

DBFree functions

Table of Contents

DBFree functions.....	1
Arrays.....	10
aGrow(aArray)	10
aSize(MyArray)	10
aadd(aArray, xVal)	10
adimens(aaa)	11
amSize(MyArray)	11
amfind(aArr,cKey)	11
amprint(aArr, nMode)	11
aprint(aaa, cSep)	11
areverse(aaa)	11
aseek(aArr,cKey,cDelim)	11
atos(aArray,cSep)	11
matrix(aName, nRows, nCols)	11
newarray(cName)	11
square2str(aArr, cSep, cDiv)	11
str2arr(cStr, cSep, aaa)	11
vector2str(aArr, cSep)	12
StrToArr(cStr, cSep, aaa)	12
StrToNumArr(cStr, cSep, aaa)	12
Bootstrap's.....	12
CalendarMonth(dToday, cTargetPage, nMode, cMainTarget)	12
alarm(cTitle,cMsg)	12
alert(cTitle,cMsg,nVal,cStyle,nShowNow)	12
alertBox(nOpt)	12
badge(cStr, cPos, cStyle, cId)	13
bottomline(cStr,cColor,cBgColor)	13
box(cMsg, nVal, cType)	13
btn2str(nType)	13
btnHide(cLabel, nType, cDomID, cStyle, cImage, cPull, nExcl)	13
btnPopUp(cLink, cLabel, nType, cTarget, cStyle, cImage, cPull, nPopType, cDomId. 13	
buildMenu(nVal)	14
buildForm()	14
button(cLink, cLabel, nType, cTarget, cStyle, cImage, cPull, cDomId, cToolTip,.....	14
calendar(dDateOrNum,cLangId,nYear)	14
caption(cTitle,cSubtitle)	14
carousel(cFolder,cMask,nImgHeight)	14
cell(cStr,nOutput)	14
checkbox(cName,cDescr,cVal,nVal,cId)	15
confirm(cTarget,cTitle,cMsgYes,cMsgNo,cStyle,nTyp)	15
container(nOpt,cStyle)	15
content(cMargin,cColor)	15
cpanel(cText1,cText2,cText3,cStyle1,cStyle2,cStyle3,cStyle)	15

curDisk()	15
divider()	15
drawer(nVal,cColSize,cId,cStyle)	15
endRadio()	15
endRow()	16
formStart(cURL, cTarget, cId, cXY, cVA, cURLparms, cComment)	16
formEnd(cComment)	16
formTitle(cBrand,cLogo)	16
formSubmit(cSubmit,cReset,cId,cButtAttr)	16
formField(cName, cLabel, cValue, cType, nFocus, cClass, cStyle, cJcode)	16
framebox(cBorderColor, cInnercolor, cFontcolor, nOutput)	16
fullscreen(nMode)	16
getDate(cName,cLabel,nFocus,dDate,cMsg,cId)	16
getEmail(cName,cMsg,cId)	16
getLogic(cTarget,cTitle,cMsgYes,cMsgNo,cStyle,nTyp)	16
getNum(cName,cLabel,nFocus,nValue,cMsg,cId,nMin,nMax)	17
getOption(cName,cOptions,cLabel,cId)	17
getOptions(cName,cOptions,cLabel,cId)	17
getStr(cName,cLabel,nFocus,cText,cMsg,cId,cType)	17
getText(cName,cLabel,cContent,cMsg,cId)	17
hdiv(nOnOff,cDomId,cClass,cStyle)	17
jLinkModal(cDiv, cLink, cStr, cType, cClass)	17
jencode(cTxt)	17
jumbo(xVal,cDomId)	17
jurl(cTxt)	17
listApps(nOpt,cFolder,nMinAppLev)	17
menuBar(nVal,cStyle)	18
menuTitle(cBrand,cLogo,cLink)	18
menu(nVal,nPos, cStyle)	18
menuItem(cLink,cDesc,cTarget,cStyle,cDomId)	18
menuDivider(cStr)	18
menuadd(nVal,cStyle)	18
off()	18
panel(nVal,cStyle)	18
panelbody(cText,cStyle)	18
panelheader(cText,cStyle)	18
panelfooter(cText,cStyle)	18
popMsg(cStr, cLabel, cClass)	19
poster(xVal,nHeight,cStyle,nMode)	19
printh(cStr,nSize,cColor)	19
pullR(xVal,nMode)	19
radioButton(cName,cDescr,cVal,nVal,cId)	19
rowEnd()	19
rowStart()	19
screen(nMode)	19
shelf(nVal,cStyle)	19
sourcebox(cSourceCode)	19
startRadio(cName,cLabel,nVal)	19

startRow()	20
submenu(cStr,cStyle)	20
well(nVal,cStyle,cContent, cId)	20
Core functions.....	20
ajRun(cUrl, cParam, cTargId)	20
ajaxCall(cUrl)	20
ajr(cUrl, cParam, cTarget).....	20
default(cVarName, anyValue)	20
desktop()	20
jclear(cVarName,cType)	20
jread(cVarName,cType)	20
jwrite(cVarName,cValue,cType)	21
jzap(cType)	21
libVer(verbose)	21
web()	21
websettings()	21
Calendar dates.....	21
CtimeDiff(cTime1,cTime2)	21
chrDate(dDate)	21
curDate()	21
curYear()	21
dayName(nnn,cLang)	21
dayOfW(dDate, cLang, lShort)	21
dayStr(dDate, cLang, nLen)	21
daysFrom(date1,date2)	22
daysOfM(date1)	22
daysOfMonth(nMonth)	22
dtoh5(dDate)	22
dtok(dValue)	22
evalTstamp(cTimeStamp,cParm)	22
firstDoM(NumOrDate)	22
firstDoY(nYear)	22
fulldate(dThis, cCountryID, lShort, lAddtime)	22
initDates()	22
isLeapYear(DateOrNum)	22
isLeap(dDate)	22
itDate(dDate, lFullname)	23
itDateStr(dDate)	23
itDow(ddd, lFullname)	23
itMonth(dDate, lFullname)	23
ktod(cStr,cCountry)	23
lastDoM(NumOrDate)	23
lastDayOfM(NumOrDate)	23
lastDoY(nYear)	23
monthName(nnn)	23
nowstr(cStr)	23
optDay(nDay)	23
optMonthIt(dDate)	23

optMonth(nMonth)	23
optYear(nYear, nDig)	24
secs2date(nSecs)	24
secs2days(nn1,nn2)	24
timestamp(cParam)	24
timeDiff(cStr1,cStr2)	24
today(cLanguage)	24
wcdow(dThis, cCountryID)	24
wcmonth(dThis, cCountryID)	24
wdow(dThis)	24
week(dThis)	24
Database functions.....	24
CloneRec(nRec)	24
NoAjax()	24
appendBlank(cKeyFld,mKeyVal)	25
dbInfo()	25
dbgrid(cFldList,nRecToMove,nMode,cLegend,cLink,nType,cInputStyle,cTableStyle,cRow).....	25
dbskip(nQuant)	25
displayStru(cAlias,nOption)	25
editpage(cFile,cMsg)	25
enableOdbc()	25
enableClipper()	25
enableFox()	25
enableDbase()	25
fieldtype(fldNameorNum)	25
fieldlen(cField)	26
fieldval(nFld)	26
fieldcount()	26
fieldfitter(cFldname, xCargo)	26
fieldfit(cFldname, xCargo)	26
fieldNum(cStr)	26
fieldList(xVal)	26
fieldsize(fldNameorNum)	26
filterkey(cAlias)	26
fit(nField, xVal)	26
fsize(nField)	26
ftype(fieldNum)	26
go2apps(cFile,cMSg)	26
here()	26
isDBD()	26
isMemory(arg_x)	27
listNext(nRecs,cFldList,cLink1,cParm,cWebvar,cLink2,nLegend)	27
pageCount(nRecs, nPgRows)	27
setDb(cDbPos, xRelative)	27
setDb3(cName, lShowOnly)	27
setLocalDb(cDbPos, lNoCreate)	27

setdbtype(cType)	27
showMemory(cFile)	27
Extended functions.....	27
appInfo(any)	27
casual()	28
convert(anyval, cType)	28
getAppVer()	28
libInfo(cLibPath,lVerbose)	28
modInfo(any)	28
random()	28
Files and folders.....	28
FileNoDrive(cPathName)	28
FileBase(cFullFileName, noSlash)	28
FileName(cFullFileName, nOption)	28
FilePath(cFullFileName, lNoDrive)	28
breadcrumb(cFolder,cTarget,cClass,cStyle)	28
createDir(cPath)	28
curDosDir(nRemoveDriveLetter,nRemoveSlash)	29
curDosPage()	29
curDrive()	29
curWebDir(full)	29
curWebPage()	29
dir(cStartPath, cLink)	29
dirName(cPath)	29
directory(cFolder,cMask,nTarget,nMode,cIcon)	29
documentRoot()	29
dos2web(cFile)	29
fcountByTime(cFold, cMask, cTime, cSign, nFlag)	29
fileScan(cFile,nRowsize,cEol)	30
fileDrive(cPathName)	30
fileclean(cFile)	30
filefolder(cFullFileName)	30
fileBone(cFile)	30
fileExt(cFilename, xExcludeDot)	30
fileSource(cFile,nLineLen)	30
file2str(cFile)	30
fileread(cFile,nMode,nBuffSize)	30
filewrite(cFile,cText,nMode)	30
getBytes(cFile,nStart,nBytes)	30
include(cFil, cSed, cMarker)	30
isDir(cFolder)	31
isFolder(cFolder)	31
jlink(cFileAndPath)	31
label(cFilename)	31
listFunc(cFile)	31
listFiles(cThisFolder,cMask,nOpt,cCargo1,cCargo2)	31
listFolders(cThisDir,xFlag)	32
makeTree(cFinalPath)	32

pageName()	32
pageFile(nMode)	32
plainPath(cPath)	32
preload(cFile, cSed, cMarker, cTagId, cString)	32
readfile(cFile,nMode,nBuffSize)	32
showpath(cFolder,nOpt,cLL,cRR,cSty)	32
storage(any)	32
subDir(cTxt)	33
sysDrive()	33
upDir(cTxt)	33
wdrive(cFilename)	33
web2dos(cFile)	33
web2java(cTxt)	33
webRoot(nStyle)	33
writefile(cFile,cText,nMode)	33
HTML related functions.....	33
CloneRecord(nRec)	33
HTML5(nMode)	33
alertOnce(cMsg,cCookieName,cStyle)	33
blanks(nChars)	34
breakline	34
brk()	34
buildComment()	34
centred(nVal)	34
clock(xPrint)	34
closeWebPage()	34
closePage()	34
codebox(cStr,cId)	34
commentln(cStr)	34
comment(cStr)	34
cvttext(cStr)	34
displayRecord(nRec, nStyle)	34
display(cFldList, nStyle)	35
editRecord(nRec, cFieldList, cUrl, nMode, nDecs, cJaction)	35
flds2form(cFldList)	35
hyperlink(cUrl, cDescr, cTarget, cStyle, cClass)	36
image(cUrl,nHeight)	36
inputRadio(cName, cValue, varToMatch)	36
input_c(cName,cId,cValue,cLabel,cPlaceholder,cHelp)	36
input_m(cName,cId,cValue,cLabel,cStyle,cHelp)	36
item(xVal,nType)	36
jhide(cDiv)	36
js(nVal)	36
jumptop(cMsg,cPos)	36
junhide(cDiv)	37
linkit(cUrl,cLabel,cTarg)	37
link2menu(cDir, cChoiches, xDebug)	37
link2app(cDir, xDebug)	37

load(cPageUrl, cHeader, cFooter)	37
navButtons(cParms,nIndexes,xOutput)	37
navBrowse(cFldList, cEditPage, cParmList, xOutput)	37
navEdit(cUrl, cFldList, nModal, xOutput)	37
openWebPage(nOpt)	38
openPage(nOpt)	38
optByDb(cTab,cFld)	38
pageReload(nOpt)	38
pl(cTxt,nSize)	38
printIni()	38
println(xVal,cSize)	38
rec2form(nFldStart,nFldEnd)	38
ruler(nDist,cColor)	38
save2Record(nRec, aWebVarList, cAlias)	39
saveRecord(nRec, aWebVarList, lDebug).....	39
setStyle(cStyle)	39
shim(nPixels,cDivTxt)	39
showMaster(aOfFlds)	39
showSlave(aOfFlds)	39
strSafe(strText)	40
text(xVal,cSize)	40
url2form(cTxt)	40
webvar(cName,cValue)	40
Login helpers.....	40
clearSess(cThisSeed)	40
displayJvars(cJvarStr)	40
displaySession(cThisSeed, cStyle)	40
displayGlobals(cFile,cStyle)	40
isLocal(cIpAddr)	40
loginButton()	41
logoutButton()	41
logwrite(cMsg, cUsrName, cCodOp, cCargo, nTTime, cClass)	41
setId()	41
Memory functions.....	41
WebVars2url(aArr, cExclude)	41
WebVars2Fields(aArr, cExclude)	41
WebVars()	41
cfg2mem(cIniFile)	41
file2vars(cTextFromJS,lVerbose)	41
getCargoFile(cSeed)	41
getMemFile(cSeedStr, vEncrypt, vUsePort)	41
getMemDir()	41
getMemPath()	42
getSessFile(cSeed)	42
getSession(cSeed)	42
getseed(nVal)	42
ini2Mem(cIniFile)	42
ini2vars(cFilename)	42

listMem(memArray)	42
listVars(cStr,cSep)	42
loadini(cFile)	42
loadIniVars(cFile,nRowsize,cEol)	42
makeSeed(nVal)	42
restmemory(ddd)	42
savememory(ddd,ppp)	42
saveSession(cSeedStr)	43
undefined(cVarName)	43
undef(cVarName)	43
varBar(cSeed,cAction)	43
webVars2str(aaa)	43
webv2f(aArr, cExclude)	43
Numerics.....	43
hex(nnn)	43
hextoken(nnn)	43
keepVal(cStr, nLow, nHigh)	43
range(nVar,nMin,nMax)	43
setDec(nnn)	43
Session handling.....	43
getGlobFile()	43
getJseed(cVarName, nAction)	44
initSession(cSeed, nOption)	44
initGlobals(norewrite)	44
initXvars()	44
initJvars()	44
initJseed(cSeed)	44
initJ(cNewSeed,cTarget)	44
readJvar(cVar,cOpt)	44
restoreSession(cSeed,nOption)	44
saveJseed(cStr,cVar)	44
showSession(nOpt)	44
showGlobals(nOpt)	45
showJvars(nOpt)	45
showCargo(nOpt)	45
showZvars(nOpt)	45
Strings.....	45
Capital(cStr)	45
StripChars(cTxt , nUpperASCII)	45
WordRight(cStr, cSep, lOffset)	45
WordLeft(cStr, cSep)	45
block(nParm)	45
break()	45
c2l(xVal)	45
c2n(anyVal,nMode)	45
cc(anyVal)	45
cleanUp(cVar)	46
cleanStr(cVar)	46

code2memo(cTxt)	46
contains(cTarget, cString, nIgnorecase)	46
cutAt(cString, cKey)	46
dblSlash(cTxt)	46
decval(cStr,cSep)	46
dslash(cTxt)	46
fChar(val)	46
ffix(cStr)	46
fixSlash(cTxt)	46
forEach(cStr,cSep,cFunc)	47
htmlMemo(cTxt)	47
ini2Str(cFile)	47
iniToStr(cFile)	47
iniVar(cFile,cVarname)	47
intval(cStr)	47
jInnerHtml(cDiv,anyVal)	47
jInsertSeed(cDiv,cThisSeed)	47
jaddP(cDiv,cElem,cStr)	47
jchange(cDiv,anyVal)	47
jfieldval(cId,xVal,cType)	47
jslash(cTxt)	48
jttitle(cStr)	48
junslash(cTxt)	48
jvalue(cDiv,cElem,anyVal)	48
leftMost(cString, cKey)	48
letterOnly(cStr)	48
memo2code(cTxt)	48
memoClean(strText)	48
memoHtml(cTxt)	48
n2c(xx,nn)	48
neutralize(cTxt)	48
nice(cFile,nOpt)	48
nm(xVal)	49
noSpaces(cStr)	49
occur(cString,cTarget)	49
onlyNum(cStr,nMode)	49
p(cStr)	49
padc(cString, nLength, cCaratt)	49
padl(cString, nLength, cCaratt)	49
padr(cString, nLength, cChar)	49
parseFor(cStr, cVarname, cOpenTag, cCloseTag, cTokenTag)	49
parseStr(cStr, cOpenTag, cCloseTag, cTokenTag)	49
print(cStr)	50
println(cStr, nParm)	50
readIniVar(cFile,cVarname,cDtype)	50
rec(nDigit)	50
revStr(cStr)	50
reverse(cStr)	50

rightMost(cString, cKey)	50
rslash(cTxt)	50
splitVal(cStr,nVal,cSep)	50
ssoff(cStr)	51
strcount(cTxt,cKey)	51
trimRight(cStr, nPos)	51
trimLeft(cStr, nPos)	51
trimLeftAt(cString, cKey)	51
trimRightAt(cString, cKey)	51
writeIniVar(cFile1,cVar1,cValue1)	51
wstr(xVal)	51
zz(anyVal,nDecim)	51
Web functions.....	51
DnsList(cFile)	51
callingpage(nStyle)	51
computername()	52
ip(nParm,cIp)	52
ip2num(cStr)	52
isMob(cUserAgentStr)	52
isWeb()	52
jreload(cLink)	52
jsetId(cElement, cProperty, cValue)	52
listWebVars(aArr, nStyle)	53
num2ip(n)	53
passVars(aArr)	53
passFields(aArr)	53
protocol(cUrl)	53
scriptName()	53
serveraddr()	53
servername()	53
serverport()	53

Arrays

aGrow(aArray)

Grows an existing array to the desired number of elements returns the number of new element added

aSize(MyArray)

-- return the dimension of a matrix (a 2D array) as string

aadd(aArray, xVal)

Grows an existing array of one element (add at the end)

adimens(aaa)

*** Undocumented (waiting submission)

amSize(MyArray)

*** Undocumented (waiting submission)

amfind(aArr,cKey)

Array multi finder Accepts a single-dimension array and returns a string with all elements matching the given key. The returned string contains elements position separated by square bracket: Example:

ainit("test","big","cal","bug","bad","try","left","light","can") ? ccc := amfind(test,"b") Result: [1][3][4]

amprint(aArr, nMode)

-- prints out the content of a 2D array (matrix) If used with a single dimension returns nothing. second parameter is numeric: if passed determines the type of putput none or 0=elements seprated by blanks,1=separated by a paragraph sign,2=using a format compatible with parsestr(),3=a format for ini file This function can be used to store matrixes directly into an ini.file (3) or into a table memo field (2) Feb 13,2016

aprint(aaa, cSep)

Prints out the content of an array (vector) separated by a sseparator The second parameter defines the char to use as separator (default is :). Example:

ainit("test","fiat","bmw","opel","seat","dacia","renault","citroen","audi") ? aprint(test) Result:
::fiat::bmw::opel::seat::dacia::renault::citroen::audi:

areverse(aaa)

Reverse the order of the elements of an array WARNING: The original array gets modified irreparably! Also returns a string with the array elements reordered

aseek(aArr,cKey,cDelim)

*** Undocumented (waiting submission)

atos(aArray,cSep)

-- array to string (IMPROVED NOV 2014))

matrix(aName, nRows, nCols)

-- creates a rectangular (bi-dimensional) 2D matrix filled with strings that represent the cell address in form row-column

newarray(cName)

*** Undocumented (waiting submission)

square2str(aArr, cSep, cDiv)

-- trasforma un array (matrice quadra) in stringa parametro: tipo di separatore

str2arr(cStr, cSep, aaa)

wrapper for StrToArr()

vector2str(aArr, cSep)

-- accapts a single-dimension array (a vector) and returns a string second (optiona) parameter is the char to

DBFree Library Functions – free.lib 2017

use as separator

StrToArr(cStr, cSep, aaa)

Example: cString := "mia&martini&grande&cantante" alnit("myArray","") strToArr(cString,"&",myArray) accepts a string with words separated by a separator and returns an array with as many elements as where the words

StrToNumArr(cStr, cSep, aaa)

Converts a string to a numeric array: Usage: cString := "100&12&5&24" alnit("myArray","") strToNumArr(cString,"&",myArray)

Bootstrap

CalendarMonth(dToday, cTargetPage, nMode, cMainTarget)

This procedure accepts a calendar date and display a grid of buttons each one representing a day of the month. Parameters are:

- p1 - a calendar date around which to build the full month display
- p2 - the target page to call when clicking on day (this page will receive some of the parameters taken from current values of this page)
- p3 - a number representing the mode of displaying the target page: 0 means full page, while 1 cause the page to display into a modal window
- p4 - A target page to call while clicking on large button on top (usually a report showing the full month or an equivalent page)

Example (opens the month into a modal window):

```
CalendarMonth(dStart,"showDay.msp",1,"listAll.msp")
```

The button on top has a default text that can be changed modifyng pb_calend_title
jchange("pb_calend_title","Click for a report") Nov 29,2015

alarm(cTitle,cMsg)

-- opens a dismissable alert box from inside a chunk (to be used for alarm messages in the headers and the like)

alert(cTitle,cMsg,nVal,cStyle,nShowNow)

builds a dismissable box with a message Parameters: Title, message, a number for the color; 1=warning, 2=info, 3=success, 4=danger 0=expects a 4th parameter with a string for the style to use defaults to 1 if cStyle is passed will be used instead of defaults WARNING: if is to be passed to javascript just use box() while if used in retro-style programming use the wrapper alarm() Dec 18,2015

alertBox(nOpt)

-- opens and closes an alert box nOpt= empty or 1 -- opens, 0 -- closes Nov 22,2015

badge(cStr, cPos, cStyle, cId)

-- generating a badge p1 -- text to show inside the badge p2 -- position of the badge, "L"=Left, "R" or 1=right, empty=none p3 -- optional style string p4 -- optional DOM ID (name will be identical but in

DBFree Library Functions – free.lib 2017

uppercase)) Dec 15,2015 - Aug 27,2015 - Sep 7,2016

bottomline(cStr,cColor,cBgColor)

Returns a fixed DIV for a bottom line Parameters are: Text to show, text color, background Dec 1,2015

box(cMsg, nVal, cType)

Similar to alert but opens and closes the box Parameters: cMsg -- if empty just opens the box (then pass box(0) to close it) nVal -- none or 1 = open the DIV, accepts an optional text to display 0 = closes the DIV 2 = info, 3=success, 4=warning, 5=danger Third parameter: if passed causes to return a string suitable to be passed to javascript (no CR_LF) Passing a TEXT as first parameters returns a white box with text

btn2str(nType)

-- reserved for internal use

btnHide(cLabel, nType, cDomID, cStyle, cImage, cPull, nExcl)

Creates a button and a hidden DIV:

- p1 -- Text to show in the button body -- passing a zero (N) closes the div
- p2 -- type of button from 1 to 20 (default 2)
- 1-5=small, 5-10=medium, 10-15=large, 15-20=block add 1=primary, 2=success,3=info,4=warning,5=danger over 20=btn-default then must provide p5 with style string p3 -- Dom ID (and DOM name converted in uppercase) of the element
- p4 -- style string
- p5 -- image to show in the button body
- p6 -- position: R or L
- p7 -- mutual exclusive -- 1=using ToggleOne() - else or none=using toggleDiv()

Usage:

```
p( btnHide("Show or Hide",3)+"This is hidden"+btnHide(0) )  
p( btnHide("Show or Hide",1,"myId") )
```

btnPopUp(cLink, cLabel, nType, cTarget, cStyle, cImage, cPull, nPopType, cDomId)

Creates a button for a PopUp window:

- p1 -- URL
- p2 -- Text to show in the button body
- p3 -- type of button from 1 to 20 (default 2) 1-5=small, 5-10=medium, 10-15=large, 15-20=block add 1=primary, 2=success,3=info,4=warning,5=danger over 20=btn-default then must provide p5 with style string p4 -- target frame (defaults to _self)
- p5 -- style string p6 -- image to show in the button body
- p7 -- position: R or L
- p8 -- pop type 0 to 4
- p9 -- DOM identifier ID for the button (identifier NAME is same but uppercase))

buildMenu(nVal)

Display an example of usage Call this function, expose the HTML code in browser then copy and paste in your code Parameters: if passed shows an example in *classic mode* default is *standard mode*

buildForm()

Display an example of usage for the form Call this function, expose the HTML code in browser then copy and paste in your code

button(cLink, cLabel, nType, cTarget, cStyle, cImage, cPull, cDomId, cToolTip,

Creates a button:

- p1 -- URL
- p2 -- Text to show in the button body
- p3 -- type of button from 1 to 20 (default 2)
 - 0=xsmall, 1-5=small, 5-10=medium, 10-15=large, 15-20=block add 1=primary, 2=success,3=info,4=warning,5=danger over 20=btn-default then must provide p5 with style string
- p4 -- target frame (defaults to _self)
- p5 -- style string p6 -- image to show in the button body
- p7 -- position: R or L
- p8 -- DOM identifier (name is the same but uppercase)
- p9 -- title for tooltip
- p10 - suppress printout and return all in a string

calendar(dDateOrNum,cLangId,nYear)

Show a calendar of a month, read only. Retrurns a string with all the HTML code.

Accepts a number (for the month) or a date: if missing defaults to current month. Optionally accepts the two-letter string for the language Optionally you can specify the year (overrides dDateOrNum)

caption(cTitle,cSubtitle)

Outputs the HTML tags for a caption and a subtitle, according to current CSS.

carousel(cFolder,cMask,nImgHeight)

Builds a fully working carousel, taking images from given folder (default is \images\show) - if not exists will be created

cell(cStr,nOutput)

-- accepts a value a return a box in non-editable input style

- p1 - X -- the content to insert in the entryfield (will be converted into text)
- p2 - X -- if passed the function won't output but etrun the HTML code

To be used for building grids from tables.

checkbox(cName,cDescr,cVal,nVal,cId)

create a checkbox parameters:

- p1 -- name to assign
- p2 -- description (shown on right of button)
- p3 -- the value to return if checked - if omitted returns the string N/A
- p4 -- 1 or 0 (checked or unchecked)
- p5 -- DOM identifier to apply (must be unique for each button)

confirm(cTarget,cTitle,cMsgYes,cMsgNo,cStyle,nTyp)

-- button for confirmation Mar 1,2017

container(nOpt,cStyle)

Makes the container; no parameters or nOpt=1 opens a standard container, with nOpt=2 the container is fluid nOpt=0 closes the container. Nov 5,2015

content(cMargin,cColor)

-- Start or close a DIV with cMargin all over and text of given color cMargin is the string for the margin (using px/em) if 0 (number) is passed closes the DIV (returning the tag) cColor -- is the string with name or number for the text color Nov 30,2015

cpanel(cText1,cText2,cText3,cStyle1,cStyle2,cStyle3,cStyle)

*** Undocumented (waiting submission)

curDisk()

returns the current disk drive (as a string of 2 chars)

divider()

return the tag for a horizontal rule Nov 4,2015

drawer(nVal,cColSize,cId,cStyle)

Starts or ends a Bootstrap COL p1 -- no parameters = starts the DIV, passing 0 = ends the DIV ANY other number up to 12 defines the WIDTH of the drawer (default is 12 that means 12/12 = a full row is taken) p2 - two chars string for column: xs/sm/md/lg -- default is md p3 - DOM identifier to assign (default is none) can be any value (will be converted to string)-useful for cycling p4 - is a string for the style Dec 2014, Sep 10,2016

endRadio()

closes a radio button group

endRow()

-- closes a DIV row (i.e. returns </div>)

formStart(cURL, cTarget, cId, cXY, cVA, cURLparms, cComment)

starts the webform - parameters: p1 -C- the string with the URL where to submit the form (default is #) p2 -C- target frame for the form (default is _self p3 -C- the DOM id and name to assign to form(default=automatic)) p4 -C- the seed value to write in field XY (default none)) - if it is passed an asterisk then the two fields are omitted from form p5 -C- the action code to write in field VA (default none)) p6 -C- a string with URL parameters to be passed along the form p7 -C- comment to insert (visible only in the HTML source)

formEnd(cComment)

ends (close) the webform

formTitle(cBrand,cLogo)

gives a title and image to form legend

formSubmit(cSubmit,cReset,cId,cButtAttr)

-- create an input field for email cSubmit = text for submit button cReset = text for abort button - if a zero (char) is passed then is not displayed cId = the DOM id and name to assign to the form cButtAttr = string: xs,sm,md,lg

formField(cName, cLabel, cValue, cType, nFocus, cClass, cStyle, cJcode)

-- returns a form field 1 cName = name of the entryfield 2 cLabel = a string with the text to display before field 3 cValue = default value to assign 4 cType = T/H/S/P/D/E/N/R/K/O/U/W/C (HTML5) 5 nFocus = none/0/1 (1=get focus) 6 cClass = CSS Class 7 cStyle = string for the style 8 cJcode = string for the javascript code (method and values) Legend for types: T=text H=hidden S=submit P=password D=date E=email N=number R=range K=search O=time U=url W=week C=color

framebox(cBorderColor, cInnercolor, cFontcolor, nOutput)

--- similar to well, but accepts directly style colors p1 - C -- border color string p2 - C -- background color string p3 - C -- font color string p4 - N -- if 0 closes the box Feb 24, 2017

fullscreen(nMode)

-- starts a container full screen passing zero (or any other parameter) will close the container Feb 8,2017

getDate(cName,cLabel,nFocus,dDate,cMsg,cId)

Mar 1,2017

getEmail(cName,cMsg,cId)

create an input field for email cId = the DOM id and name to assign to form

getLogic(cTarget,cTitle,cMsgYes,cMsgNo,cStyle,nTyp)

-- FUNZIONE SPERIMENTALE - per campi logici Apr 14,2017

getNum(cName,cLabel,nFocus,nValue,cMsg,cId,nMin,nMax)

p1 - name for the entryfield p2 - text to show aside the field p3 - focus -- 1 to get focus0 p4 - initial value p5 - tooltip message p6 - DOM id p7 - range value - min p8 - range value - max Sep 10,2016

getOption(cName,cOptions,cLabel,cId)

create a select group field with single choice cId = the DOM id and name to assign to select

getOptions(cName,cOptions,cLabel,cId)

create a select group field with multiple choice cId = the DOM id and name to assign to select

getStr(cName,cLabel,nFocus,cText,cMsg,cId,cType)

Places a pseudo@get in a web form for getting a string Notice that the FORM must be declared before (and closed after) 1 cName -- name to assign to entryfield (i.e. EF_MYFIELD) 2 cLabel -- text to show aside the field (if any) 3 nFocus -- none, 0 or 1. If 1 then this field will get focus 4 cText -- the value to assign to entryfield (text up to 128 chars) 5 cMsg -- an optional help message to display below the field 6 cID -- an optional DOM ID to assign to entryfield for furthrt manipulation 7 cType -- the HTML5 type for the field, overriding default one (i.e. password,mail,ecc)

getText(cName,cLabel,cContent,cMsg,cId)

generate a web form text area Parameters: cName = name of the entryfield cContent = text to insert in the textarea cLabel = Text to show above the textarea cMsg = TExt to display below the area cId = the DOM id and name to assign to form

hdiv(nOnOff,cDomId,cClass,cStyle)

-- hidden DIV shortcut: p1 -- none or 1 open the hidden div,0 close it. If a string then is taken as DOM id p2 -- DOM id identifier for the DIV (name is forced to uppercase) default 'hdiv' p3 -- class for the closeAll javascript function (default is hdiv) p4 -- additional style instructions (except display) Aug 8,2016

jLinkModal(cDiv, cLink, cStr, cType, cClass)

inserts a button for callin a modal cStr (opt) is the text in the button cType (opt) is the style 0-1-2-3-4 cClass (opt) to specify a custom class

jencode(cTxt)

accepts a string and returns a string suitable to be passed as parameters to javascript functions (double the slashes and so on)

jumbo(xVal,cDomId)

Returns the tags for a jumbotron large box p1 -- None or (N) 1 opens the jumbo, 0 closes if a string is passed it is inserted in the jumbo box then closed p2 -- DOM id to use Aug 16,2016

jurl(cTxt)

accepts an URL in DOS/WEB format and return a string encoded in the HTML format

listApps(nOpt,cFolder,nMinAppLev)

-- returns a list of applications found in the given folder p1 - N -- 1=buttons 2=panels 3=table 4=list 5=list with CRLF (default 1) p2 - C -- string with pah for the starting folder (default is /APPS) p3 - N -- number for minimum security level user must have to see this app appear in listings (default is 12) Jul 2015 -- Feb 27,2016 -- Mar 06,2017

menuBar(nVal,cStyle)

MENU SYSTEM - step 1 No parameters = start the menu bar structure 0 = close the structure 1 = bar standard fixed on top 2 = bar inverse fixed on top 3 = bar inverse at the bottom 4 = bar standard at the bottom second parameter (optional) is a string for style

menuTitle(cBrand,cLogo,cLink)

MENU SYSTEM - step 2 cBrand = a string to use as title cLogo = the filename and path of the image to display before the title (optional)

menu(nVal,nPos, cStyle)

MENU SYSTEM - step 3 p1 - empty == starts the collapsible section of the menu passing 0 = ends the collapsible section p2 - if passed the menu is placed at bottom of page p3 - optional style string Notice: the menu items (drop downs or not) must be placed inside this section Dec 2013 -- Sep 7,2016

menuItem(cLink,cDesc,cTarget,cStyle,cDomId)

MENU SYSTEM - menu item p1 -- the URL to open when clicked p2 -- the text to display in the button p3 -- target frame or window p4 -- style text p5 -- DOM Id to use Aug 15,2016

menuDivider(cStr)

MENU SYSTEM- an item divider - accepts a string if omitted display an horizontal line

menuadd(nVal,cStyle)

MENU SYSTEM - step 4 Adds an optional menu on the right of menubar It works exactly as menu() function except that the whole section is aligned on the right of the menubar Notice: section is also collapsible JAN 7,2017

off()

returns a </DIV> sometime is useful to avoid cluttering the code

panel(nVal,cStyle)

Builds a panel nVal -- empty=open it, 0=close it, 1=heading, 2=body, 3=footer, other=default Dec 18,2015

panelbody(cText,cStyle)

returns a panel body

panelheader(cText,cStyle)

*** Undocumented (waiting submission)

panelfooter(cText,cStyle)

*** Undocumented (waiting submission)

popMsg(cStr, cLabel, cClass)

shows a brief screen-centered message that disappears when clicking anywhere on screen p1 -- the message to show p2 -- the text for the link p3 -- the class to use (default a standard link) IMPORTANT! it requires POPMSG.H to be included in the page with: include("/common/popmsg.h") Mar 07,2016 - Feb 8,2017

poster(xVal,nHeight,cStyle,nMode)

Returns the tags for a jumbo box with a background image p1 -- No parameters= opens the jumbo DIV without background. String= name and path of the image to use as background Number (zero) closes the DIV p2 -- number of pixel for minimum image height (default: full height) p3 -- a string for the style p4 -- if empty using poster size (huge) else a simple well Nov 5,2015 -- Feb 27,2016 -- Mar 12,2017

printh(cStr,nSize,cColor)

Prints out the string surrounded by HTML <hn> tags Parameters are: the string to print out in the web page the font size form 1 (larger) to 6(smaller) the color (default is black) Nov 4,2015

pullR(xVal,nMode)

-- returns the HTML code for a span right section p1 -- nothing=opens the span, 0=closes the span, passing a string will be returned inside the section p2 - N -- 1=returns the text into a variable, otherwise prints it to page Feb 27,2016 -- Mar 2,2017

radioButton(cName,cDescr,cVal,nVal,cId)

create a group of radio buttons parameters: cName...: name to assign (all radio buttons must have same name) cDescr...: description (shown on right of button) cVal...: the value to return if checked - if omitted returns the string N/A nVal...: 1 or 0 (checked or unchecked) cId...: DOM identifier to apply (must be unique for each button) Nov 2015 --Feb 12,2017

rowEnd()

-- closes a DIV row (i.e. returns </div>)

rowStart()

-- starting a DIV row

screen(nMode)

-- starts a container screen centered on page - max size determined by current CSS passing zero (or any other parameter) will close the container Feb 8,2017

shelf(nVal,cStyle)

starts or ends a Bootstrap row (to be loaded with drawers) p1 -- (N) - none or 1 = starts the DIV, 0 = ends the DIV p2 -- (C) - string for the style - default none Oct 2015, Aug 9,2016

sourcebox(cSourceCode)

-- show a box with text preformatted (i.e. honouring Line Feeds)

startRadio(cName,cLabel,nVal)

opens a radio button group cName -- DOM name and ID of the group cLabel -- text to display on the left of buttons nVal -- options (actually not used) If cName is omitted default is div_buttr_grp If cLabel is omitted default is Select an option Nov 2015 -- Feb 12,2017

startRow()

-- starting a DIV row

submenu(cStr,cStyle)

MENU SYSTEM: places a button and starts a dropdown list parameters: If a string is passed the dropdown group of items is opened and the string is used as title of the button. Notice: the menu items that will be part of the dropdown must be placed inside this section If 0 (or any number) is passed as parameter the dropdown section is then closed cStyle is the style to apply (optional) JAN 8,2017

well(nVal,cStyle,cContent, cId)

Opens and close a well p1 -C- no parameters or nVal=1 opens the well p2 -N- 0=closes the well p3 -C- a string for the style attribute p4 -X- optional content to show in the well p5 -C- DOM identifier id and name Alternative usage: well("Hello there")

Core functions

ajRun(cUrl, cParam, cTargId)

*** Undocumented (waiting submission)

ajaxCall(cUrl)

Makes an Ajax call to the given page. The page is executed asynchronously without notices. Nothing is returned to caller (page is run without a clue for the user).

ajr(cUrl, cParam, cTarget)

Ajax runner (internal use)

default(cVarName, anyValue)

Mimics the command DEFAULT TO of Clipper Declares a variable if not already declared otherwise its value is assigned NOTICE that cVarName must be passed as string (i.e. between double quotes)! Usage: myVar := ("myVar", "Hello guys") - Dec 15,2015

desktop()

Starts the DBFree library activating MaxScript extensions for a desktop page

jclear(cVarName,cType)

clears the browser local/session variable - variable will be voided but conserved in memory (i.e. will contain an empty string) cType -- 0/none=local, 1=session Dec 10,2015

jread(cVarName,cType)

Reads a variable from browser local/session storage cType -- 0/none=local, 1=session Dec 10,2015

jwrite(cVarName,cValue,cType)

Writes a string into a variable of the browser local/session storage cType -- 0/none=local, 1=session Dec 10,2015 Mar 17,2016

jzap(cType)

releases ALL the browser local/session variables (i.e. will contain an empty string) cType -- 0/none=local,

DBFree Library Functions – free.lib 2017

1=session Dec 10,2015

libVer(verbose)

Returns a string with the version number and build date of current library

web()

Starts the DBFree library activating MaxScript extensions for a web page

websettings()

-- activate the settings optimized for the web like dates for autocompletions etc. Jun 25,2017

Dates

CtimeDiff(cTime1,cTime2)

-- Accepts two time strings and returns a string representing the elapsed time in time format HH:MM Oct 5,2015

chrDate(dDate)

*** Undocumented (waiting submission)

curDate()

-- return current date in a string formatted for HTML5 date-picker (yyyy-mm-dd) Feb 11,2017

curYear()

-- returns the number representing the current year Feb 11,2017

dayName(nnn,cLang)

Accept a number between 1 to 7 and returns a string with day name in the desired language /US/UK/ES/OT/FR/DE/IT) Nov 29,2015

dayOfW(dDate, cLang, lShort)

Same as dayStr() but in english only and US format for compatibility only - it is preferable to use DayStr() Oct 5,2015

dayStr(dDate, cLang, nLen)

-- accepts a date and returns the name of the day Except for US the week starts at Monday Default for english is US (otherwise you must set cLang=UK) With nLen you can decide how long will be the string

daysFrom(date1,date2)

-- returns the number of days past from the given dates Oct 5,2013

daysOfM(date1)

-- accepts a date and return a number representing the days of month it belongs to. Honours the leap year (february of 29 days) Oct 5,2013

DBFree Library Functions – free.lib 2017

daysOfMonth(nMonth)

Accepts a MONTH NUMBER and returns the number of days of the given month Oct 5,2015

dtoh5(dDate)

-- transforms a standard date in a string formatted for HTML5 new input date-picker type (yyyy-mm-dd)
p1 -- the date to convert into formatted text if omitted will return current date Sep 13,2016 -- Feb 11,2017 -- MAr 24,2017

dtok(dValue)

Date to keyval Accepts a calendar date and returns a string representing the date in format YYYY-MM-DD Nov 28,2015

evalTstamp(cTimeStamp,cParm)

*** Undocumented (waiting submission)

firstDoM(NumOrDate)

-- accepts a calendar date and returns a date representing the first day of same month Oct 5,2013

firstDoY(nYear)

-- return a calendar date representing the FIRST day of the year Oct 5,2013 - Apr 10,2016

fulldate(dThis, cCountryID, IShort, IAddtime)

-- returns a string with day of the week, day month and year of the date passed. In missing assumes the current date. second paramater identifies the country (IT/FR/DE/ES/PT) WARNING: first day of the week is MONDAY except for USA/CAN which starts on SUNDAY this means that if you use lang=U it will follow north american rules to switch to UK (European) rules specify lang=UK if IShort is passes names will be shortened to three letters if IAddTime is specified current time will be added Jul 15,2010 - Nov 04,2015 - Mar 07,2016

initDates()

*** Undocumented (waiting submission)

isLeapYear(DateOrNum)

-- accepts a full date, a year number (4 digits) or a date-format string (DD/MM/YYYY) and returns true if it is a leap year Nov 29,2015

isLeap(dDate)

-- Accepts a date and return true if is a leap year WARNING: works only for current century -- use isLeapYear() for more precise caluclation Oct 5,2015

itDate(dDate, IFullname)

*** Undocumented (waiting submission)

itDateStr(dDate)

-- Accepts a date and returns a string in italian Oct 5,2015

DBFree Library Functions – free.lib 2017

itDow(ddd, lFullname)

*** Undocumented (waiting submission)

itMonth(dDate, lFullname)

-- returns the italian name of the month of given date Oct 5,2015

ktod(cStr,cCountry)

Keyval to date-- (reverse function of dtok() Attention: it returns a calendar date not a string Nov 28,2015

lastDoM(NumOrDate)

-- return the calendar date of last day of month Oct 5,2013

lastDayOfM(NumOrDate)

-- return a number representing of last day of given month month can be passed as number (1.12) or as calendar date Oct 5,2013

lastDoY(nYear)

-- return a calendar date representing LAST day of the given year Oct 5,2013 - Apr 10,2016

monthName(nnn)

returns the english name of given month (passed as number) Oct 5,2013

nowstr(cStr)

If no parameter is passed returns a string to be used for naming files or records and representing current date and time in format YYYY-MM-DD_HH-MM-SS Passing a string previously generated by the same function it reverses the function and returns the corresponding calendar date (time is lost). For more flexibility just use timestamp() Dec 10,2015

optDay(nDay)

*** Undocumented (waiting submission)

optMonthlt(dDate)

*** Undocumented (waiting submission)

optMonth(nMonth)

*** Undocumented (waiting submission)

optYear(nYear, nDig)

*** Undocumented (waiting submission)

secs2date(nSecs)

-- accepts the output of seconds() and returns the corresponding day as date Oct 5,2015

secs2days(nn1,nn2)

-- return the difference between two dates (passed in seconds) as number of days Oct 5,2015

DBFree Library Functions – free.lib 2017

timestamp(cParam)

- P1 (C) -- defines the output type: for example if date is Nov 3,2015 h7:50:52:14
 - none = 03/11/2015-0750
 - H = 20151103-07
 - M = 20151103-0750
 - S = 20151103-075052
 - T = 03/11/2015-0750
 - D = 2015-11-03
 - I = 03-11-2015_07_50
 - C = 2015-1103-0750-14

timeDiff(cStr1,cStr2)

-- returns the time difference between two time strings

today(cLanguage)

-- returns current date in the local format specified

- p1 - C -- AM/IT/UK/FR/DE/ES/JS/ANSI/DMY/MDY/YMD default AMERICAN Mar 28,2017

wcdow(dThis, cCountryID)

-- european name of the weekday (starts by monday=1) accepts only dates (IT/FR/DE/ES/PT) Oct 5,2015

wcmonth(dThis, cCountryID)

-- european name of the month (IT/FR/DE/ES/PT) accepts only a date Oct 5,2015

wdow(dThis)

-- european week (starts by monday=1) accepts dates or day number - return a number Oct 5,2015

week(dThis)

-- returns the number of current week based on the date passed as parameter: weeks go from 1 to 53 starting from january 1 nMode=1 week starts from sunday (default no) Oct 5,2015

Database functions

CloneRec(nRec)

*** Undocumented (waiting submission)

NoAjax()

Display the template for a basic ajax module to insert into the page (experimental) Jun 25,2017

appendBlank(cKeyFld,mKeyVal)

--- counterpart of APPEND BLANK command - returns the record number of new record cKeyFld is the

DBFree Library Functions – free.lib 2017

field to fill with mKeyVal (usually the field upon the index is based) Be always sure not to create empty records because they will cause error when reindexing Jul 10, 2007 - Montpellier

dbInfo()

-- shows all tables in use and their structure

dbgrid(cFldList,nRecToMove,nMode,cLegend,cLink,nType,cInputStyle,cTableStyle,cRow)

Loops down the table records and build a grid cell for each one of them

Each grid cell contains a separate web form with one of the fields of the record Each field reacts to changes by calling asynchronously this same page without reloading it

The Ajax call updates the field value in the table and immediately terminates without output

NOTE -- AJAX PARTS MUST BE INCLUDED IN THE PAGE OTHERWISE WON'T SAVE CHANGES

- p1 - C -- string for the fields to show (names separated by asterisks) default all
- p2 - N -- record to move to
- p3 - N -- 0=readonly, 1=edit
- p4 - C -- string for the legend (names separated by asterisks) default fieldnames
- p5 - C -- link for the button (if empty the button is not shown)
- p6 - C -- type of button: 0=popup, 1=modal, 2=link -- default popup
- p7 - C -- style string for input cells (INPUT tag attribute)
- p8 - C -- style for table (TABLE tag attribute)
- p9 - C -- style for rows (TD tag attribute)

dbskip(nQuant)

-- counterpart of SKIP command Oct 2007 - Florence

displayStru(cAlias,nOption)

Return a string containing the HTML code with the internal structure of the given table

- p1 -- (C) the alias to use, default current area
- p2 -- (N) output option: 1=return a string of text, else prints to screen

editpage(cFile,cMsg)

*** Undocumented (waiting submission)

enableOdbc()

*** Undocumented (waiting submission)

enableClipper()

*** Undocumented (waiting submission)

DBFree Library Functions – free.lib 2017

enableFox()

*** Undocumented (waiting submission)

enableDbase()

*** Undocumented (waiting submission)

fieldtype(fldNameorNum)

-- returns the data-type of a given field (expects a name or a number) Feb 14,2016

fieldlen(cField)

*** Undocumented (waiting submission)

fieldval(nFld)

*** Undocumented (waiting submission)

fieldcount()

*** Undocumented (waiting submission)

fieldfitter(cFldname, xCargo)

-- obsolete -- use fieldfit() Mar 18,2017

fieldfit(cFldname, xCargo)

*** Undocumented (waiting submission)

fieldNum(cStr)

-- accepts a string representing a field name and returns the numeral position of corresponding field in the record of table currently used Jul 2007

fieldList(xVal)

*** Undocumented (waiting submission)

fieldsize(fldNameorNum)

-- returns the maximum size of a given field (expects a name or a number) Jan 28,2017

filterkey(cAlias)

*** Undocumented (waiting submission)

fit(nField, xVal)

-- Ensures the value passed is converted in the corresponding data type of the field - use by DBGrid

fsize(nField)

*** Undocumented (waiting submission)

ftype(fieldNum)

-- returns the data-type of a given field (expects a number)

DBFree Library Functions – free.lib 2017

go2apps(cFile,cMSg)

*** Undocumented (waiting submission)

here()

*** Undocumented (waiting submission)

isDBD()

*** Undocumented (waiting submission)

isMemory(arg x)

*** Undocumented (waiting submission)

listNext(nRecs,cFldList,cLink1,cParm,cWebvar,cLink2,nLegend)

-- Prints out next nRecs of the current table

- p1 - N - Next records to show
- p2 - C - Field list as string containing two-digit numbers (zero-filled)
- p3 - C - page to call with the primary button created for each record
- p4 - C - string of common parameters to pass to the page
- p5 - C - webvar to use for passing the record number to the page
- p6 - C - URL with parameters for another page to call using secondary button
- p7 - N - 1=legenda of field names, none = no legenda

Usage; `listNext(10,"03,04,02,16",my,"&VA=EDIT","VR", "#top",1)`

pageCount(nRecs, nPgRows)

-- returns pages required to display all rows of the table currently open jun 2007

setDb(cDbPos, xRelative)

Obsolete -- returns the dos path pointing to the given database. If database does not exists it is created.

Passing any value as second parameter causes a nested database to be created (if not already existing).

Nested DB are subfolders of the folder of current database, which is a subfolder of _DATADIR

setDb3(cName, IShowOnly)

-- returns the dos path pointing to the given database. If database does not exists it is created.

- If passed IShowOnly (true) does not create database even if missing.

setLocalDb(cDbPos, INoCreate)

*** Undocumented (waiting submission)

setdbtype(cType)

*** Undocumented (waiting submission)

showMemory(cFile)

*** Undocumented (waiting submission)

Extended functions

appInfo(any)

-- returns infos on libraries installed (obsolete)

- passing any parameter will cause the info to be printed without comment tag /thus displayed to screen)

casual()

-- returns a string containing a casual number always different but progressive to be used for automatic DOM ID assignments Jul 08,2016

convert(anyval, cType)

-- convert a value of a type into another

- p1 -- the original value
- p2 -- the letter (C/D/L) of the desired value default is text

getAppVer()

returns the current application version

libInfo(cLibPath,IVerbose)

returns infos about a library file (default the one currently used) Parameters: library path to search, show extended infos

modInfo(any)

Provides some module infos to be embedded into the page as HTML comments

random()

-- generate a random numberbetween 100,000 and 999999 Aug 05,2016

Files and folders

FileNoDrive(cPathName)

*** Undocumented (waiting submission)

FileBase(cFullFileName, noSlash)

*** Undocumented (waiting submission)

FileName(cFullFileName, nOption)

Returns the filename formatted as determined by nOption:

DBFree Library Functions – free.lib 2017

- p1 (C) – Filename and path
- P2 – (N)
 - No parameters = filename without drive and path but with dot and extension
 - 0 = dot and extension are trimmed out
 - 1 = drive and path omitted, but with dot and extension

FilePath(cFullName, INoDrive)

*** Undocumented (waiting submission)

breadcrumb(cFolder,cTarget,cClass,cStyle)

Builds a breadcrumb list of links to be printed out in the current position.

- P1 – folder to set as starting point
- p2 – target frame (defaults to _self) cClass=
- p3 – class to use (defaults to button xs small)
- p4 – string for optional style override

createDir(cPath)

-- create a folder into current path Obsolete -- Just use makeTree()

curDosDir(nRemoveDriveLetter,nRemoveSlash)

-- shows the path of current DOS dir if passed parameters (0/1) omits the drive letter and/or the slash at the end of the string 0=option inactive, 1= option activated

curDosPage()

*** Undocumented (waiting submission)

curDrive()

-- returns the drive letter (and columns) where wb site is located Aug 24,2016

curWebDir(full)

-- ritorna il percorso web della cartella che contiene il file MSP attualmente in esecuzione

curWebPage()

*** Undocumented (waiting submission)

dir(cStartPath, cLink)

Returns the directory listing of current DB

Esempio: cLnk := "/apps/assistant/browse-edit.msp?VAR_TABLE=|" <%=dir(cLnk)%>

dirName(cPath)

returns the folder name of given path Nov 14,2015

directory(cFolder,cMask,nTarget,nMode,clcon)

-- Display a list of files depending by parameters passed:

- cFolder -- path to use
- cMask -- mask for files
- nTarget -- 0 or none=self, 1=pop up modal
- nMode -- 1= with info column
- clcon -- icon to display before the name of file

documentRoot()

-- same as webroot

dos2web(cFile)

-- trasforma un percorso DOS in percorso web a partire dalla document root del server web

fcountByTime(cFold, cMask, cTime, cSign, nFlag)

-- counts (or list) how many files are newer/older than given hour time. it returns a number or a formatted string depending by nFlag

- cFold == the folders to be scanned
- cMask == mask for files to search
- cTime == time in format HH:MM:SS (only hours will be considered) (default is current hour time)
- cSign == sign for comparison (more,less,equal - one char string) (default is: older than cTime)
- nFlag == any number: if passed return the result as a list of files separated by the \$ sign (to be used for array loading) If nFlag is passed and no files satisfy the condition will return \$ Jan 19,2016

fileScan(cFile,nRowsize,cEol)

-- read a file using low level routine

fileDrive(cPathName)

*** Undocumented (waiting submission)

fileclean(cFile)

*** Undocumented (waiting submission)

filefolder(cFullFileName)

*** Undocumented (waiting submission)

fileBone(cFile)

*** Undocumented (waiting submission)

fileExt(cFilename, xExcludeDot)

*** Undocumented (waiting submission)

fileSource(cFile,nLineLen)

*** Undocumented (waiting submission)

file2str(cFile)

*** Undocumented (waiting submission)

fileread(cFile,nMode,nBuffSize)

A substitute of memoread() First parameter is the file to read: if not existing the function will return an empty string. Second parameter tells the kind of file to read: None or zero=standard, 1=only 8-bit ASCII chars Third parameter is buffer size (default=auto detect) Use this function to avoid unwanted garbage chars when reading or saving into memo fields and/or small files (under 2 Kb)

filewrite(cFile,cText,nMode)

Substitute for memowrit - it can overwrite or append to existing file plus it doesn't insert special end-of-file markers unlike memowrit() nMode -- 0=overwrite, 1= insert at the beginning, 2=insert at the end default is 0

getBytes(cFile,nStart,nBytes)

reads nBytes from any file starting at pos nStart (low level function)

include(cFil, cSed, cMarker)

-- usage: ? include(cTextFile) html ? include("/common/header.h",cSeed)) html For direct usage see: preload() For paseing the seed the file to be included must contain a call with this sintax: XY=* then the asterisk will be replace with the sed passed as parameter. If a third parameter (string) is passed it will be used instead Dec 30, 2015 - Sep 30,2016 - Feb 4,2017

isDir(cFolder)

*** Undocumented (waiting submission)

isFolder(cFolder)

*** Undocumented (waiting submission)

jlink(cFileAndPath)

*** Undocumented (waiting submission)

label(cFilename)

*** Undocumented (waiting submission)

listFunc(cFile)

*** Undocumented (waiting submission)

listFiles(cThisFolder,cMask,nOpt,cCargo1,cCargo2)

-- returns a string containing a list of files of given folder honouring the given mask separated by cCargo1 The string can then be processed to an array using str2arr()

- p1 - C -- folder DOS path

DBFree Library Functions – free.lib 2017

- p2 - C -- DOS mask for files to show
- p3 - N
 - 0=no printout (returns all in a string with separators)
 - 1=with full pathname
 - 2=filename only
 - 3=as web link
 - 4=as dos link
- 5=as list of web links
- 6=as popup button with parameters passed in cargo1 and class passed in cargo2
- 7=as link with parameters passed in cargo1 and class passed in cargo2
- p4 - C -- cargo parameter 1 (default is #)
- p5 - C -- cargo parameter 2

Usage:

```
listFiles( cDir,"*.dbf",qq,"btn btn-xs btn-danger")
```

```
listFiles( cDir,"*.dbf",7,qq,"btn btn-xs btn-danger")
```

listFolders(cThisDir,xFlag)

-- returns a list of subfolders contained into cThisDir xFlags determines the type of output:

- empty -- names between tags
- 1 -- names between asterisks
- 2 -- names between square brackets
- 3 -- number of folders found
- 4 -- an HTML string with all links
- 5 -- clickable buttons

makeTree(cFinalPath)

-- build all the upper level folders necessary to fulfill the giveb path.

pageName()

-- per compatibilità con apache: da usare al posto di getenv("SCRIPT_NAME")

pageFile(nMode)

-- returns the DOS path and name of current web page p1 -- (N) passing 1 will return the name with slashes doubled for using with javascript Dec 2006 -- Sep 01,2016

plainPath(cPath)

returns the folder name of given path (similar to dirName but using different method) Nov 14,2015

preload(cFile, cSed, cMarker, cTagId, cString)

reads a text file and inserts its content directly into the HTML flow of page for indirect usage see also: include()

- p1 - C -- file to load
- p2 - C -- seed to pass to page
- p3 - C -- string to search and replace with the seed
- p4 - C -- HTML tag to search for insertion
- p5 - C -- string to insert in the tag

Usage:

```
preload("/common/header.h", cSeed, "XYZ", "pagetitle", wordleft(filebone(pagefile())), ".")
```

readfile(cFile, nMode, nBuffSize)

Same as fileread()

showpath(cFolder, nOpt, cLL, cRR, cSty)

*** Undocumented (waiting submission)

storage(any)

*** Undocumented (waiting submission)

subDir(cTxt)

*** Undocumented (waiting submission)

sysDrive()

Return a 2-character string for system drive (i.e. C:) jun 29,2016

upDir(cTxt)

*** Undocumented (waiting submission)

wdrive(cFilename)

*** Undocumented (waiting submission)

web2dos(cFile)

-- trasforma un percorso web in percorso dos Dec 2006 -- Sep 12,2016

web2java(cTxt)

*** Undocumented (waiting submission)

webRoot(nStyle)

Return the string for the path of Web Root

- p1 – N -- 1=DOS, 2=DOS Nodrive

compatible with MonGoose Web Server -- if used in DBDesk returns root dir - DBMAX returns current main

DBFree Library Functions – free.lib 2017

folder -- compatible with IIS10

writefile(cFile,cText,nMode)

Same as filewrite()

HTML related functions

CloneRecord(nRec)

Takes a record and duplicates it adding a new one Jul 15,2007

HTML5(nMode)

-- needed to handle some HTML5 elements (e.g. calendar dates)

- p1 - N -- none==enables HTML5 support, 0==disable

alertOnce(cMsg,cCookieName,cStyle)

-- display a warning panel red and white only once for session

- p1 -- text to display -- if empty just opens the alert box and return if passed zero or any other numeric closes the box
- p2 -- the ID of the session object (cookie) to be checked or saved becomes the DOM ID of the panel DIV the alert box is displayed in (default name/id is 'warnonce')
- p3 -- a string for the optional style

blanks(nChars)

-- returns non-breaking white spaces HTML Mar 18,2017

breakline

return a HTML tag for line break

brk()

returns an HTML break tag Mar 2,2017

buildComment()

prints out in the HTML code a template for top-of-page comments Call this function, expose the HTML code in browser then copy and paste in your code

centred(nVal)

returns a DIV for centering text in the page.

- P1 – N – none==open the DIV, i0==close the DIV

clock(xPrint)

-- return the javascript code for a clock

DBFree Library Functions – free.lib 2017

- p1 - any -- if passed returns the code else outputs to page

closeWebPage()

Same as ClosePage() but to be used in CGI style Nov 5,2015

closePage()

inserts the code for closing a page (and its container) Oct 11,2015 Mar 7,2016

codebox(cStr,cId)

display the given text with option ID as PRE element (useful for displaying source code) Oct 31,2015

commentln(cStr)

-- inserts an HTML comment containing with the string passed

comment(cStr)

-- inserts an HTML comment block containing the string passed

cvttext(cStr)

-- internal -- used by text() e println()

displayRecord(nRec, nStyle)

-- generate a web form (read only) taking all the fields of current record:

- p1 – N --Record number to go to
- p2 – N –display style to apply
 - 0=clean
 - 1=row num
 - 2=show type
 - 3=type and size
 - 4=no output, only text

Usage: `<% =displayRecord(recno(),1) % >`

display(cFldList, nStyle)

-- AUTOMATICALLY DISPLAY A FORM FOR CURRENT RECORD

- p1 -C- list of fields to show (2-digit field number) separated by asterisks or commas (default display all fields))
- p2 -N- display mode:
 - 0=clean
 - 1=row num
 - 2=show type
 - 3=type and size

DBFree Library Functions – free.lib 2017

NOTICE: the pseudo-form is DOM identified upon the record number with prefix RECD_

editRecord(nRec, cFieldList, cUrl, nMode, nDecs, cJaction)

Generate a customizable and editable web form for current record:

- p1 - N -- record number to go (default current record)
- p2 - C -- string with field names to display separated by asterisks (default all)
- p3 - C -- the target file to call submitting the form (default self)
- p4 - N -- none or 1=edit allowed, if 0 only display
- p5 - N -- decimals to use for numerics
- p6 - C -- javascript action to bind to SAVE button (by default passes "window.parent.location.reload();" to close a modal and reload the parent page after saving the page)

flds2form(cFldList)

Generate a raw web form for current record containing only the given fields.

- P1 – C – a list of fields as a string where each field to show is represented by its 2-digit number separated by asterisks

Usage: `flds2form("01*03*05*10")`

hyperlink(cUrl, cDescr, cTarget, cStyle, cClass)

-- return a string to be used as HTML link

- p1 (C) - URL of the link
- p2 (C) - text to show in the link
- p3 (C/N) - target (if passing "_pop" will open in a popup) or if passing a number is ignored
- p4 (C) - optional style string
- p5 (C) - optional class override

image(cUrl,nHeight)

-- return the HTML tag for an image

- p1 - C - image URL - if file does not exist will return empty
- p2 - C/N - string for height (with measuring unit) or height in pixels

inputRadio(cName, cValue, varToMatch)

*** Undocumented (waiting submission)

input c(cName,cId,cValue,cLabel,cPlaceholder,cHelp)

-- returns a form field

input m(cName,cId,cValue,cLabel,cStyle,cHelp)

-- returns a form field

DBFree Library Functions – free.lib 2017

item(xVal,nType)

-- builds an HTML item list with or :

- p1 – N/C – empty=start the list, 0=close the list , a string=the item
- p2 - N -- 0=pointed (default) - 1=numbered

jhide(cDiv)

-- immediately hide an element of the page (not suitable for using with buttons or javascripts!)

p1 (C) - DOM name of the DIV to hide

Use in WebObjects actions or for hiding elements of static code loaded by using the functions *include()* or *preload()*

js(nVal)

--- opens a javascript code section in the page

- p1 (N) – none=open the section, 0=close the section

jumptop(cMsg,cPos)

Returns a link for moving to top of page

- p1 (C) - optional, text to display in the link
- p2 (C) - position "L" or "R"

junhide(cDiv)

-- immediately unhide an hidden element of the page (not suitable for using with buttons or javascripts!)

- p1 (C) - DOM name of the DIV to unhide

Use in WebObjects actions or for hiding elements of static code loaded by *include()* or *preload()*

linkit(cUrl,cLabel,cTarg)

-- produces a link from a field (function to be used in commands)

Usage:

```
cLink := "customers.msp?VAR_SOMEVAR="
use countries
list off iif(empty(CODE),"...",CODE), linkit( cLink+CODE, COUNTRY, "frm2")
```

link2menu(cDir, cChoiches, xDebug)

-- returns a link from a list of possibilities in the given directory by default looks into current directory, passing a zero looks in upper one

If an alternative list is passed this must be in form of a string with filenames (incl. extension) separated by asterisks

link2app(cDir, xDebug)

-- build a link to a Web App Mar 15,2012

load(cPageUrl, cHeader, cFooter)

-- open a page or builds it if not exists Mar 15,2012

navButtons(cParms,nIndexes,xOutput)

– first of 2 functions to generate a form for browsing, must be followed by a call to function navBrowse() . If third parameter passed just outputs the code to screen.

Usage:

```
<%=navButtons("?")%> //(shows instructions)
```

navBrowse(cFldList, cEditPage, cParmList, xOutput)

-- second of 2 functions to implement a browse: the call to this function can be omitted and substituted by your own browse code

Usage:

```
navBrowse("?") (shows the code for customization)
navBrowse("9,3,5,4") (field list)
navBrowse("1,2,3","edit.msp","XY") (webvars to pass - VR is the default)
```

navEdit(cUrl, cFldList, nModal, xOutput)

-- simple record editor to be called by the button of navBrowse

- p1 - C -- target page for the form
- p2 - C -- list of fields number separated by asterisks
- p3 - N -- running in 1=modal window - 0=standalone page

Usage:

```
navEdit(cSelf,"01*02*03")
```

openWebPage(nOpt)

Same as OpenPage() but to be used in CGI style Nov 5,2015

openPage(nOpt)

inserts the code for a basic page header (including container open) Oct 11,2015 Mar 7,2016

optByDb(cTab,cFld)

-- popola le option di una select usando un DB come fonte la tabella deve essere sul percorso dati corrente

Usage:

```
<%=optByDb("GNX_UTENTI", "USR_NAME")%>
```

pageReload(nOpt)

causes the page to be immediately reloaded

pl(cTxt,nSize)

print a line of text (or any other text) as HTML formatted paragraph

- p1 -C- text to print
- p2 -N- dimension (larger to smaller 1,2,3,4,5,6) default is 5 (<h5>)

DBFree Library Functions – free.lib 2017

printlni()

-- visualizza in HTML il contenuto di un file ini Oct 5, 2013

println(xVal,cSize)

-- (print line) prints out the given string between HTML tags for the paragraph size

- p1 -- the string
- p2 -- a size code as a string:
 - minute,normal,medium,large,big,huge
 - 7,6,5,4,3,2,1 (small to large)
 - xs,s,m,l,xl,xxl,xxxl

rec2form(nFldStart,nFldEnd)

-- Builds a list of entryfields for current record

ruler(nDist,cColor)

return a HTML tag for an horizontal ruler

- p1 -- optional margin top and bottom distance in pixel
- p2 -- optional color string

save2Record(nRec, aWebVarList, cAlias)

-- Save into record the incoming webVars matching field names (special function)

- p1 -N- record number to go to (mandatory))
- p2 -A- webvars stack page part (mandatory - must be added to current page)
- p3 -C- alias name of the target table

TO BE CALLED FROM AUTOMATIC FORM BUILDERS like formstart() etc

saveRecord(nRec, aWebVarList, lDebug)

Saves all the webVARs prefixed by EF_ into matching fields of current record. The prefix is ignored in the matching. WebVars not matching are discarded.

- p1 (N) – the number of record to process (that must match current record)
- p2 (A) – webVars stack
- p3 (L) – if passed caused some output useful for debugging

setStyle(cStyle)

-- sets the styles for bootstrap use

- p1 (C) - no parameters=default.css
 - 0=bootstrap
 - 1=bootstrap-theme
 - 2=theme

DBFree Library Functions – free.lib 2017

- 3=takes the name from CSS entry in mspconfig file

shim(nPixels,cDivTxt)

-- inserts a DIV of the given height (if passed a numeric) or a text if given a text Second parameter is the content of the div

Usage:

```
shim(60)
shim(60,"Hello")
shim("class=well","HELLO") Jul 20,2012
```

showMaster(aOfFlds)

-- show master record of a master-slave table relation:

- p1 (A) – array of fields to show

showSlave(aOfFlds)

-- show slave record of a master-slave table relation:

- p1 (A) – array of fields to show

strSafe(strText)

-- returns safe code for preloading in the RTE

text(xVal,cSize)

-- returns a string between HTML tags for the paragraph size

- p1 - C -- the string
- p2 - X -- a size code as a string or number:
 - minute,normal,medium,large,big,huge
 - 7,6,5,4,3,2,1 (small to large)
 - xs,s,m,l,xl,xxl,xxxl

Usage:

```
print( text("Hello world","huge") )
```

url2form(cTxt)

-- returns a list of fields from current record as a string with a separator

- p1 -C- list of fields (2-digits) default all fields
- p2 -C- separator to use (default blank space)

Usage:

```
flds2str("02,03,05","*")
```

webvar(cName,cValue)

Returns an hidden entryfield for web forms

DBFree Library Functions – free.lib 2017

- p1 -C- the name of the webvar (DOM Name for entryfield)
- p2 -X- the value (that will be converted into string)

Login helpers

clearSess(cThisSeed)

*** Undocumented (waiting submission)

displayJvars(cJvarStr)

*** Undocumented (waiting submission)

displaySession(cThisSeed, cStyle)

display the stock session vars (excluding those added by modules) as declared in the header

displayGlobals(cFile,cStyle)

*** Undocumented (waiting submission)

isLocal(clpAddr)

*** Undocumented (waiting submission)

loginButton()

*** Undocumented (waiting submission)

logoutButton()

*** Undocumented (waiting submission)

logwrite(cMsg, cUserName, cCodOp, cCargo, nTTime, cClass)

*** Undocumented (waiting submission)

setId()

Memory functions

WebVars2url(aArr, cExclude)

-- trasforma tutte le webVar dell'array in una stringa da URL

WebVars2Fields(aArr, cExclude)

-- trasforma tutte le webVar dell'array in campi hidden il secondo parametro è la maschera di esclusione

WebVars()

-- prints out the content of the webVars array

returns the current number of elements of the array WARNING: the webVars stack MUST be declared BEFORE

DBFree Library Functions – free.lib 2017

calling this function NOTICE that if the file could not be saved returns the string error.mem

cfg2mem(cIniFile)

Reads a configuration file CFG and loads its content into memory WARNING: if file does not exist will be generated with default values

file2vars(cTextFromJS,IVerbose)

loads variables in memory from file (file must contain variable names and values pairs [varName=varvalue]
RETURN a string with separator containing the list of var names Second parameter is for debug and only show the process WARNING: uses a private array named aParsed that MUST ne passed to function parseStr()
DONT CHANGE THE NAME

getCargoFile(cSeed)

Returns a filename for storing Z-VARS Nov 4,2015 - Dec 10,2016

getMemFile(cSeedStr, vEncrypt, vUsePort)

returns a filename for storing session variables parameters are for encryption and including port If this is not necessary just use getSessFile() Oct 5,2013

getMemDir()

returns the path where memory files are actually stored

getMemPath()

returns the path where memory files are actually stored

getSessFile(cSeed)

Returns a filename for storing X-VARS (ignoring encryption and port) Nov 4,2015

getSession(cSeed)

Returns a filename for storing X-VARS (ignoring encryption and port) same as getSessFile

getseed(nVal)

generate a new progressive number (as string) to be used for naming sessions passing nVal=0 the seed will be fixed (not change) using the IP of the current connection instead of a unique ID Oct 5,2013 -- JAn 13,2016

ini2Mem(cIniFile)

Reads from a INI file and loads all the content into memory p1 - C -- the file to read (default "app.ini") if not exists it is created NOTE: variables declared using a INI file must be prefixed according to standard rules those not recognised will be intialized as strings The functions outputs a list of the variables loaded in memory June 23.2017

ini2vars(cFilename)

-- reads a given ini file and intializes all the vars if there are sections (between square brackets) the name of the section is added to the name of the variables they contain

listMem(memArray)

returns the content of a memory array (typically aVars)

listVars(cStr,cSep)

-- inspects a formatted string and extracts variable names it may contain default separator is an asterisk

loadini(cFile)

-- reads an ini file standard and initialies the variables with their values it returns a list of names for vars created separated by spaces Oct 5,2013

loadIniVars(cFile,nRowsize,cEol)

Loads variables into memory from a large ini file Default row size is 128 and default end-of-line marker is CHR10

makeSeed(nVal)

wrapper for getseed() Oct 5,2013

restmemory(ddd)

*** Undocumented (waiting submission)

savememory(ddd,ppp)

-- saves the memory variable into memory.mem file Usage: x2 := "WORLD" ? savememory("x*") clear memory ? restmemory() ? x1, x2 Oct 05,2013

saveSession(cSeedStr)

-- saves all X-Vars to a session file: returns the name of the file Nov 18,2015

undefined(cVarName)

-- ritorna vero se la variabile passata (tra virgolette) non è stata dichiarata prima

undef(cVarName)

*** Undocumented (waiting submission)

varBar(cSeed,cAction)

-- shows a bar with session variables must receive the seed and the action code as parameters WARNING: needs the appropriate javascript to be loaded BEFORE being called

webVars2str(aaa)

Converts all the elements of the array passed as parameter into a string to be used as URL (EF_ as formfield prefix and &= as separator) If the parameter is not an array returns an empty string Oct 5, 2013 - Nov 18,2015

webv2f(aArr, cExclude)

Same as webVar2fields()

Numerics

hex(nnn)

*** Undocumented (waiting submission)

hextoken(nnn)

*** Undocumented (waiting submission)

keepVal(cStr, nLow, nHigh)

*** Undocumented (waiting submission)

range(nVar,nMin,nMax)

*** Undocumented (waiting submission)

setDec(nnn)

*** Undocumented (waiting submission)

Session handling

getGlobFile()

returns the name of file containing global variables

getJseed(cVarName, nAction)

* retrieves a seed previously saved in browser local storage and optionally reloads the page You can specify the name of the item to retrieve (default is JSEED uppercase) nAction -- 1=reload, empty or 0 only display the result

initSession(cSeed, nOption)

initializes as public, preloads and saves X-Vars for the given seed returns the name of the file where the vars have been saved

- p1 -- the seed to use (if empty creates a new one) REMEMBER to save the x variables created here
- p2 -- if any value is passed saves the variables

initGlobals(norewrite)

initializes the global variables (if parameter passed does not save to disk) it also creates a text file with a default configuration Globals are stored in a public array named _GLOBALS Initializes, preload and save G-Vars (Global variables) returns the name of the file where the variables have been saved

initXvars()

-- initializes as public all X-Vars (no preload is done) see also initSession()

initJvars()

Initializes as public the variable to be used in J-Detection forms

DBFree Library Functions – free.lib 2017

initJseed(cSeed)

* initializes the browser-based seed system The item saved into localStorage is named JSEED (containing the current seed) The function returns the javascript code as a string (after execution) for debugging

initJ(cNewSeed,cTarget)

*** Undocumented (waiting submission)

readJvar(cVar,cOpt)

-- Reads a variable stored in browser Nov 3,2015

restoreSession(cSeed,nOption)

-- restore a session from a given seed -- see initSession() p1 -- (C) - the seed if seed is not passed or is a dollar sign a new session is created if the sees is passed but it is empty no session will be created The function returns the name of memory file used (or an empty string if seed was empty) p2 -- (N) - passing 1 force the loading of mem file Nov 3,2015 - Jan 2,2016

saveJseed(cStr,cVar)

Saves cStr as local storage item cVar If cVar is omitted saves as JSEED

showSession(nOpt)

-- prints out the xVars if any parameter passed return all as string

showGlobals(nOpt)

-- prints out the globals p1 -- if passed uses CR_LF, otherwise HTML break line Aug 14,2016

showJvars(nOpt)

-- prints out the session variables xVars p1 -- if passed uses CR_LF, otherwise HTML break line Aug 14,2016

showCargo(nOpt)

Same as showZvars()

showZvars(nOpt)

-- prints out the cargo variables Z-Vars if any parameter passed return all as string

Strings

Capital(cStr)

Converts first letter to uppercase and the resto fo the string to lowercase

StripChars(cTxt , nUpperASCII)

Removes all non printable characters (less than 65 or more of 254) Second parameters is to tell what is the upper limit (typically 126 instead of 256) Oct 5,2013

WordRight(cStr, cSep, lOffset)

returns the rightmost portion of the string delimited by cSep if lOffset is specified the separator is omitted from result see also rightMost() Jul 15,2007

WordLeft(cStr, cSep)

returns the leftmost portion of the string delimited by cSep jul 15,2007

block(nParm)

CGI style -- output a division no parameter=open the DIV otherwise 0 to close it 1 or empty=left alignment, 2=center, 3=right

break()

CGI style -- output a break

c2l(xVal)

check if the given variable contains a logical value Accepts strings in search of an identifier for logical value strings: YES,NO,Y,N,1,0,TRUE,FALSE are considered logical values numbers: 1 or 0 are considered logical values Any other result returns an empty string

c2n(anyVal,nMode)

-- converts any value to number if nMode=1 all zeros in the strings are ignored Dec 18,2015 -- Feb 16,2016

cc(anyVal)

-- convert any value to character. Numbers are truncated to 0 decimals Apr 13,2017

cleanUp(cVar)

Removes angular brackets,single and double quotes, CR_LF and the words href,script Returns a string with the HTML equivalents of removed chars Oct 5,2013 - Feb 8,2017

cleanStr(cVar)

-- clean up a string to be then used with wz_tooltip.js Oct 5,2013

code2memo(cTxt)

-- replace all angular brackets with square brackets used for neutralizing the HTML code before saving in a memo Mar 24,2017

contains(cTarget, cString, nIgnorecase)

-- Return true if the first string contains the second

p1 -- text string to be emaninated

p2 -- the string to look for

p3 -- none==case sensitive, 1==ignore case

cutAt(cString, cKey)

(*deprecated) -- cut a string starting to left until it reaches cKey same as using leftmost()

DBFree Library Functions – free.lib 2017

dblSlash(cTxt)

-- same as jslash (obsolete)

decval(cStr,cSep)

Accepts a string representing a number with a decimal separator and returns the equivalent number Second parameter is for the separator (default is a dot or comma) usage: `decval(MSPFieldValue("ANY"))` Jan 4, 2016

dslash(cTxt)

-- same as jslash (obsolete) Jul 15,2007

fChar(val)

-- Accepts any value and returns it as string – similar to zz()

ffix(cStr)

Filename Fixer - Ensure that a filename has a drive letter and removes double slashes from the path - converts web slash to DOS current drive and path is taken as defaults. Can also be used to create a filename from a numeric value.

fixSlash(cTxt)

Replaces single slashes with back slashes and removes double back slashes Returns a cleaned string suitable to use in DOS operations -- see also ffix()

forEach(cStr,cSep,cFunc)

Usage:

```
cList := "jan,feb,mar,may"  
? forEach(cList,,"upper(x)")
```

htmlMemo(cTxt)

-- replace all angular brackets with square brackets used for neutralizing the HTML code before saving in a memo see also memoHtml()

ini2Str(cFile)

-- wrapper for iniToStr

iniToStr(cFile)

-- return the content of a ini file between fences

iniVar(cFile,cVarname)

Search cVarName into an .ini file and returns true or false if found

intval(cStr)

Accepts a string and returns a number without decimals.

This is somewhat the counterpart of decval() function. Returns an integer number ignoring decimals (number is not rounded) but decimals simply stripped away, unlike c2n() Useful in some rare cases to clean up recno() if

passed via webVars

jInnerHtml(cDiv,anyVal)

Same as jchange

jInsertSeed(cDiv,cThisSeed)

-- add seed to a static link that has no parameters if there are already these parameters please use jaddP()
Usage:

```
jInsertSeed('div_login',cSeed)
```

jaddP(cDiv,cElem,cStr)

-- adds a parameter (typically the seed) to a static URL parameter is added at the end of URL without checking if already exists

if cElem and cStr are omitted it is added the seed

jchange(cDiv,anyVal)

-- Change the HTML content of a given DOM element (using the ID= not the NAME=) with the value passed as parameter. Value is always converted to string.

Usage: `jchange("myCounter", nnn)`

jfieldval(cId,xVal,cType)

changes a value of a webform element (i.e. ENTRYFIELDS) deciding if the final value is string or numeric For other elements use jvalue() instead

Usage: `jfieldval("VR",recno(),"N") jfieldval("VA","GO")`

jslash(cTxt)

-- double the forward slash to allow use of strings returning DOS paths into javascripts

jtitle(cStr)

-- changes the title of the web page Dec 18,2015

junslash(cTxt)

-- removes double slashes form the string (opposite of jslash) Oct 5,2013

jvalue(cDiv,cElem,anyVal)

-- Change the HTML content of a given DOM object (using the ID and its element) with the given value. Value must be any value and will be a string in resulting java code If element is omitted will default to innerHTML (acting exactly like jInnerHTML function) -- usage:

```
jchange("div_date", "value", dtoc(date()) )
```

leftMost(cString, cKey)

-- return the leftmost part of a string delimited by cKey same as using cutAt()

letterOnly(cStr)

-- strips away any characters except ASCII letters (USA)

DBFree Library Functions – free.lib 2017

- p1 - C -- the string to clean up

memo2code(cTxt)

-- converts text (angular brackets) back to HTML (it is the reverse of code2memo)

memoClean(strText)

-- returns safe code for preloading in the RTE same as safeRTE

memoHtml(cTxt)

-- replace all square brackets with angular brackets used for activating HTML code deactivated by htmlMemo()

n2c(xx,nn)

-- exactly equivalent to zz() but by default decimals are off

neutralize(cTxt)

-- neutralizing javascript and maxscript code in a text then to be shown only by using PRE tag (or saved in memo field)

nice(cFile,nOpt)

-- strips underscores from a string and replaces it with blank spaces while converting first letter uppercase

- p1 - filename
- p2 - Option=1 converts all first letters of each word uppercase Option=2 removes the file extension

nn(xVal)

-- returns always a number from string, dates or logical if a date is passed returns the days past from today if array, memo or none returns 0

noSpaces(cStr)

Removes all spaces from the string

occur(cString,cTarget)

Counts the occurrences of <cString> into <cTarget>

- p1 -- (C) - string to search for
- p2 -- (C) - text where to search
- p3 -- none==case sensitive, 1==case insensitive

onlyNum(cStr,nMode)

Returns a string stripped from all characters and symbols and containing only numbers

if nMode=1 all zeros in the string are ignored

p(cStr)

Abbreviation for PRINT() function NOTE that this function prints out DIRECTLY to web page without

DBFree Library Functions – free.lib 2017

intermediation of any sort

padc(cString, nLength, cCaratt)

Centers the text Returns a string of nLenght filled with cChar

padl(cString, nLength, cCaratt)

Padding the string to the left Returns a string of nLenght filled with cChar on left

padr(cString, nLength, cChar)

Padding the string to the right Returns a string of nLenght filled with cChar on right

parseFor(cStr, cVname, cOpenTag, cCloseTag, cTokenTag)

-- Parse a string cStr in search of the text cVname (which represents the name of a variable) and its value (separated by cTokenTag) by using cOpenTage and cCloseTag as delimiters

Example:

```
parseFor(cTxt, "NavScreenSize")
parseFor(cTxt, "NavScreenSize", "[", "]", "=") //--same as above
parseFor(cTxt, cStr, "(", ")", ":")
```

parseStr(cStr, cOpenTag, cCloseTag, cTokenTag)

-- fills an array aParsed (to be declared before using the function) with the results of the search of ANY string delimited by cOpenTag and cCloseTag and separated internally by cTokenTag By defaults tags are square brachets and the separator is an equal sign.

Example:

```
aInit("aParsed", "")
for iii=1 to parseStr(cTxt)
? aParsed[iii]
next
```

print(cStr)

same as PRINT command but suitable for direct output to a web page because does not insert a
 as end of the line. Specific for usage with Bootstrap Functions

The passed parameter is converted to string. If no parameter was passed returns an empty line (CR+LF)

printline(cStr, nParm)

CGI style -- output a line of text

- p1 (C) the string to output
- p2 (N) - a number for the size tag 1=<h1> 4=<h4> etc. if no cStr is passed prints an horizontal line

readIniVar(cFile,cVname,cDtype)

Returns a value of a variable stored into an ini.file If file does not exists returns NIL (use writeinivar() to write the vars to file)

- cFile = full file path and name of the file to read cVname = variable to read (case insensitive)
- cDtype = 1 char string for type of data to return. Accepts N/C/L.

rec(nDigit)

return the current record number as a string

- p1 -- (N) number of digit (zero leaded) of the returned string

revStr(cStr)

reversing the content of a string

reverse(cStr)

reversing the content of a string same as revStr()

rightMost(cString, cKey)

-- return the rightmost part of a cString delimited by cKey

rslash(cTxt)

Ensures the last character of the string is a backslash

splitVal(cStr,nVal,cSep)

-- recupera da una stringa la porzione di testo prima o dopo il separatore <cSep> Se <nVal=0> la porzione sinistra, altrimenti la destra. Per recuperare la coppia completa chiamarla due volte. Es:

```
cTxt := "mario/rossi"  
c1 := splitVal(cTxt,0,"/")  
c2 := splitVal(cTxt,1,"/")
```

Utile per leggere le coppie dai file INI

ssoff(cStr)

(server side tags off) - removes all server side tags from the given string

strcount(cTxt,cKey)

count how many occurrences of a string are found into another

- cTxt -- the text to be searched
- cKey -- the string to count within the text see also occur()

trimRight(cStr, nPos)

-- Trims a string of nPos characters starting from right

if cStr is shorter than nPos the string returned is null

trimLeft(cStr, nPos)

-- Trims a string of nPos characters starting from left

if cStr is shorter than nPos the string returned is null

trimLeftAt(cString, cKey)

-- return the leftmost part of a string delimited by cKey same as using cutAt()

trimRightAt(cString, cKey)

-- return the rightmost part of a string delimited by the first occurrence from left of cKey see also rightMost()

writelnVar(cFile1,cVar1,cValue1)

Salva una variabile e il suo valore in un file ini di tipo DBFree Se il file non esiste viene creato Ritorna il nome del file INI in cui ha scritto

wstr(xVal)

(web string) obsolete - senza spazi davanti e dietro usata per passare valori via URL - see zz()

zz(anyVal,nDecim)

-- convert any value to a string Updated and fixed version to reflect the decimals setting with no limit on length Second parameter is to set how many decimals to use

Web functions

DnsList(cFile)

-- returns the content of local DNS file (Windows XP and above) Oct 4,2013

callingpage(nStyle)

-- returns the URL of current page (relative to web root)

- nStyle:
 - 0=only name and port
 - 1=full address but dropping port
 - 2=full address (protocol, server, port)

computername()

-- return the computer name as seen by the OS (if this is supported by the web server in use)

folderBar(cHere,cClass,cTarget)

-- returns a string containing navigation links that take to current folder

ip(nParm,clp)

returns a string representing the IP address of the remote client as seen by server

- p1 (N) – output style for the IP detected
 - none or 0 = formatted like in 127.0.0.1
 - 1 = formatted like in 127.0.0.1:80 (with port in use)
 - 2 = formatted like in 127.000.000.001
 - 3 = formatted like in 127.000.000.001:80 (with port in use)
 - 4 = formatted like in 127_000_000_001

DBFree Library Functions – free.lib 2017

- 5 = calculated as string like in 2130706433
- 6 = calculated as string like in 2130706433:80 (with port in use)
- 7 = calculated as numeric like in 2130706433

p2 (C) – a different IP than the one detected (to be formatted as set by p1)

ip2num(cStr)

– accepts an IP as string and return its calculated number as numeric IP.

isMob(cUserAgentStr)

-- returns false or true if the string passed contains one of the names corresponding to an user agent/browser Usage: isMob(xNavUserAgent)

isWeb()

-- return true if the code is running through a web server

jreload(cLink)

Forces the web page to load a new URL if no parameter is passed reloads the same page

jsetId(cElement, cProperty, cValue)

-- find (by ID) the cElement in the page and set its cProperty to cValue

listWebVars(aArr, nStyle)

-- list all webVars stored in the given array (usually aVars)

- nStyle
 - 0 -- returns the array as string delimited with asterisks (default)
 - 1 -- returns a string to be used with pair parsers (square brackets)
 - 3 -- same as 2 but using braces and colons

num2ip(n)

– accepts a numeric with an IP number and returns the corresponding IP string

passVars(aArr)

(Superseded) – used by some older parts for passing the webVars stack inside web forms

passFields(aArr)

(Superseded) – transforms an array of fields into hidden fields to put in a form

protocol(cUrl)

Returns the protocol used by the given URL - if the URL does not contains a valid protocol specification return an empty string

DBFree Library Functions – free.lib 2017

scriptName()

(Superseded) – to be used with older web servers like Apache 1.x and 2.x

serveraddr()

Return the IP address of the server hosting DBFree

servername()

-- return the server name as seen by the client (in effect the URL used for the connection as seen by the DNS)

serverport()

-- returns the current port actually used for the connection Nov