

Unicode

Introduction

Unicode Support.

Warning
This extension is still in development and it isn't available to public yet.

Installing/Configuring

Requirements

» [ICU](#) 3.4 or later is required.

Installation

First you should download and install ICU:

Example #1 - install ICU on Unix

```
./configure --disable-threads --enable-extras --enable-icuio --enable-layout  
make && make install
```

Then checkout latest PHP and configure it `--with-icu-dir=<dir>` option, where `<dir>` was the dir to where you installed ICU. You don't need to explicitly use this option if you install ICU to a standard location.

Runtime Configuration

The behaviour of these functions is affected by settings in *php.ini*.

Unicode Configuration Options

Name	Default	Changeable	Changelog
unicode.fallback_encoding	NULL	PHP_INI_ALL	Available since PHP 6.0.0.
unicode.from_error_mode	"2"	PHP_INI_ALL	Available since PHP 6.0.0.
unicode.from_error_subst_char	"3f"	PHP_INI_ALL	Available since PHP 6.0.0.
unicode.http_input_encoding	NULL	PHP_INI_ALL	Available since PHP 6.0.0.
unicode.output_encoding	NULL	PHP_INI_ALL	Available since PHP 6.0.0.

unicode.runtime_encoding	NULL	PHP_INI_ALL	Available since PHP 6.0.0.
unicode.script_encoding	NULL	PHP_INI_ALL	Available since PHP 6.0.0.

For further details and definitions of the PHP_INI_* constants, see the [php.ini directives](#).

Here's a short explanation of the configuration directives.

`unicode.output_encoding` [string](#)
Default encoding for output.

Resource Types

This extension has no resource types defined.

Predefined Constants

The constants below are defined by this extension, and will only be available when the extension has either been compiled into PHP or dynamically loaded at runtime.

constant	value	description
U_INVALID_STOP	0	stop at first invalid character
U_INVALID_SKIP	1	skip invalid characters
U_INVALID_SUBSTITUTE	2	replace invalid characters
U_INVALID_ESCAPE	3	escape invalid characters

Unicode Functions

unicode_decode

unicode_decode -- Convert a binary string into a Unicode string

Description

unicode **unicode_decode** (string \$input, string \$encoding [, int \$errmode])

Convert a binary string encoded in *encoding* to a **unicode** string.

Parameters

input

Try **string** that is converted.

encoding

The encoding of *input*.

errmode

Conversion error mode. This parameter determines the action to take when the converter cannot convert a character. For a list of available modes, refer to [unicode_set_error_mode\(\)](#). If the parameter is not set, the global error mode is used.

Return Values

A **unicode** string or **FALSE** on failure.

Errors/Exceptions

Emits a **E_WARNING** level error if a converter cannot be created for the desired *encoding*.

Notes

Warning
This function is <i>EXPERIMENTAL</i> . The behaviour of this function, its name, and surrounding documentation may change without notice in a future release of PHP. This function should be used at your own risk.

See Also

- [unicode_set_error_mode\(\)](#)
- **unicode_semantics()**
- [unicode_encode\(\)](#)

unicode_encode

unicode_encode -- Convert a unicode string in any encoding

Description

string **unicode_encode** ([unicode](#) \$input, string \$encoding [, int \$errmode])

Takes a [unicode](#) string and converts it to a string in the specified *encoding*.

Parameters

input

The [unicode](#) string that is converted.

encoding

The new encoding for *input*.

errmode

Conversion error mode. This parameter determines the action to take when the converter cannot convert a character. For a list of available modes, refer to [unicode_set_error_mode\(\)](#). If the parameter is not set, the global error mode is used.

Return Values

A [string](#) on success, or **FALSE** on failure.

Errors/Exceptions

Emits a **E_WARNING** level error if a converter cannot be created for the desired *encoding*.

Examples

Example #2 - A [unicode_encode\(\)](#) example

Note: The characters will be seen instead of entities in the output.

```
<?php
header ( 'Content-Type: text/plain; charset=ISO-8859-2' );

$encoded = unicode_encode ( '\u0150\u0179', 'ISO-8859-2' );

echo 'Unicode semantics: ', ini_get ( 'unicode_semantics' ), PHP_EOL;
echo 'The string itself:', $encoded, PHP_EOL;
```

```
echo 'The length of the string: ', strlen ($encoded);  
?>
```

The above example will output something similar to:

```
Unicode semantics: 1  
The string itself: &#336;&#377;  
The length of the string: 2
```

Notes

Warning

This function is *EXPERIMENTAL*. The behaviour of this function, its name, and surrounding documentation may change without notice in a future release of PHP. This function should be used at your own risk.

See Also

- [unicode_set_error_mode\(\)](#)
- [unicode_semantics\(\)](#)
- [unicode_decode\(\)](#)

unicode_get_error_mode

unicode_get_error_mode -- Get the error mode for strings conversions

Description

```
int unicode_get_error_mode ( int $direction )
```

Return the current error mode for string conversions in *direction*.

Parameters

direction

Get the error mode for conversion *direction*. This can either be **FROM_UNICODE** or **TO_UNICODE**. See [unicode_set_error_mode\(\)](#) for more details.

Return Values

This function returns one of the error mode constants or **FALSE** on failure.

Errors/Exceptions

Emits a **E_WARNING** level error if *direction* is invalid.

Notes

Warning
This function is <i>EXPERIMENTAL</i> . The behaviour of this function, its name, and surrounding documentation may change without notice in a future release of PHP. This function should be used at your own risk.

See Also

- [unicode_set_error_mode\(\)](#)
- [unicode_encode\(\)](#)
- [unicode_decode\(\)](#)

unicode_get_subst_char

unicode_get_subst_char -- Get the substitution character for string conversion errors

Description

[unicode](#) `unicode_get_subst_char` (void)

Return the substitution character for string conversion errors set by [unicode_set_subst_char\(\)](#).

Return Values

The [unicode](#) substitution character is returned.

Notes

Warning
This function is <i>EXPERIMENTAL</i> . The behaviour of this function, its name, and surrounding documentation may change without notice in a future release of PHP. This function should be used at your own risk.

See Also

- [unicode_set_error_mode\(\)](#)
- [unicode_set_subst_char\(\)](#)

unicode_set_error_mode

unicode_set_error_mode -- Set the error mode for strings conversions

Description

bool **unicode_set_error_mode** (int \$direction, int \$mode)

This function sets the error mode for string conversions between different encodings. An error can occur during conversion when an illegal character is encountered or if a character cannot be represented into the new encoding. By default, when an error is encountered, the conversion stops.

Parameters

direction

The *direction* sets the conversion direction to which the error mode will apply. This can either be **FROM_UNICODE**, which will set the error mode for conversions from a [unicode](#) string to a binary string or **TO_UNICODE**, which will set the error mode for conversions from a binary string to a [unicode](#) string.

mode

mode determines how the conversion errors are handled. It should be one of the following constants:

Available modes

Mode	Description
U_CONV_ERROR_STOP	Stop the conversion. This is the default mode.
U_CONV_ERROR_SKIP	Skip the character.
U_CONV_ERROR_SUBST	Substitute the character. The substituting character can be set with unicode_set_subst_char() .
U_CONV_ERROR_ESCAPE_UNICODE	Escape the faulty bytes and represent them in the Unicode character format.
U_CONV_ERROR_ESCAPE_ICU	Escape the faulty bytes and represent them in the ICU character format.
U_CONV_ERROR_ESCAPE_JAVA	Escape the faulty bytes and print them in the Java character format.
U_CONV_ERROR_ESCAPE_XML_DEC	Escape the faulty bytes and represent them

	in decimal format.
U_CONV_ERROR_ESCAPE_XML_HEX	Escape the faulty bytes and represent them in hexadecimal format.

Return Values

Returns **TRUE** on success or **FALSE** on failure.

Errors/Exceptions

Emits a **E_WARNING** level error if the direction or the mode are invalid.

Notes

Warning
This function is <i>EXPERIMENTAL</i> . The behaviour of this function, its name, and surrounding documentation may change without notice in a future release of PHP. This function should be used at your own risk.

See Also

- [unicode_encode\(\)](#)
- [unicode_decode\(\)](#)

unicode_set_subst_char

unicode_set_subst_char -- Set the substitution character for string conversion errors

Description

bool **unicode_set_subst_char** ([unicode](#) \$character)

When converting a string from or to unicode, an invalid or unconvertable character may be encountered. When the conversion error mode is set to **U_CONV_ERROR_SUBST**, the character that caused the error is substituted by a character set by this function. The default substitution character is `?`.

Parameters

character

The substitution character to use.

Return Values

Returns **TRUE** on success or **FALSE** on failure.

Errors/Exceptions

Emits a **E_WARNING** level error if *character* is invalid.

Notes

Warning
This function is <i>EXPERIMENTAL</i> . The behaviour of this function, its name, and surrounding documentation may change without notice in a future release of PHP. This function should be used at your own risk.

See Also

- [unicode_set_error_mode\(\)](#)
- [unicode_get_error_mode\(\)](#)
- [unicode_encode\(\)](#)
- [unicode_decode\(\)](#)