

● sys モジュール述語

sysモジュールはシステムに組み込まれた標準のライブラリモジュールの集合です。
呼び出すときには、“::sys”に続けて記述します。

::sys <args 変数>

変数に、デカルト言語を起動したときの引数を設定します。

::sys <DLIBPATH 変数>

デカルト言語の使用するパスDLIBPATHを変数に表示します。

::sys <mkpred 引数>

引数を述語に変換します。

::sys <writeln リスト>

リストを出力した後、改行します。

::sys <write リスト>

::sys <w リスト>

リストを出力します。

::sys <wnl>

改行を出力します。

::sys <wo 数値>

数値を8進数で出力します。

::sys <wx 数値>

数値を16進数で出力します。

::sys <wf 数値>

数値を浮動小数点数として出力します。

::sys <wg 数値>

数値を浮動小数点数と最適な形式で出力します。

::sys <wtab>

タブを出力します。

::sys <fr 変数 文字列 幅>

文字列を幅で右寄せした結果を変数に設定します。

```
::sys <fl 変数 文字列 幅>
```

文字列を幅で左寄せした結果を変数に設定します。

```
::sys <isNil 引数>  
::sys <isAtom 引数>  
::sys <isList 引数>  
::sys <isPred 引数>  
::sys <isVar 引数>  
::sys <isUndefVar 引数>  
::sys <isFloat 引数>  
::sys <isInteger 引数>
```

引数を判定して該当すればtrueを返します。該当しなければunknownを返します。

```
::sys <isTrue 述語>  
::sys <isFalse 述語>  
::sys <isUnknown 述語>
```

引数の述語の結果を判定し、該当すればtrueを返します。該当しなければunknownを返します。

```
::sys <max 変数 リスト>
```

リストの中の要素の最大の整数値を変数に設定する。

```
::sys <min 変数 リスト>
```

リストの中の要素の最小の整数値を変数に設定する。

```
::sys <maxf 変数 リスト>
```

リストの中の要素の最大の浮動小数点値を変数に設定する。

```
::sys <minf 変数 リスト>
```

リストの中の要素の最小の浮動小数点値を変数に設定する。

```
::sys <regex 正規表現パターン 文字列 前文字列 マッチ文字列 後文字列>
```

文字列に正規表現パターンを適用した結果をマッチ文字列に設定し、前後の文字列を前文字列と後文字列に設定します。
(Windows上では動作しません。)

```
::sys <sub 正規表現パターン 文字列 置換文字列 出力文字列>
```

文字列に正規表現パターンを適用して該当した部分を、置換文字列に置換えた結果を出力文字列に設定します。
置換えは最初の1回だけ行われます。
(Windows上では動作しません。)

```
::sys <gsub 正規表現パターン 文字列 置換文字列 出力文字列>
```

文字列に正規表現パターンを適用して該当した部分を、置換文字列に置換えた結果を出力文字列に設定します。置換えは文字列のすべての該当部分に行われます。(Windows上では動作しません。)

::sys <split 変数 文字列 [区切り文字]>

文字列を区切り文字で分けてリストにしたものを
変数に設定します。区切り文字が指定されていない
場合は、空白とタブで区切られます。

::sys <toupper 変数 文字列>

文字列を大文字にします。

::sys <tolower 変数 文字列>

文字列を小文字にします。

::sys <length 変数 リスト>

リストの長さを変数に設定します。

::sys <random 変数>

変数に乱数を設定します。

::sys <sin 変数 ラジアン>

::sys <cos 変数 ラジアン>

::sys <tan 変数 ラジアン>

三角関数

::sys <asin 変数 値>

::sys <acos 変数 値>

::sys <atan 変数 値>

逆三角関数

::sys <sinh 変数 ラジアン>

::sys <cosh 変数 ラジアン>

::sys <tanh 変数 ラジアン>

ハイパボリック三角関数

::sys <asinh 変数 値>

::sys <acosh 変数 値>

::sys <atanh 変数 値>

ハイパボリック逆三角関数

::sys <log 変数 値>

::sys <log10 変数 値>

::sys <exp 変数 値>

::sys <pow 変数 値1 値2>

対数関数

::sys <sqrt 変数 値>

平方根

::sys <abs 変数 値>

絶対値

::sys <int 変数 値>

整数値

::sys <floor 変数 値>

引き数を越えない最大の整数値

::sys <car 変数 値>

::sys <cdr 変数 値>

リストのcar, cdr

::sys <cons 変数 リスト1 リスト2>

リストの連結

::sys <code コード>

文字コードの設定。UTF8, EUCJP, SJISが指定できます。

::sys <char 変数 文字列>

文字列を文字ごとに分解してリストにして、変数に設定します。漢字のような多バイト文字も正しく一文字ごとに分解します。

::sys <byte 変数 文字列>

文字列をバイトごとに分解してリストにして変数に設定します。漢字のような多バイト文字もバイト単位に分解されます。

::sys <asciichar 変数 文字列>

::sys <utf8char 変数 文字列>

::sys <eucchar 変数 文字列>

::sys <sjischar 変数 文字列>

文字列を文字コードごとに分解してリストにして変数に設定します。

::sys <concat 変数 リスト>

文字のリストを合体させ文字列を合成して変数に設定します。

::sys <concatcode 変数 リスト>

文字コードのリストを合体させ文字列を合成して変数に設定します。

```
::sys <bitand 変数 数値1 数値2>  
::sys <bitor 変数 数値1 数値2>  
::sys <bitxor 変数 数値1 数値2>  
::sys <bitnot 変数 数値1>
```

bit演算

```
::sys <shiftrl 変数 数値 シフト数>  
::sys <shiftr 変数 数値 シフト数>
```

整数値のビットシフト。
shiftrlは左へ、shiftrは右へシフトさせます。

```
::sys <eq 引数1 引数2>  
::sys <noteq 引数1 引数2>  
::sys <is 引数1 引数2>
```

引数1と引数2の比較

```
::sys <getc 変数>
```

変数に 1 char入力します。

```
::sys <putc 文字>
```

1charを出力します。

```
::sys <getline 変数 [述語...]>
```

1行入力して変数に設定します。述語が設定されている場合は
変数に設定された文字列を入力として述語が実行されます。

```
::sys <syntax 文字列 述語...>
```

文字列を入力ファイルとして述語が実行されます。

```
::sys <tmpfile 変数>
```

テンポラリファイル名を変数に設定します。

```
::sys <openr ファイル名 述語...>
```

ファイル名のファイルを読み取り用にオープンして、
述語を実行します。

```
::sys <openw ファイル名 述語...>
```

ファイル名のファイルを書き込み用にオープンして、
述語を実行します。

```
::sys <openwp ファイル名 述語...>
```

ファイル名のファイルを追記書き込み用にオープン
して、述語を実行します。

::sys <gettime 変数>
現在の時刻をマイクロ秒単位で変数に設定します。

::sys <time 変数>
実行中の述語の (user時間, sys時間, elapsed時間) のリストを
変数に設定する。

::sys <date 変数>
日時を変数に設定します。

::sys <countnode 変数>
使用しているノード数を変数に設定します。

::sys <gc>
ガーベージコレクタを起動します。

以上。