NAME

ConversionsUtil

SYNOPSIS

```
use ConversionsUtil;

use ConversionsUtil qw(:math);

use ConversionsUtil qw(:all);
```

DESCRIPTION

ConversionsUtil module provides the following functions:

BinaryToDecimal, BinaryToHexadecimal, DecimalToBinary, DecimalToHexadecimal, DecimalToOctal, DegreesToRadians, HexadecimalToBinary, HexadecimalToDecimal, HexadecimalToOctal, OctalToDecimal, OctalToHexadecimal, RadiansToDegrees, StringToBinary, StringToHexadecimal

FUNCTIONS

BinaryToDecimal

```
$Decimal = BinaryToDecimal($Binary);
```

Converts a *Binary* string to Decimal string.

BinaryToHexadecimal

```
$Hexadecimal = BinaryToHexadecimal($Binary);
```

Converts a *Binary* string to Hexadecimal string.

DecimalToBinary

```
$Binary = DecimalToBinary($Decimal);
```

Converts a *Decimal* string to Binary string.

DecimalToHexadecimal

```
$Hexadecimal = DecimalToHexadecimal($Decimal);
```

Converts a *Decimal* string to Hexadecimal string.

DecimalToOctal

```
$Octal = DecimalToOctal($Decimal);
```

Converts a *Decimal* string to Octal string.

DegreesToRadians

```
$Radians = DegreesToRadians($Degrees, [$DoNotWrapValue]);
```

Converts degrees to radians in the range from 0 to 2PI or to corresponding radians without wrapping the converted value to 0 to 2PI. Default is to wrap the converted value.

HexadecimalToBinary

```
$Binary = HexadecimalToBinary($Hexadecimal);
```

Converts a *Hexadecimal* string to Binary string.

HexadecimalToDecimal

```
$Decimal = HexadecimalToDecimal($Hexadecimal);
```

Converts a *Hexadecimal* string to Decimal string.

HexadecimalToOctal

```
$Octal = HexadecimalToOctal($Hexadecimal);
```

Converts a *Hexadecimal* string to Octal string.

OctalToDecimal

```
$Decimal = OctalToDecimal($Octal);
```

Converts a *Octal* string to Decimal string.

OctalToHexadecimal

```
$Hexadecimal = OctalToHexadecimal($Octal);
```

Converts a *Octal* string to Hexadecimal string.

RadiansToDegrees

```
$Degrees = RadiansToDegrees($Radians, [$DoNotWrapValue]);
```

Converts radians to degrees in the range from 0 to 360 or to corresponding degrees without wrapping the converted value to 0 to 360. Default is to wrap the converted value.

StringToBinary

```
$BinaryString = StringToBinary($String, [$UseReverseBitOrder]);
```

Converts specified *String* into a Binarystring. Going from left to right, two ways of arranging bits inside each byte are available: Most Significat Bits (MSB) first or Least Significat Bits (LSB) first. Default is MSB corresponding to descending bits order (PerlSpeak) inside each each packed byte (Most singificat bits first).

StringToHexadecimal

```
$HexadecimalString = StringToHexadecimal($String,
                     [$UseReverseBitOrder]);
```

Convert string into a hexadecimal string. Two ways of arranging nybbles (pair of 4 bits in each byte) are available: high nybbles first or low nybbles first. Default is MSB corresponding to high nybbles (PerlSpeak) first. Low and high nybbles correspond to pair of a low and high four bits in a byte.

AUTHOR

Manish Sud <msud@san.rr.com>

SEE ALSO

Constants.pm, MathUtil.pm

COPYRIGHT