt Key | | 19 | 13 | Pause Key | | 20 | 14 | Caps Lock Key | | 27 | 1B | ESC Key | | 32 | 20 | SpaceBar | | 33 | 21 | Page Up Key | | 34 | 22 | Page Down Key | | 35 | 23 | End Key | | 36 | 24 | Home Key | | 37 | 25 | Left Arrow Key | | 38 | 26 | Up Arrow Key | | 39 | 27 | Right Arrow Key | | 40 | 28 | Down Arrow Key | | 41 | 29 | Select Key | | 42 | 2A | Print Key | | 43 | 2B | Execute Key | | 44 | 2C | Print Screen Key | | 45 | 2D | Ins Key | | 46 | 2E | Del Key | | 47 | 2F | Help Key | | 48 | 30 | 0 Key | | 49 | 31 | 1 Key | | 50 | 32 | 2 Key | | 51 | 33 | 3 Key | | 52 | 34 | 4 Key | | 53 | 35 | 5 Key | | 54 | 36 | 6 Key | | 55 | 37 | 7 Key | | 56 | 38 | 8 Key | | 57 | 39 | 9 Key | | 65 | 41 | A Key | | 66 | 42 | B Key | | 67 | 43 | C Key | | 68 | 44 | D Key | | 69 | 45 | E Key | | 70 | 46 | F Key | | 71 | 47 | G Key | | 72 | 48 | H Key | | 73 | 49 | I Key | | 74 | 4A | J Key | | 75 | 4B | K Key | | 76 | 4C | L Key | | 77 | 4D | M Key | | 78 | 4E | N Key | | 79 | 4F | O Key | | 80 | 50 | P Key | | 81 | 51 | Q Key | | 82 | 52 | R Key | | 83 | 53 | S Key | | 84 | 54 | T Key | | 85 | 55 | U Key | | 86 | 56 | V Key | | 87 | 57 | W Key | | 88 | 58 | X Key | | 89 | 59 | Y Key | | 90 | 5A | Z Key | | 112 | 70 | F1 Key | | 113 | 71 | F2 Key | | 114 | 72 | F3 Key | | 115 | 73 | F4 Key | | 116 | 74 | F5 Key | | 117 | 75 | F6 Key | | 118 | 76 | F7 Key | | 119 | 77 | F8 Key | | 120 | 78 | F9 Key | | 121 | 79 | F10 Key | | 122 | 7A | F11 Key | | 123 | 7B | F12 Key | | 144 | 90 | Num Lock Key | | 145 | 91 | Scroll Lock Key | |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | TRUE : 키 입력.  FALSE : 키 미입력 |

**Method Name : \_ext\_getFindProcessInfo(strProcessName)**

▶ **Description**

**Process 이름으로 해당 프로세스 파일의 정보를 리턴 받는다.**

**프로세스 명이 같은 프로세스를 모두 검색하여 Array로 리턴 합니다.**

**64BIT 경로의 프로그램의 경우 접근이 되지 않습니다.**

▶ **Syntax**

var ret = this. objExtCommon.\_ext\_getFindProcessInfo("chrome.exe");

trace(ret);

var len = ret.length;

trace( len );

for( var Rowindex=0; Rowindex<len; Rowindex++)

{

trace( "Rowindex: " +Rowindex+ " info " + ret[Rowindex]);

var RetLen = ret[Rowindex].length;

for( var Columnindex=0; Columnindex < RetLen; Columnindex++)

{

if( Columnindex == 4)

{

trace("------------Process File Info----------------");

}

trace("Col " + Columnindex + " : " + ret[Rowindex][Columnindex]);

}

trace( "\r\n");

}

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| strProcessName | **String** | 검색하고자 하는 프로세스 명을 String 으로 입력합니다. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Array | [0][0] : 프로세스 아이디.  [0][1] : 프로세스의 핸들 (윈도우가 있는 것만 검색 됨.)  [0][2] : 프로세스가 위치해 있는 경로.  [0][3] : 프로세스의 Title. (윈도우가 있는 것만 검색 됨.)  //-------------------------------------//  [0][4] : 파일 속성 중 “설명”에 해당하는 정보.  [0][5] : 파일 속성 중 “파일버전”에 해당하는 정보  [0][6] : 파일 속성 중 “InternalName”에 해당하는 정보.  [0][7] : 파일 속성 중 “CompanyName”에 해당하는 정보.  [0][8] : 파일 속성 중 “Copyright”에 해당하는 정보.  [0][9] : 파일 속성 중 “OrignalFileName”에 해당하는 정보.  [0][10]: 파일 속성 중 “ProductName”에 해당하는 정보.  [0][11]: 파일 속성 중 “ProductVersion”에 해당하는 정보.  ◈찾지 못하였을 경우 이거나 데이터가 없을 경우 “” 비어있는 데이터로 리턴. |

**Method Name : \_ext\_getFindAllWinHwnd(strTitle);**

▶ **Description**

**입력 하는 윈도우 타이틀 명을 이용하여 같은 이름으로 된 윈도우 핸들을 모두 취득..**

▶ **Syntax**

var ret = this. objExtCommon.\_ext\_getFindAllWinHwnd("새 탭 - Chrome");

trace("array len : " + ret.length +" :: " + ret);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| strTitle | **String** | 찾고자 하는 윈도우의 title |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Array | Array 로 리턴 되며, 데이터 타입은 strTitle에 해당하는 문자열이 있는 경우 “handle:title”,”handle:title”규칙으로 array return 되요. |

**Method Name : \_ext\_getNumLocaleInfo(strLCID/intLCID,intFmt)**

▶ **Description**

**입력 하는 윈도우 타이틀 명을 이용하여 같은 이름으로 된 윈도우 핸들을 모두 취득..**

▶ **Syntax**

var str = this. objExtCommon.\_ext\_getNumLocaleInfo("1041",4);

trace(str);

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| StrLCID  or  intLCID | **Integer/String** | 국가 코드 LCID  StrLCID : 다국어 언어 코드 사용.( ISO Language Code Table)  참고:  <https://msdn.microsoft.com/ko-kr/library/windows/apps/jj657969.aspx>  IntLCID 의 경우 HEX 코드 및 DEC 사용가능  참고  <https://msdn.microsoft.com/en-us/goglobal/bb964664.aspx> |
| IntFmt | **integer** | 소수점 자릿수 표현. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Array | **[0]** : NumDigits   : 소숫점이하를 표시할 자릿수  **[1]** : LeadingZero   : 만약에 0라면 NumDigits 에서 선언한 자리수에 0 을 채워넣지않고 만일 0 이외의 값이라면     0을 채워넣는다. (예를 들어 NumDigits 가 3이고 숫자가 1.23 이라면 그리고 LeadingZero 가     0이라면 1.23 이라고 변환 될것이고 만일 LeadingZero 가 0 이외의 값이라면 1.230 이라고     변환 될것임) **[2]** : Grouping    : 구분할 단위값을 정의합니다. 일반적으로 3을 사용하고 1,234,567 라고 변환된다. 0~9 까지 가능하다. **[3]** : lpDecimalSep   : 소숫점을 표현할 character 를 정의한다. **[4]** : lpThousandSep   : 단위 구분자를 표현할 character 를 정의한다. **[5]** : NegativeOrder   : 음수값을 변환하는 방법을 정의한다.   예) -11.1    0 일 경우 11.1    1 일 경우 -11.1    2 일 경우 - 11.1 ( - 다음에 space 를 삽입한다.)    3 일 경우 11.1-    4 일 경우 11.1 - ( - 전에 space 를 삽입한다.) |

**~~Method Name : \_ext\_winExec(strCommand, intStateFlg)~~ (2018.11.21버전부터 지원안함)**

~~▶~~ **~~Description~~**

**~~외부 프로그램을 실행 한다.~~**

**~~중요 : 호출한 프로그램에 대한 리턴 값이 없음~~**

**~~(프로그램이 실행하고 나면 이전의 프로세스가 새로이 생성된 프로세스에 접근할 수 없다. [정상구동이 되었는지, 종료 시 리턴 값 등] )~~**

~~▶~~ **~~Syntax~~**

~~var ret = this. objExtCommon.\_ext\_getFindAllWinHwnd("새 탭 - Chrome");~~

~~trace("array len : " + ret.length +" :: " + ret);~~

~~▶~~ **~~Parameter~~**

|  |  |  |
| --- | --- | --- |
| **~~Parameters~~** | **~~Type~~** | **~~Description~~** |
| **~~strCommand~~** | **~~String~~** | ~~절대 경로를 입력시에~~**~~절대경로~~**~~에 맞춰서 실행됩니다. 그 외에~~**~~상대경로~~**~~를 입력시 1. 함수를 실행하는 프로그램이 존재하는 디렉토리 2. Windows System 디렉토리 3. Windows 디렉토리 4. :: PATH :: 환경변수에 등록된 디렉토리 를 순서로 검색하여 실행합니다.  \* lpCmdLine 입력시 보안상 문제가 발생한다. WinExec( "C:\\Program Files\\Games.... " , ...); 이렇게 입력했다고 가정했을때, lpCmdLine으로 넘어가는 인자의 값은 c: -> Program Files -> Games -> 로 검색해서 파일을 찾는다. 하지만 중간에 누군가 악의적으로 "Program.exe"라는 파일을 만들어 놓는다면 위처럼 Program Files를 호출하는 모든 프로세스가 Program.exe 파일을 실행하게 된다. 따라서 인자를 넘길때 " (\" ) 따옴표로 묶는걸 권장한다.~~ |
| **intStateFlg** | **Integer** | **Default** : 0 입니다.( 화면에 보이지 않습니다);  #define SW\_HIDE 0  #define SW\_SHOWNORMAL 1  #define SW\_NORMAL 1  #define SW\_SHOWMINIMIZED 2  #define SW\_SHOWMAXIMIZED 3  #define SW\_MAXIMIZE 3  #define SW\_SHOWNOACTIVATE 4  #define SW\_SHOW 5  #define SW\_MINIMIZE 6  #define SW\_SHOWMINNOACTIVE 7  #define SW\_SHOWNA 8  #define SW\_RESTORE 9  #define SW\_SHOWDEFAULT 10  #define SW\_FORCEMINIMIZE 11  #define SW\_MAX 11 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Integer | **성공**   * **31 보다 높은 값.**   **실패**   * **0** : out of memory, out of resource * **11** : 실행 할 수 없는 파일. * **2** : 정해진 경로에 파일이 없다. * **3** : 정해진 경로가 존재 하지 않는다. |

**Method Name : \_ext\_getLocaleFormat(strTitle);**

▶ **Description**

* 국가별 언어설정 및 표현 형식 정보 취득.
* WindowsApi 함수인 [ GetLocaleInfo ] 를 이용.
  + 상세한 정보는 Windows API 함수 인

int **GetLocaleInfo**( LCID Locale, LCTYPE LCType, LPTSTR lpLCData, int cchData );

에서 [ **LCTYPE** LCType ] 에 관한 정보를 검색하여 주세요.

▶ **Syntax**

trace( "LANG : " + this. objExtCommon.\_ext\_getLocaleFormat("Korean","LOCALE\_SLANGUAGE") + " : " );

trace( "LANG : " + this. objExtCommon.\_ext\_getLocaleFormat("ko","LOCALE\_SLANGUAGE") + " : ");

trace( "LANG : " + this. objExtCommon.\_ext\_getLocaleFormat(1042,"LOCALE\_SLANGUAGE") + " : " );

trace( "Short Date : " + this. objExtCommon.\_ext\_getLocaleFormat("Korean","LOCALE\_SSHORTDATE") + " : " );

trace( "Short Date : " + this. objExtCommon.\_ext\_getLocaleFormat("ko","LOCALE\_SSHORTDATE") + " : " );

trace( "Short Date : " + this. objExtCommon.\_ext\_getLocaleFormat(1042,"LOCALE\_SSHORTDATE") + " : " );

trace( "Long Date :" + this. objExtCommon.\_ext\_getLocaleFormat("Korean","LOCALE\_SLONGDATE") + " : " );

trace( "Long Date :" + this. objExtCommon.\_ext\_getLocaleFormat("ko","LOCALE\_SLONGDATE") + " : " );

trace( "Long Date :" + this. objExtCommon.\_ext\_getLocaleFormat(1042,"LOCALE\_SLONGDATE") + " : " );

trace( "Time Format : " + this. objExtCommon.\_ext\_getLocaleFormat("Korean","LOCALE\_STIMEFORMAT") + " : " );

trace( "Time Format : " + this. objExtCommon.\_ext\_getLocaleFormat("ko","LOCALE\_STIMEFORMAT") + " : " );

trace( "Time Format : " + this. objExtCommon.\_ext\_getLocaleFormat(1042,"LOCALE\_STIMEFORMAT") + " : " );

var strFirstDay = this. objExtCommon.\_ext\_getLocaleFormat("Korean","LOCALE\_IFIRSTDAYOFWEEK");

var strFirstDay = this. objExtCommon.\_ext\_getLocaleFormat("ko","LOCALE\_IFIRSTDAYOFWEEK");

var strFirstDay = this. objExtCommon.\_ext\_getLocaleFormat(1042,"LOCALE\_IFIRSTDAYOFWEEK");

if( strFirstDay == 0 )

{

trace( "First Day 0 : " + this. objExtCommon.\_ext\_getLocaleFormat("Korean","LOCALE\_SDAYNAME1") );

}

else if( strFirstDay == 1 )

{

trace( "First Day 1 : " + this. objExtCommon.\_ext\_getLocaleFormat("Korean","LOCALE\_SDAYNAME2") );

}

else if( strFirstDay == 2 )

{

trace( "First Day 2 : " + this. objExtCommon.\_ext\_getLocaleFormat("Korean","LOCALE\_SDAYNAME3") );

}

else if( strFirstDay == 3 )

{

trace( "First Day 3 : " + this. objExtCommon.\_ext\_getLocaleFormat("Korean","LOCALE\_SDAYNAME4") );

}

else if( strFirstDay == 4 )

{

trace( "First Day 4 : " + this. objExtCommon.\_ext\_getLocaleFormat("Korean","LOCALE\_SDAYNAME5") );

}

else if( strFirstDay == 5 )

{

trace( "First Day 5 : " + this. objExtCommon.\_ext\_getLocaleFormat("Korean","LOCALE\_SDAYNAME6") );

}

else if( strFirstDay == 6 )

{

trace( "First Day 6 :" + this. objExtCommon.\_ext\_getLocaleFormat("Korean","LOCALE\_SDAYNAME7") );

}

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| strTitle | **String** | **국가 명을 문자열로 입력 하세요.**  **( 입력하는 문자열이 아래와 같아야 합니다. )**  **◈ 입력 데이터의 경우 String / Integer 중 하나의 값을 입력.**  **참고 :** [**https://msdn.microsoft.com/en-us/goglobal/bb964664.aspx**](https://msdn.microsoft.com/en-us/goglobal/bb964664.aspx)   |  |  |  | | --- | --- | --- | | **Locale Name**  **( Type : String)** | **LCID string**  **( Type : String)** | **LCID Decimal**  **( Type : Integer)** | | **Afrikaans** | af | 1078 | | **Albanian** | sq | 1052 | | **Amharic** | am | 1118 | | **Arabic (Algeria)** | ar-dz | 5121 | | **Arabic (Bahrain)** | ar-bh | 15361 | | **Arabic (Egypt)** | ar-eg | 3073 | | **Arabic (Iraq)** | ar-iq | 2049 | | **Arabic (Jordan)** | ar-jo | 11265 | | **Arabic (Kuwait)** | ar-kw | 13313 | | **Arabic (Lebanon)** | ar-lb | 12289 | | **Arabic (Libya)** | ar-ly | 4097 | | **Arabic (Morocco)** | ar-ma | 6145 | | **Arabic (Oman)** | ar-om | 8193 | | **Arabic (Qatar)** | ar-qa | 16385 | | **Arabic (Saudi Arabia)** | ar-sa | 1025 | | **Arabic (Syria)** | ar-sy | 10241 | | **Arabic (Tunisia)** | ar-tn | 7169 | | **Arabic (U.A.E)** | ar-ae | 14337 | | **Arabic (Yemen)** | ar-ye | 9217 | | **Armenian** | hy | 1067 | | **Assamese** | as | 1101 | | **Azeri** | az-az | 2092 | | **Azeri (Cyrillic)** | az-az | 2092 | | **Azeri (Latin)** | az-az | 1068 | | **Basque** | eu | 1069 | | **Belarusian** | be | 1059 | | **Bengali** | bn | 2117 | | **Bengali (Bangladesh)** | bn | 2117 | | **Bengali (India)** | bn | 1093 | | **Bosnian** | bs | 5146 | | **Bulgarian** | bg | 1026 | | **Catalan** | ca | 1027 | | **Chinese** | zh-cn | 2052 | | **Chinese (Hong Kong)** | zh-hk | 3076 | | **Chinese (Macau)** | zh-mo | 5124 | | **Chinese (PRC)** | zh-cn | 2052 | | **Chinese (Singapore)** | zh-sg | 4100 | | **Chinese (Taiwan)** | zh-tw | 1028 | | **Croatian** | hr | 1050 | | **Czech** | cs | 1029 | | **Danish** | da | 1030 | | **Divehi** | dv | 1125 | | **Dutch** | nl-nl | 1043 | | **Dutch (Belgian)** | nl-be | 2067 | | **Dutch (Netherlands)** | nl-nl | 1043 | | **English** | en-us | 1033 | | **English (Australian)** | en-au | 3081 | | **English (Belize)** | en-bz | 10249 | | **English (Canadian)** | en-ca | 4105 | | **English (Caribbean)** | en-cb | 9225 | | **English (UK)** | en-gb | 2057 | | **English (Irish)** | en-ie | 6153 | | **English (Jamaica)** | en-jm | 8201 | | **English (New Zealand)** | en-nz | 5129 | | **English (South Africa)** | en-za | 7177 | | **English (Trinidad)** | en-tt | 11273 | | **English (USA)** | en-us | 1033 | | **English (Zimbabwe)** |  | 12297 | | **Estonian** | et | 1061 | | **Faroese** | fo | 1080 | | **Farsi** | fa | 1065 | | **Finnish** | fi | 1035 | | **French** | fr-fr | 1036 | | **French (Belgian)** | fr-be | 2060 | | **French (Canadian)** | fr-ca | 3084 | | **French (Luxembourg)** | fr-lu | 5132 | | **French (Monaco)** |  | 6156 | | **French (Swiss)** | fr-ch | 4108 | | **Galician** |  | 1110 | | **Georgian** |  | 1079 | | **German (Austrian)** | de-at | 3079 | | **German** | de-de | 1031 | | **German (Liechtenstein)** | de-li | 5127 | | **German (Luxembourg)** | de-lu | 4103 | | **German (Swiss)** | de-ch | 2055 | | **Greek** | el | 1032 | | **Gujarati** | gu | 1095 | | **Hebrew** | he | 1037 | | **Hindi** | hi | 1081 | | **Hungarian** | hu | 1038 | | **Icelandic** | is | 1039 | | **Indonesian** | id | 1057 | | **Italian** | it-it | 1040 | | **Italian (Swiss)** | it-ch | 2064 | | **Japanese** | ja | 1041 | | **Kannada** | kn | 1099 | | **Kashmiri** | ks | 2144 | | **Kashmiri (South Asia)** |  | 1120 | | **Kashmiri (India)** |  | 2144 | | **Kazakh** | kk | 1087 | | **Konkani** |  | 1111 | | **Korean** | ko | 1042 | | **Korean (Extended Wansung)** | Ko-kr | 1042 | | **Kyrgyz** |  | 1088 | | **Latvian** | lv | 1062 | | **Lithuanian** | lt | 1063 | | **Macedonian** |  | 1071 | | **Malay** | ms-my | 1086 | | **Malay (Brunei Darussalam)** | ms-bn | 2110 | | **Malay (Malaysia)** | ms-my | 1086 | | **Malayalam** | ml | 1100 | | **Maltese** | mt | 1082 | | **Manipuri** |  | 1112 | | **Marathi** | mr | 1102 | | **Mongolian** | mn | 2128 | | **Nepali** | ne | 1121 | | **Norwegian (Bokmal)** | no-no | 1044 | | **Norwegian (Nynorsk)** | no-no | 2068 | | **Oriya** | or | 1096 | | **Portuguese (Brazilian)** | pt-br | 1046 | | **Portuguese** | pt-pt | 2070 | | **Punjabi** | pa | 1094 | | **Romanian** | ro | 1048 | | **Russian** | ru | 1049 | | **Sanskrit** | sa | 1103 | | **Serbian (Cyrillic)** | sr-sp | 3098 | | **Serbian** | sr-sp | 2074 | | **Sindhi** | sd | 1113 | | **Slovak** | sk | 1051 | | **Slovenian** | sl | 1060 | | **Spanish** | es-es | 1034 | | **panish (Argentina)** | es-ar | 11274 | | **Spanish (Bolivia)** | es-bo | 16394 | | **Spanish (Chile)** | es-cl | 13322 | | **Spanish (Colombia)** | es-co | 9226 | | **Spanish (Costa Rica)** | es-cr | 5130 | | **Spanish (Dominican Republic)** | es-do | 7178 | | **Spanish (Ecuador)** | es-ec | 12298 | | **Spanish (El Salvador)** | es-sv | 17418 | | **Spanish (Guatemala)** | es-gt | 4106 | | **Spanish (Honduras)** | es-hn | 18442 | | **Spanish (Mexican)** | es-mx | 2058 | | **Spanish (Nicaragua)** | es-ni | 19466 | | **Spanish (Spain - Modern Sort)** |  | 3082 | | **Spanish (Panama)** | es-pa | 6154 | | **Spanish (Paraguay** | es-py | 15370 | | **Spanish (Peru)** | es-pe | 10250 | | **Spanish (Puerto Rico)** | es-pr | 20490 | | **Spanish (Spain)** | es-es | 1034 | | **Spanish (Uruguay)** | es-uy | 14346 | | **Spanish (Venezuela)** | es-ve | 8202 | | **Swahili** | sw | 1089 | | **Swedish (Finland)** | sv-fi | 2077 | | **Swedish** | sv-se | 1053 | | **Syriac** |  | 1114 | | **Tamil** | ta | 1097 | | **Tatar** | tt | 1092 | | **Telugu** | te | 1098 | | **Thai** | th | 1054 | | **Turkish** | tr | 1055 | | **Ukrainian** | uk | 1058 | | **Unicode** | UTF-8 | 0 | | **Urdu** | ur | 1056 | | **Urdu (Pakistan)** | ur | 1056 | | **Urdu (India)** |  | 2080 | | **Uzbek** | uz-uz | 1091 | | **Uzbek (Cyrillic)** | uz-uz | 2115 | | **Uzbek (Latin)** | uz-uz | 1091 | | **Vietnamese** | vi | 1066 | |
| strParater | **String** | **Windows Api 함수 GetLocaleInfo 의 두번째 파라메터와 같은 이름**   |  |  |  | | --- | --- | --- | | **Parameter** | **설명** | **출력 데이터**  **(Sample)**  **< 한국 >** | | **LOCALE\_ILANGUAGE** | 국가별 언어 코드 | “0412”; | | **LOCALE\_SLANGUAGE** | 국가별 전체 언어 | “한국어(Korean)”; | | **LOCALE\_SABBREVLANGNAME** | 국가/지역 약식 이름 / ISO 표준 639 | “KOR”; | | **LOCALE\_SNATIVELANGNAME** | 언어 기본 이름 | “한국어”; | | **LOCALE\_ICOUNTRY** | 국제 전화. | “82”; | | **LOCALE\_SCOUNTRY** | 국가/지역의 전체 지역화 된 이름 | “대한민국(Korea)”; | | **LOCALE\_SABBREVCTRYNAME** | 국가/지역의 약식 이름/ISO 표준 3166 | “KOR”; | | **LOCALE\_SNATIVECTRYNAME** | 국가/지역의 고유 이름 | “대한민국”; | | **LOCALE\_IDEFAULTLANGUAGE** | 언어 식별자( default 식별자 ) | “0412”; | | **LOCALE\_IDEFAULTCOUNTRY** | 주요 국가/지역 코드 | “82”; | | **LOCALE\_IDEFAULTCODEPAGE** | 국가/지역 OEM 코드 페이지 | “949”; | | **LOCALE\_SLIST** | 별도의 목록 항목(쉽표)에 사용 된 문자 | **“,”;** | | **LOCALE\_SDECIMAL** | 소수 구분 기호 문자 | **".";** | | **LOCALE\_STHOUSAND** | 소수의 왼쪽에 자리 별도의 그룹에 사용되는 문자 | **“,”;** | | **LOCALE\_SGROUPING** | 소수의 왼쪽 자리의 각 그룹에 대한 크기.명시적인 크기는 각 그룹에 대해 필요하고 크기는 세미콜론으로 분리된다.마지막 값이 0 인 경우, 이전 값이 반복된다. 예를 들어, 그룹 수천, 3을 지정 0. 인도어 로케일 그룹 3으로 표시됩니다 hundredsfor 예 12,34,56,789에 의해 천 먼저 다음 그룹 2 0. | "3;0"; | | **LOCALE\_IDIGITS** | 소수 자릿수의 수 | "2"; | | **LOCALE\_ILZERO** | 소수 필드에 0을 선두에 대한 지정자. 이 문자열에 허용되는 최대 문자 수는 2 개이다.지정은 다음 값 중 하나 일 수 있습니다.  Value Meaning  0 : No leading zeros for months  1 : Leading zeros for months | "0"; | | **LOCALE\_SNATIVEDIGITS** | 기본 등가물 ASCII 코드 0 ~ 9 | "0123456789" | | **LOCALE\_SCURRENCY** | 국가/지역 통화 기호. | "₩" | | **LOCALE\_SINTLSYMBOL** | ISO 4217에 규정 된 국제 통화 기호 | "KRW" | | **LOCALE\_SMONDECIMALSEP** | 문자 (들) 통화 소수 구분 기호로 사용. 이 문자열에 허용되는 최대 문자 수는 4 | "."; | | **LOCALE\_SMONTHOUSANDSEP** | 문자 (들), 소수의 왼쪽에 자리 그룹 간의 통화 구분 기호로 사용. 이 문자열에 허용되는 최대 문자 수는 4이다. | ","; | | **LOCALE\_SMONGROUPING** | 소수의 왼쪽에 통화 자리의 각 그룹에 대한 크기.명시적인 크기는 각 그룹에 대해 필요하고 크기는 세미콜론으로 분리된다.마지막 값이 0 인 경우, 이전 값이 반복된다. 예를 들어, 그룹 수천, 3을 지정 0. 인도어 그룹 3으로 표시됩니다 hundredsfor 예에 의한 천 먼저 다음 그룹, 12,34,56,789, 2 0. 이 문자열에 허용되는 최대 문자 수는 4이다. | "3;0"; | | **LOCALE\_ICURRDIGITS** | 현지 통화 형식의 소수 자릿수의 수. 이 문자열에 허용되는 최대 문자 수는 3이다. | "0"; | | **LOCALE\_IINTLCURRDIGITS** | 국제 통화 형식의 소수 자릿수의 수. 이 문자열에 허용되는 최대 문자 수는 3이다. | "2"; | | **LOCALE\_ICURRENCY** | 양의 통화 모드에서 통화 기호의 위치. 이 문자열에 허용되는 최대 문자 수는 2 개이다. 모드는 다음 값 중 하나 일 수 있습니다.   |  |  | | --- | --- | | **Value** | **Meaning** | | 0 | Prefix, no separation, for example $1.1 | | 1 | Suffix, no separation, for example 1.1$ | | 2 | Prefix, 1-character separation, for example $ 1.1 | | 3 | Suffix, 1-character separation, for example 1.1 $ | | “0”; | | **LOCALE\_INEGCURR** | 음의 통화 모드. 이 문자열에 허용되는 최대 문자 수는 3이다.모드는 다음 값 중 하나 일 수 있습니다.   |  |  | | --- | --- | | **Mode** | **Format for a negative currency** | | 0 | Left parenthesis, monetary symbol, number, right parenthesis.  Example: ($1.1) | | 1 | Negative sign, monetary symbol, number. Example: -$1.1 | | 2 | Monetary symbol, negative sign, number. Example: $-1.1 | | 3 | Monetary symbol, number, negative sign. Example: $1.1- | | 4 | Left parenthesis, number, monetary symbol, right parenthesis. Example: (1.1$) | | 5 | Negative sign, number, monetary symbol. Example: -1.1$ | | 6 | Number, negative sign, monetary symbol. Example: 1.1-$ | | 7 | Number, monetary symbol, negative sign. Example: 1.1$- | | 8 | Negative sign, number, space, monetary symbol (like #5, but with a space before the monetary symbol). Example: -1.1 $ | | 9 | Negative sign, monetary symbol, space, number (like #1, but with a space after the monetary symbol). Example: -$ 1.1 | | 10 | Number, space, monetary symbol, negative sign (like #7, but with a space before the monetary symbol). Example: 1.1 $- | | 11 | Monetary symbol, space, number, negative sign (like #3, but with a space after the monetary symbol). Example: $ 1.1- | | 12 | Monetary symbol, space, negative sign, number (like #2, but with a space after the monetary symbol). Example: $ -1.1 | | 13 | Number, negative sign, space, monetary symbol (like #6, but with a space before the monetary symbol). Example: 1.1- $ | | 14 | Left parenthesis, monetary symbol, space, number, right  parenthesis (like #0, but with a space after the monetary symbol). Example: ($ 1.1) | | 15 | Left parenthesis, number, space, monetary symbol, right parenthesis (like #4, but with a space before the monetary symbol). Example: (1.1 $) | | “1”; | | **LOCALE\_SDATE** | 날짜 구분 기호에 대한 문자 (들). 이 문자열에 허용되는 최대 문자 수는 4이다. | "-"; | | **LOCALE\_STIME** | 시간 구분에 대한 문자 (들). 이 문자열에 허용되는 최대 문자 수는 4이다. | ":"; | | **LOCALE\_SSHORTDATE** | 이 로케일에 대한 간단한 날짜 형식 문자열. 문자열은 일, 월, 년에 정의 된 년, 월, 일의 포맷 영상의 조합으로 구성하고, 국가 별 언어 지원 상수의 시대 형식 그림 테이블 수 있습니다. 이 문자열에 허용되는 문자의 최대 수는 80입니다. | "yy-MM-dd"; | | **LOCALE\_SLONGDATE** | 이 로케일의 긴 날짜 형식 문자열. 문자열은 국가 별 언어 지원 상수의 년, 월, 일, 그리고 시대 형식 그림 테이블과 작은 따옴표로 묶인 모든 문자열에 정의 된 년, 월, 일의 포맷 영상의 조합으로 구성 할 수 있습니다. 작은 따옴표 문자로 지정 남아있다. 이 문자열에 허용되는 문자의 최대 수는 80입니다. | "yyyy년 M월 d일” | | **LOCALE\_IDATE** | 간단한 날짜 형식 - 주문 지정자. 이 문자열에 허용되는 최대 문자 수는 2 개이다.지정 값은 다음 중 하나가 될 수 있습니다   |  |  | | --- | --- | | **Value** | **Meaning** | | 0 | Month-Day-Year | | 1 | Day-Month-Year | | 2 | Year-Month-Day | | "2"; | | **LOCALE\_ILDATE** | 긴 날짜 형식   |  |  | | --- | --- | | **Value** | **Meaning** | | 0 | Month-Day-Year | | 1 | Day-Month-Year | | 2 | Year-Month-Day | | "2"; | | **LOCALE\_ITIME** | 지역 시간 지정자 | "1"; | | **LOCALE\_ICENTURY** | 만 짧은 날짜 2 자리 또는 4 자리의 세기에 대한 지정자. 이 문자열에 허용되는 최대 문자 수는 2 개이다.지정은 다음 값 중 하나 일 수 있습니다.   |  |  | | --- | --- | | **Value** | **Meaning** | | 0 | Abbreviated 2-digit century | | 1 | Full 4-digit century | | "0"; | | **LOCALE\_ITLZERO** | 시간 필드에 0을 선두에 대한 지정자. 이 문자열에 허용되는 최대 문자 수는 2 개이다. 지정은 다음 값 중 하나 일 수 있습니다.   |  |  | | --- | --- | | **Value** | **Meaning** | | 0 | No leading zeros for hours. | | 1 | Leading zeros for hours | | "1"; | | **LOCALE\_IDAYLZERO** | 짧은 날짜를 하루 필드에 0을 선두에 대한 지정자   |  |  | | --- | --- | | **Value** | **Meaning** | | 0 | No leading zeros for days | | 1 | Leading zeros for days | | "1"; | | **LOCALE\_IMONLZERO** | 짧은 날짜 달 필드에 0을 선두에 대한 지정자.   |  |  | | --- | --- | | **Value** | **Meaning** | | 0 | No leading zeros for months | | 1 | Leading zeros for months | | "1"; | | **LOCALE\_S1159** | 오전 지정자의 문자열. 이 문자열에 허용되는 최대 문자 수는 아홉이다. | "오전"; | | **LOCALE\_S2359** | 오후 지정자의 문자열. 이 문자열에 허용되는 최대 문자 수는 아홉이다. | "오후"; | | **LOCALE\_SDAYNAME1** | 월요일 에 대한 기본 긴 이름. | "월요일"; | | **LOCALE\_SDAYNAME2** | 화요일 에 대한 기본 긴 이름. | "화요일"; | | **LOCALE\_SDAYNAME3** | 수요일 에 대한 기본 긴 이름. | "수요일"; | | **LOCALE\_SDAYNAME4** | 목요일 에 대한 기본 긴 이름. | "목요일"; | | **LOCALE\_SDAYNAME5** | 금요일 에 대한 기본 긴 이름. | "금요일"; | | **LOCALE\_SDAYNAME6** | 토요일 에 대한 기본 긴 이름. | "토요일"; | | **LOCALE\_SDAYNAME7** | 일요일 에 대한 기본 긴 이름. | "일요일"; | | **LOCALE\_SABBREVDAYNAME1** | 월 에 대한 기본 짧은 이름 | "월"; | | **LOCALE\_SABBREVDAYNAME2** | 화 에 대한 기본 짧은 이름 | "화"; | | **LOCALE\_SABBREVDAYNAME3** | 수 에 대한 기본 짧은 이름 | "수"; | | **LOCALE\_SABBREVDAYNAME4** | 목 에 대한 기본 짧은 이름 | "목"; | | **LOCALE\_SABBREVDAYNAME5** | 금 에 대한 기본 짧은 이름 | "금"; | | **LOCALE\_SABBREVDAYNAME6** | 토 에 대한 기본 짧은 이름 | "토"; | | **LOCALE\_SABBREVDAYNAME7** | 일 에 대한 기본 짧은 이름 | "일"; | | **LOCALE\_SMONTHNAME1** | 1월 기본 긴 이름 | "1월”; | | **LOCALE\_SMONTHNAME2** | 2월 기본 긴 이름 | “2월”; | | **LOCALE\_SMONTHNAME3** | 3월 기본 긴 이름 | “3월”; | | **LOCALE\_SMONTHNAME4** | 4월 기본 긴 이름 | “4월”; | | **LOCALE\_SMONTHNAME5** | 5월 기본 긴 이름 | “5월”; | | **LOCALE\_SMONTHNAME6** | 6월 기본 긴 이름 | “6월”; | | **LOCALE\_SMONTHNAME7** | 7월 기본 긴 이름 | “7월”; | | **LOCALE\_SMONTHNAME8** | 8월 기본 긴 이름 | “8월”; | | **LOCALE\_SMONTHNAME9** | 9월 기본 긴 이름 | “9 월”; | | **LOCALE\_SMONTHNAME10** | 10월 기본 긴 이름 | “10 월”; | | **LOCALE\_SMONTHNAME11** | 11월 기본 긴 이름 | “11 월”; | | **LOCALE\_SMONTHNAME12** | 12월 기본 긴 이름 | “12월”; | | **LOCALE\_SABBREVMONTHNAME1** | 1월 기본 이름 | “1월“; | | **LOCALE\_SABBREVMONTHNAME2** | 2월 기본 이름 | “2월“; | | **LOCALE\_SABBREVMONTHNAME3** | 3월 기본 이름 | “3월”; | | **LOCALE\_SABBREVMONTHNAME4** | 4월 기본 이름 | “4월”; | | **LOCALE\_SABBREVMONTHNAME5** | 5월 기본 이름 | “5월”; | | **LOCALE\_SABBREVMONTHNAME6** | 6월 기본 이름 | “6월”; | | **LOCALE\_SABBREVMONTHNAME7** | 7월 기본 이름 | “7월”; | | **LOCALE\_SABBREVMONTHNAME8** | 8월 기본 이름 | “8월”; | | **LOCALE\_SABBREVMONTHNAME9** | 9월 기본 이름 | “9월”; | | **LOCALE\_SABBREVMONTHNAME10** | 10월 기본 이름 | “10월”; | | **LOCALE\_SABBREVMONTHNAME11** | 11월 기본 이름 | “11월”; | | **LOCALE\_SABBREVMONTHNAME12** | 12월 기본 이름 | “12월”; | | **LOCALE\_SPOSITIVESIGN** | 양의 기호에 대한 문자열 값입니다. 이 문자열에 허용되는 최대 문자 수는 다섯입니다. | ""; | | **LOCALE\_SNEGATIVESIGN** | 음의 부호에 대한 문자열 값입니다. 이 문자열에 허용되는 최대 문자 수는 다섯입니다. | "-"; | | **LOCALE\_IPOSSIGNPOSN** | 양의 값에 대한 서식 인덱스입니다. 이 인덱스는 인덱스 0을 사용하지 않는다는 점을 제외 LOCALE\_INEGSIGNPOSN 것과 동일한 값을 사용한다. 이 문자열에 허용되는 최대 문자 수는 2 개이다. | "3"; | | **LOCALE\_INEGSIGNPOSN** | 통화 가치에 음의 부호에 대한 서식 인덱스입니다. 이 문자열에 허용되는 최대 문자 수는 2 개이다. 인덱스는 다음 값 중 하나 일 수 있습니다.   |  |  | | --- | --- | | **Value** | **Meaning** | | 0 | Parentheses surround the amount and the monetary symbol. | | 1 | The sign precedes the number. | | 2 | The sign follows the number. | | 3 | The sign precedes the monetary symbol. | | 4 | The sign follows the monetary symbol. | | "3"; | | **LOCALE\_IPOSSYMPRECEDES** | 양의 금전적 가치에 통화 기호의 위치. 그것을 다음 경우 통화 기호가 긍정적 인 양, 제로에 선행하는 경우 이 값은 1이다. 이 문자열에 허용되는 최대 문자 수는 2 개이다. | "1"; | | **LOCALE\_IPOSSEPBYSPACE** | 양의 금전적 가치에 통화 기호의 분리. 그렇지 않은 경우 통화 기호가 긍정적 인 양, 0에서 공백으로 구분되어있는 경우이 값은 1이다. 이 문자열에 허용되는 최대 문자 수는 2 개이다 | "0"; | | **LOCALE\_INEGSYMPRECEDES** | 음의 금전적 가치에 통화 기호의 위치. 그것을 다음 경우 통화 기호가 음수 금액, 0을 앞에 경우이 값은 1이다. 이 문자열에 허용되는 최대 문자 수는 2 개이다. | "1"; | | **LOCALE\_INEGSEPBYSPACE** | 금전적 가치에 음의 부호의 분리. 그렇지 않은 경우 통화 기호가 음수 금액, 0에서 공백으로 구분되어있는 경우이 값은 1이다. 이 문자열에 허용되는 최대 문자 수는 2 개이다. | "0"; | | **LOCALE\_SENGCOUNTRY** | 국가 / 지역의 전체 영어 이름. 이 작업은 항상 ASCII 127 문자 부분 집합에 매핑 할 수있는 문자로 제한됩니다. | "Korea"; | | **LOCALE\_SENGLANGUAGE** | 이것은 항상 ASCII 127 문자 부분 집합에 매핑 할 수있는 문자로 제한됩니다 표준화기구 (ISO) 표준 639에 대한 국제기구에서 언어의 전체 영어 이름. 이것은 항상 LOCALE\_SLANGUAGE의 영어 버전에 해당되지 않습니다. | "Korean"; | | **LOCALE\_IFIRSTDAYOFWEEK** | 일주일의 첫날 지정자. 이 문자열에 허용되는 최대 문자 수는 2 개이다. 지정은 다음 값 중 하나 일 수 있습니다.   |  |  | | --- | --- | | **Value** | **Meaning** | | 0 | LOCALE\_SDAYNAME1 | | 1 | LOCALE\_SDAYNAME2 | | 2 | LOCALE\_SDAYNAME3 | | 3 | LOCALE\_SDAYNAME4 | | 4 | LOCALE\_SDAYNAME5 | | 5 | LOCALE\_SDAYNAME6 | | 6 | LOCALE\_SDAYNAME7 | | "6"; | | **LOCALE\_IFIRSTWEEKOFYEAR** | 해의 최초의 주에 대한 지정자. 이 문자열에 허용되는 최대 문자 수는 2 개이다. 지정은 다음 값 중 하나 일 수 있습니다.   |  |  | | --- | --- | | **Value** | **Meaning** | | 0 | Week containing 1/1 is the first week of that year. | | 1 | First full week following 1/1 is the first week of that year. | | 2 | First week containing at least four days is the first week of that year. | | "0"; | | **LOCALE\_IDEFAULTANSICODEPAGE** | 이 로케일과 관련된 미국 국립 표준 협회 (ANSI) 코드 페이지.로케일은 ANSI 코드 페이지를 사용하지 않는 경우, 값은 0이 문자열에 허용 된 최대 문자 수는 여섯이다. ANSI 및 다른 코드 페이지의 목록은 코드 페이지 식별자를 참조하십시오. | "949"; | | **LOCALE\_INEGNUMBER** | 음의 숫자 모드, 음수에 대한 형식입니다. 이 문자열에 허용되는 최대 문자 수는 2 개이다.모드는 다음 값 중 하나 일 수 있습니다   |  |  | | --- | --- | | **Value** | **Format** | | 0 | Left parenthesis, number, right parenthesis. Example: (1.1) | | 1 | Negative sign, number. Example: -1.1 | | 2 | Negative sign, space, number. Example: - 1.1 | | 3 | Number, negative sign. Example: 1.1- | | 4 | Number, space, negative sign. Example: 1.1 - | | "1"; | | **LOCALE\_STIMEFORMAT** | 시간이 로케일 문자열을 포맷. 문자열은 국가 별 언어 지원 상수의 시, 분, 시간, 분에 정의 된 두 번째 형식의 사진, 그리고 두 번째 형식 그림 테이블의 조합으로 구성 할 수 있습니다. 이 문자열에 허용되는 문자의 최대 수는 80입니다. | "HH:mm:ss"; | | **LOCALE\_ITIMEMARKPOSN** | 시간 마커 문자열 (오전 또는 오후)에 선행 또는 시간 문자열을 다음과 하는 지 여부를 나타내는 지정자. 레지스트리 값은 이전 Windows 아시아 버전과의 호환성을 위해 iTimePrefix입니다.지정은 다음 값 중 하나를 취할 수 있습니다.   |  |  | | --- | --- | | **Value** | **Meaning** | | 0 | Use as suffix. | | 1 | Use as prefix. | | "0"; | | **LOCALE\_ICALENDARTYPE** | 현재 달력 유형입니다. 이 문자열에 허용되는 최대 문자 수는 2 개이다. 이 유형은 다음 값 중 하나 일 수 있습니다.   |  |  | | --- | --- | | **Value** | **Meaning** | | 1 | Gregorian (localized) | | 2 | Gregorian (English strings always) | | 3 | Year of the Emperor (Japan) | | 4 | Taiwan Calendar | | 5 | Tangun Era (Korea) | | 6 | Hijri (Arabic lunar) | | 7 | Thai | | 8 | Hebrew (Lunar) | | 9 | Gregorian Middle East French calendar | | 10 | Gregorian Arabic calendar | | 11 | Gregorian Transliterated English calendar | | 12 | Gregorian Transliterated French calendar | | "1"; | | **LOCALE\_IOPTIONALCALENDAR** | 이 LCID 사용할 수 있습니다 옵션 달력 형식. 이것은 다음의 일정 유형 값 중 하나가 될 수 있다.   |  |  | | --- | --- | | **Value** | **Meaning** | | 0 | No additional types valid | | 1 | Gregorian (localized) | | 2 | Gregorian (English strings always) | | 3 | Year of the Emperor (Japan) | | 4 | Taiwan Calendar | | 5 | Tangun Era (Korea) | | 6 | Hijri (Arabic lunar) | | 7 | Thai | | 8 | Hebrew (Lunar) | | 9 | Gregorian Middle East French calendar | | 10 | Gregorian Arabic calendar | | 11 | Gregorian-transliterated English calendar | | 12 | Gregorian-transliterated French calendar | | "2"; | | **LOCALE\_SMONTHNAME13** | 13 월의 기본 이름은 경우가 존재한다. LOCALE\_SMONTHNAME1에 대한 참고 사항을 참조하십시오. | ""; | | **LOCALE\_SABBREVMONTHNAME13** | 13 월에 대한 기본 단축 이름은 경우가 존재한다. | ""; | |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| String  Or  Integer | 성공 : String : 취득 된 Data  실패 : integer : -1 을 리턴합니다. |

**Method Name : \_ext\_getDefaultLocaleInfo(strType)**

▶ **Description**

* **피시에 설치된 기본 언어 설정.**

▶ **Syntax**

trace(this.objExtCommon.\_ext\_getDefaultLocaleInfo("GetSystemDefaultUILanguage"));

trace(this.objExtCommon.\_ext\_getDefaultLocaleInfo("GetSystemDefaultLCID"));

trace(this.objExtCommon.\_ext\_getDefaultLocaleInfo("GetSystemDefaultLangID"));

trace(this.objExtCommon.\_ext\_getDefaultLocaleInfo("GetUserDefaultLangID"));

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **strType** | **String** | 인스톨된 언어 값을 구할 때.   |  |  | | --- | --- | | ***Parameter*** | ***설명*** | | GetSystemDefaultUILanguage | 인스톨된 언어 값을 구할 때 | | GetSystemDefaultLCID | 시스템 locate 기본 언어 값을 구할 때. | | GetSystemDefaultLangID | 시스템 기본 언어 값을 구할 때. | | GetUserDefaultLangID | 유저 기본 언어 값을 구할 때. | |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Array | [0]. : 코드 페이지  [1]. : 국가명  [2] : 국가 코드 (short string )  [3] : LCID |

**Method Name : \_ext\_setURLencode(strEncode, strType)**

▶ **Description**

* **입력된 문자열을 URL에 사용할 수 있는 형태로 encode 하는 Basic API 입니다.**

**영숫자 및 "$", "\_", ".", "!", "\*", "'", "(", ")" 을 제외한 모든 문자를 %HH (H는 16진수 문자)의 형태로 변경합니다..**

▶ **Syntax**

//Default : ANSI

var ansi = this. objExtCommon**.\_ext\_setURLencode**("abcdefg/한%글##테%%스^^트123;;;");

trace( "ANSI encode:: " + ansi);

ansi=this.objExtCommon.\_**ext\_setURLencod**e("abcdefg/한%글##테%%스^^트123;;;4","CP\_ACP");

trace( "ANSI encode:: " + ansi);

ansi = this.objExtCommon**.\_ext\_setURLencode**("abcdefg/한%글##테%%스^^트123;;;","ANSI");

trace( "ANSI encode:: " + ansi);

trace( "ANSI Decode ::" + this. objExtCommon.\_ext\_getURLdecode(ansi, "ANSI"));

//////////////////////////////////////////////////////////////////////////

var encode=this. objExtCommon**.\_ext\_setURLencode**("abcdefg/한%글##테%%스^^트123;;;","utf-8");

trace( "UTF-8 encode:: " + encode);

encode=this. objExtCommon**.\_ext\_setURLencode**("abcdefg/한%글##테%%스^^트123;;;","UTF-8");

trace( "UTF-8 encode:: " + encode);

trace( "UTF-8 Decode ::" +this. objExtCommon.\_ext\_getURLdecode(encode));

//////////////////////////////////////////////////////////////////////////

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **strEncode** | **String** | URL을 통해 전송할 문자열. |
| **strType** | **String** | Format 지정.  **Default** : **ANSI** ( MiPlatform 과 동일 )  ANSI Format : ANSI , ansi  UTF-8 Format : UTF-8 , utf-8 ( XPlatform 과 동일 ) |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| String | URL에 사용가능하도록 Encode 된 문자열. |

**Method Name : \_ext\_getURLdecode (strDecode, strType)**

▶ **Description**

**입력된 문자열을 URL에 사용할 수 있는 형태에서 실제 문자열로 Decode 하는 API 입니다.**

▶ **Syntax**

//Default : ANSI

var ansi = this. objExtCommon.\_ext\_setURLencode("abcdefg/한%글##테%%스^^트123;;;");

trace( "ANSI encode:: " + ansi);

ansi=this.objExtCommon.\_ext\_setURLencode("abcdefg/한%글##테%%스^^트123;;;4","CP\_ACP");

trace( "ANSI encode:: " + ansi);

ansi = this.objExtCommon.\_ext\_setURLencode("abcdefg/한%글##테%%스^^트123;;;","ANSI");

trace( "ANSI encode:: " + ansi);

trace( "UTF-8 Decode ::" +this.objExtCommon**.\_ext\_getURLdecode(**encode));

trace( "ANSI Decode ::" + this.objExtCommon**.\_ext\_getURLdecode**(ansi, "ANSI"));

//////////////////////////////////////////////////////////////////////////

var encode = this.objExtCommon.\_ext\_setURLencode("abcdefg/한%글##테%%스^^트123;;;","utf-8");

trace( "UTF-8 encode:: " + encode);

var encode = this.objExtCommon.\_ext\_setURLencode("abcdefg/한%글##테%%스^^트123;;;","UTF-8");

trace( "UTF-8 encode:: " + encode);

trace( "UTF-8 Decode ::" +this. objExtCommon**.\_ext\_getURLdecode(**encode,”utf-8”));

//////////////////////////////////////////////////////////////////////////

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **strEncode** | **String** | URL에 사용가능한 형태로 Encode 된 문자열. |
| **strType** | **String** | Format 지정.  **Default** : **ANSI** ( MiPlatform 과 동일 )  ANSI Format : ANSI , ansi  UTF-8 Format : UTF-8 , utf-8 ( XPlatform 과 동일 ) |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| String | Decode 된 문자열. Encode 되기 전의 원래 문자열 |

**Method Name : \_ext\_setBase64encode(strEncode, strType)**

▶ **Description**

* **입력된 문자열을 Base64 Encode 하는 기능**
* **디폴트 ANSI 로 문자열을 변환 후 Encode 함.**
  + **Nexacro API base64encode 의 경우 UTF-8 이 기본값.**

▶ **Syntax**

//default :: ANSI

var enANSIE = this.objExtCommon**.\_ext\_setBase64encode("한글테스트");**

trace( "Base64 encode::" +enANSIE);

trace( "Base64 decode::" + this.bjExtCommon.\_**ext\_getBase64decode**(enANSIE) );

//UTF-8 , utf-8

var enUTF8 = this.objExtCommon**.\_ext\_setBase64encode("한글테스트", "UTF-8");**

trace( "Base64 encode::" +enUTF8);

trace( "Base64 decode::" + this.objExtCommon.\_ext\_getBase64decode(enUTF8,"UTF-8") );

**}**

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **strEncode** | **String** | Base64 로 인코딩 하려는 문자열 |
| **strType** | **String** | Format 지정.  **Default** : **ANSI** ( MiPlatform 과 동일 )  ANSI Format : ANSI , ansi  UTF-8 Format : UTF-8 , utf-8 ( XPlatform 과 동일 ) |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| String | * Encode 된 문자열. |

**Method Name : \_ext\_getBase64decode (strEncode, strType)**

▶ **Description**

* **Base64encode 된 문자열을 Base64 decode 하는 기능**
* **디폴트 ANSI 로 문자열을 변환 후 decode 함.**
  + **Nexacro API base64dncode 의 경우 UTF-8 이 기본값.**

▶ **Syntax**

//default :: ANSI

var enANSIE = this. objExtCommon.\_ext\_setBase64encode("한글테스트");

trace( "Base64 encode::" +enANSIE);

trace( "Base64 decode::" +this.objExtCommon**.\_ext\_getBase64decode(enANSIE) );**

//UTF-8 , utf-8

var enUTF8 = this.objExtCommon.\_ext\_setBase64encode("한글테스트", "UTF-8");

trace( "Base64 encode::" +enUTF8);

trace( "Base64 decode::" + this.objExtCommon**.\_ext\_getBase64decode(enUTF8,"UTF-8")** );

**}**

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **strEncode** | **String** | Base64 로 인코딩 된 문자열 |
| **strType** | **String** | Format 지정.  **Default** : **ANSI** ( MiPlatform 과 동일 )  ANSI Format : ANSI , ansi  UTF-8 Format : UTF-8 , utf-8 ( XPlatform 과 동일 ) |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| String | * Decode 된 문자열. |

**Method Name : \_ext\_existFile (string filepath)**

▶ **Description**

* + **파일 존재 유무**

▶ **Syntax**

var ret = this.objExtCommon.\_ext\_existFile(strFilePath);

if(ret)

trace( "Success " + ret );

else

trace( "Fail " + ret );

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| strFilePath | **string** | **File path** |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | TRUE : 존재  FALSE : 없음 |

## **Method Name : \_ext\_setPrintOrientation (int opt)**

▶ **Description**

* + **프린트 페이지 출력 방향 설정. 1 :세로 , 2: 가로**

▶ **Syntax**

this.objExtCommon.\_ext\_setPrintOrientation(1); //세로

this.objExtCommon.\_ext\_setPrintOrientation(2); // 가로

;

**}**

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| opt | **integer** | **1 :세로 2: 가로** |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | TRUE : 성공  FALSE : 실패 |

## **Method Name : \_ext\_getFindMiplatformOnlyOne(string Pojectkey, string sitekey, string MiplatformProcName)**

▶ **Description**

* + **해당 프로젝트 키 와 사이트 키로 MiPLATFORM,EXE 의 활성화 유무 확인.**

▶ **Syntax**

var bRet = this.objExtCommon.\_ext\_getFindMiplatformOnlyOne("TobeNext\_U", "", "Miplatform320u") ;

if(bRet)

{

trace("Success 해당 프로그램이 활성화 되어 있지 않습니다.");

}

else

{

trace("Failed 해당 프로그램이 활성화 되어 있습니다.");

}

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| Pojectkey | string | MiPlatform 활성화 Project Regstry key  ( -K 해당 하는 키 값) |
| sitekey | string | MiPlatform 활성화 사용 되는 키 ( -S 해당하는 키 값 )  Local에 있는 Script 파일을 사용하는 경우 Script 파일이 있는 경로 |
| MiplatformProcName | string | 해당 키로 사용 되는 MiPALTFORM 프로세스 명.   |  |  | | --- | --- | | arg | 설명 | | Miplatform320 | Miplatform 320 ANSI 재품 | | Miplatform320U | Miplatform 320 Unicode 재품 | | Miplatform330 | Miplatform 330 ANSI 재품 | | Miplatform33U | Miplatform 330 Unicode 재품 | |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | TRUE : Success 해당 프로그램이 활성화 되어 있지 않습니다.  FALSE : Failed 해당 프로그램이 활성화 되어 있습니다. |

## **Method Name : \_ext\_getLengthB( strData, strCodePage)**

▶ **Description**

* 입력값 형태에 따라서 길이 또는 범위를 구하는 Basic API 입니다.
* 문자열의 Binary Data의 크기등을 byte 단위로 구합니다

▶ **Syntax**

var AnsiLenb = this.objExtCommon**.\_ext\_getLengthB**("Text가나다");

trace( "ANSI LengthB " + AnsiLenb );

var Utf8Lenb = this.objExtCommon**.\_ext\_getLengthB**("Text가나다","utf-8");

trace( "UTF-8 LengthB " + Utf8Lenb );;

**}**

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **strData** | **STRING** | Binary Data의 length 를 구하려는 string data |
| **strCodePage** | **STRING** | 해당 코드 페이지를 이용하여 변환 하려는 코드 값.  Default : ANSI |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| INTEGER | 해당 코드 페이지를 이용하여 변환한 String 의 Binary 데이터의 길이 |

## **Method Name : \_ext\_getGlobalMemoryStatusEx()**

▶ **Description**

* **호출한 시점에서의 PC 의 메모리 상태를 취득합니다.**

**참고 :**

* + **Windows API 함수인 [ GlobalMemoryStatusEx ] 이용.**

▶ **Syntax**

var ret;

ret = this.objExtCommon**.\_ext\_getGlobalMemoryStatusEx**(); //호출한 시점에서의 자신의 Memory Info

var Success = **ret[0];** // 함수 호출의 성공 여부.

var ullTotalPhys = 0;

var ullAvailPhys = 0;

var ullTotalPageFile = 0;

var ullAvailPageFile = 0;

var ullTotalVirtual = 0;

var ullAvailVirtual = 0;

if( Success == 1) // 함수 호출의 성공 여부.

{

ullTotalPhys = **ret[1];**

ullAvailPhys = **ret[2]**;

ullTotalPageFile = **ret[3];**

ullAvailPageFile = **ret[4];**

ullTotalVirtual = **ret[5];**

ullAvailVirtual = **ret[6];**

trace( "Total Physical Memory [ ullTotalPhys ] : " + (ullTotalPhys) + " KB");

trace( "Free Physical Memory [ ullAvailPhys ] : " + ullAvailPhys + " KB" );

trace( "Total Virtual Memory (Physical + Page Files size) [ ullTotalPageFile ] : " + (ullTotalPageFile) + " KB");

trace( "Virtual Memory [ ullAvailPageFile ] : " + (ullAvailPageFile) + " KB");

trace( "Total Addr. Space (Current Process) [ ullTotalVirtual ] : " + (ullTotalVirtual)+ " KB");

trace( "Free Addr. Space (Current Process) [ ullAvailVirtual ] : " + ullAvailVirtual + " KB" );

}

else

{

trace("Fail " + Success );

}

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
|  |  | 없음 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Array  [Integer] | <메모리 정보를 array로 리턴 합니다.>   * [0] :   함수 호출의 성공 여부 [-1 : 실패 , 1 : 성공] [ integer ]   * [1] : ( ullTotalPhys )   시스템에 설치된 물리 메모리의 총 바이트 수를 나타낸다.  (물론 정확한 값보다 조금 낮다. 이유는 부팅 과정에서 논페이지 풀을 위한 저장소를 일부 예약해 두는데, 이 메모리는 커널에 의해서도 가용 메모리로 고려되지 않는 메모리다. )  [리턴 되는 데이터의 타입은 **KB** 입니다. ]   * [2] : (ullAvailPhys)   할당 가능한 물리적 메모리의 전체 크기를 가리킨다.  [리턴 되는 데이터의 타입은 **KB** 입니다. ]   * [3] : (ullTotalPageFile)   하드 디스크상에 존재하는 페이징 파일이 수용할 수 있는 최대 바이트 수를 나타낸다.  [리턴 되는 데이터의 타입은 **KB** 입니다. ]   * [4] : (ullAvailPageFile)   현재까지 커밋 되지 않은 ( 실질적 으로 이용 가능한) 페이징 파일 크기를 가리킨다.  [리턴 되는 데이터의 타입은 **KB** 입니다. ]   * [5] : (ullTotalVirtual)   각 프로세스 주소 공간 내의 전체 메모리 크기를 나타낸다.  [리턴 되는 데이터의 타입은 **KB** 입니다. ]   * [6] : (ullAvailVirtual)   이 값은 각 프로세스 주소 공간의 사용 가능한 메모리크기를 나타낸다.  [리턴 되는 데이터의 타입은 **KB** 입니다. ] |

## **Method Name : \_ext\_getProcessMemoryInfo ()**

▶ **Description**

* + 호출한 시점에서의 자신의 프로세스(Nexacro) 메모리 상태를 취득합니다.

참고 :

Windows API 함수인 [GetProcessMemoryInfo ] 를 이용.

▶ **Syntax**

var ret;

ret = this.objExtCommon**.\_ext\_getProcessMemoryInfo();** //호출한 시점에서의 자신의 Nexacro Process Memory Info

var Success = **ret[0];** // 함수 호출의 성공 여부.

var PeekWorkingSetSzie = 0;

var WorkingSetSize = 0;

var PrivateUsage = 0;

var WorkingSetPrivate = 0; **//2017.03.06 add**

if( Success == 1) // 함수 호출의 성공 여부.

{

PeekWorkingSetSzie **= ret[1];** // byte 수 / 애플리케이션이 시작된후 가장 많은 램을 사용한 순간의 크기 [ **byte** ]

WorkingSetSize **= ret[2]**; // byte 수 / GetProcessMemoryInfo 함수가 호출된 순간의 램 사용 크기 [ **byte** ]

PrivateUsage = **ret[3];** // byte 수 / 애플리케이션에서 명시적으로 할당한 메모리의 크기 [ **byte** ]

WorkingSetPrivate = **ret[4];** // KB 로 전송 됨 / 작업 관리자 메모리. ( 개인 작업 집합 ) [ KB / kilo byte ]

trace("PeekWorkingSetSzie : " + **(PeekWorkingSetSzie/1024) + " KB")**;

trace("WorkingSetSize : " **+ (WorkingSetSize/1024) + " KB");**

trace("PrivateUsage : " + **(PrivateUsage/1024) + " KB");**

**//[ 2017.03.06 add ]**

trace("WorkingSetPrivate : " + (WorkingSetPrivate/1024) + " MB");

}else{

trace("Fail " + Success );

}

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
|  |  |  |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Array | <호출한 자신의 프로세스 메모리 정보를 취득합니다.>   * [0] :   함수 호출의 성공 여부 [ -1 : 실패 , 1 : 성공 ]   * [1] : (PeakWorkingSetSize)   애플리케이션이 시작된후 가장 많은 램을 사용한 순간의 크기 [ **byte** ]   * [2] :( WorkingSetSize)   GetProcessMemoryInfo 함수가 호출된 순간의 램 사용 크기 [ **byte** ]   * [3] :( PrivateUsage)   애플리케이션에서 명시적으로 할당한 메모리의 크기 [ **byte** ]   * [4] :( WorkingSetPrivate) //[2017.03.06 add]   // KB 로 전송 됨 / 작업 관리자 메모리. ( 개인 작업 집합 ) [ **KB / kilo byte** ] |

## **Method Name : \_ext\_getUseFont(szFont)**

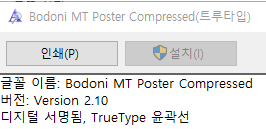
▶ **Description**

* **현재 피시에 폰트 설치 유무 확인 합니다.**
* **폰트 이름 적용 시. 아래 이미지와 같이 실제 폰트 명을 입력 하여야 합니다.**

1. **제어판에서 보여지는 이름**



1. **실제 폰트 명**



**참고 : jquery 사용 시..**

**[ https://github.com/beseku/jquery.font ]**

▶ **Syntax**

var szfont = this.edt\_Font.value;

trace("Font :: " + szfont);

var bReturn = this.objExtCommon.\_ext\_getUseFont(szfont);

if( bReturn )

{

alert( "^.^ Font 가 있습니다. : " + bReturn );

}

else

{

alert( " >.< Font 가 없습니다. >.< : " + bReturn );

}

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **szFont** | **STRING** | 찾고자 하는 폰트 이름. |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| BOOL | * TRUE : 설치 되어 있습니다. * FALSE : 설치 되어 있지 않습니다. |

## **Method Name : \_ext\_getComPortList()**

▶ **Description**

현재 PC 에 연결 된 사용 가능한 COMPORT를 취득 합니다.

**참고**:

* + - * 연결되고, 사용 중( connect) 인 COMPORT 는 취득 하지 않습니다.
      * Connect 되지 않고, 사용 가능한 comport 만 취득 합니다.

▶ **Syntax**

var arrayCOMPORT;

arrayCOMPORT = this.objExtCommon.\_ext\_getComPortList();

if( arrayCOMPORT )

{

var nCnt = arrayCOMPORT.length;

if( nCnt != 0 )

{

for( var index=0; index < nCnt; index++)

{

trace("사용 가능한 COMPORT : " + arrayCOMPORT[index] );

}

}

else

{

alert("COMPORT 가 없습니다.");

}

}

else

{

alert("COMPORT 가 없습니다.");

}

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
|  |  | **없음** |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| ARRAY | * TYPE : **STRING** * 연결되어 있는 COMPORT를 array 로 리턴 합니다. |

## **Method Name : \_ext\_getCurrentProcessCPU()**

▶ **Description**

호출 한 프로세스의 CPU 사용량을 리턴 합니다. [ % ] .

**참고**:

* + - * **너무 자주 호출되면 측정 자체가 결과에 큰 영향 을 줍니다**
      * **작업 관리자의 CPU 사용률가 차이가 있습니다.** 
        + **이는 호출 시점 과. CPU 사용률 계산 시간과의 딜레이 차이로 인함.**

▶ **Syntax**

**var nCpu = this.objExtCommon.\_ext\_getCurrentProcessCPU()**

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
|  |  | **없음** |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| integer | * Integer % 로 리턴 합니다. (소수점 2째 자리 까지. 표현.) |

## **Method Name : \_ext\_getProcessModule(nProcessID)**

▶ **Description**

프로세스 아이디를 이용하여 해당 프로세스의 Module 들을 array로 취득 합니다.

**참고**:

* + - * + **프로세스 보안 솔루션(안랩, nProtect 등…)으로 인해 모듈정보들을 취득할 수 없을 수 있습니다.**

▶ **Syntax**

var nProcessID = this.objExtCommon.\_ext\_getCurrentProcessID();

var arrayModule = this.objExtCommon**.\_ext\_getProcessModule(nProcess**ID);

var nCnt = arrayModule.length;

if( nCnt != 0 )

{

for( var index=0; index < nCnt; index++)

{

var tmpModule = arrayModule[index];

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

var fileName = path.pop(); // 마지막 array 데이터 취득

trace(“Module Name : ” + fileName);

}

}

else

{

alert("Fail Module List.");

}

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **nProcessID** | **integer** | **찾고자 하는 프로세스 아이디를 입력 합니다.** |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| Array | * Type : String   + Process 에서 로드된 Module path |

**Method Name : \_ext\_getIPAdapterInfo**

▶ **Description**

* + 현재 피시의 어뎁터의 정보를 취득합니다.
  + 어뎁터의 정보를 2차원 배열로 취득합니다.

▶ **Syntax**

//배열로 리턴 합니다.

var ipArray = this.objExtCommon.\_ext\_getIPAdapterInfo();

trace( "Adp info : " + ipArray);

var Adplen = ipArray.length;

var index=0;

for(index = 0; index < Adplen; index++)

{

var detailinfo = ipArray[index];

trace( "설명 : " + detailinfo[0]);

trace( "이름 : " + detailinfo[1]);

trace( "포멧 : " + detailinfo[2]);

trace( "아이피 : " + detailinfo[3]);

trace( "마스크 : " + detailinfo[4]);

trace( "게이트웨이 : " + detailinfo[5]);

trace( "MAC : " + detailinfo[6]);

trace( "IPV6 : " + detailinfo[7]);

}

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Arg** | **Type** | **Description** |
|  |  | 없음 |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| array | [0][0] : 설명  [0][1] : 이름  [0][2] : 포멧  [0][3] : 아이피  [0][4] : 마스크  [0][5] : 게이트웨이  [0][6] : mac  [0][7] : ipv6 (Windows XP 에서는 IPV6 값을 취득 할 수 없습니다.  출력 되는 값은 “Windows XP Not Support” ) |

## **Method Name : \_ext\_getPrimaryHDDSerial(nNumberDisk)**

▶ **Description**

물리 하드 디스크 정보 취득 하는 함수.

▶ **Syntax**

**var nNumberDisk = 0;**

**var szSerial = this.objExtCommon.\_ext\_getPrimaryHDDSerial(nNumberDisk);**

**trace("==================================================");**

**trace( "Seria Number : " + szSerial[0] );**

**trace( "Product ID : " + szSerial[1] );**

**trace( "Device's product revision : " + szSerial[2] );**

**trace( "Vender ID : " + szSerial[3] );**

**trace( "Bus Type : " + szSerial[4] );**

**if( szSerial[5] == true )**

**{**

**trace( "장치 제거 가능." + szSerial[5] );**

**}**

**else**

**{**

**trace( "장치 제거 불가능." + szSerial[5] );**

**}**

**trace("==================================================");**

▶ **Parameter**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Description** |
| **nNumberDisk** | **integer** | **물리 하드 디스크 장비 번호.** |

▶ **Return.**

|  |  |
| --- | --- |
| **Type** | **Description** |
| **array** | [0] : String : 하드디스크 시리얼 번호.  [1] : String : 제품 아이디.  [2] : String : 제품 버전 정보.  [3] : String : 벤더사 아이디.  [4] : String : 연견 방식. [SCSI, ATAPI, ATA, IEEE1394, BusTypeSsa, BusTypeFibre, USB, RAID, iSCSI, SerialATA, SD, MMC, Not Found ] 중 하나.  [5] : bool : 제거 가능 여부 : ( true : 가능 ) false : 불가. |