

File Information

Introduction

The functions in this module try to guess the content type and encoding of a file by looking for certain *magic* byte sequences at specific positions within the file. While this is not a bullet proof approach the heuristics used do a very good job.

Installing/Configuring

Requirements

magic_open library is needed to build this extension.

Installation

Information for installing this PECL extension may be found in the manual chapter titled [Installation of PECL extensions](#). Additional information such as new releases, downloads, source files, maintainer information, and a CHANGELOG, can be located here: » <http://pecl.php.net/package/fileinfo>

Runtime Configuration

This extension has no configuration directives defined in *php.ini*.

Resource Types

There is one resource used in Fileinfo extension: a magic database descriptor returned by [finfo_open\(\)](#).

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.

FILEINFO_NONE ([integer](#))

No special handling.

FILEINFO_SYMLINK ([integer](#))

Follow symlinks.

FILEINFO_MIME ([integer](#))

Return a mime string, instead of a textual description.

FILEINFO_COMPRESS ([integer](#))

Decompress compressed files.

FILEINFO_DEVICES ([integer](#))

Look at the contents of blocks or character special devices.

FILEINFO_CONTINUE ([integer](#))

Return all matches, not just the first.

FILEINFO_PRESERVE_ETIME ([integer](#))

If possible preserve the original access time.

FILEINFO_RAW ([integer](#))

Don't translate unprintable characters to a \ooo octal representation.

Fileinfo Functions

finfo_buffer

finfo_buffer -- Return information about a string buffer

Description

```
string finfo_buffer ( resource $finfo, string $string [, int $options [, resource $context ] ] )
```

finfo

```
string buffer ( string $string [, int $options [, resource $context ] ] )
```

This function is used to get information about binary data in a string.

Parameters

finfo

Fileinfo resource returned by [finfo_open\(\)](#).

string

Content of a file to be checked.

options

One or disjunction of more [Fileinfo constants](#).

context

Return Values

Returns a textual description of the *string* argument, or **FALSE** if an error occurred.

Examples

Example #1 - A finfo_buffer() example
--

<pre><?php \$finfo = new finfo(FILEINFO_MIME); echo \$finfo->buffer(\$_POST["script"]) . "\n"; ?></pre>
--

The above example will output something similar to:

```
application/x-sh
```

See Also

- [finfo_file\(\)](#)

finfo_close

finfo_close -- Close fileinfo resource

Description

bool **finfo_close** (resource \$finfo)

finfo

__destruct (void)

This function closes the resource opened by [finfo_open\(\)](#).

Parameters

finfo

Fileinfo resource returned by [finfo_open\(\)](#).

Return Values

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

finfo_file

finfo_file -- Return information about a file

Description

```
string finfo_file ( resource $finfo, string $file_name [, int $options [, resource $context ] ] )
```

finfo

```
string file ( string $file_name [, int $options [, resource $context ] ] )
```

This function is used to get information about a file.

Parameters

finfo

Fileinfo resource returned by [finfo_open\(\)](#).

file_name

Name of a file to be checked.

options

One or disjunction of more [Fileinfo constants](#).

context

For a description of *contexts*, refer to [Stream Functions](#).

Return Values

Returns a textual description of the contents of the *filename* argument, or **FALSE** if an error occurred.

Examples

Example #2 - A finfo_file() example
--

<pre><?php \$finfo = finfo_open(FILEINFO_MIME); // return mime type ala mimetype extension</pre>

```
foreach (glob("*") as $filename) {  
    echo finfo_file($finfo, $filename) . "\n";  
}  
finfo_close($finfo);  
?>
```

The above example will output something similar to:

```
text/html  
image/gif  
application/vnd.ms-excel
```

See Also

- [finfo_buffer\(\)](#)

finfo_open

finfo->__construct()

finfo_open -- finfo->__construct() -- Create a new fileinfo resource

Description

Procedural style

resource **finfo_open** ([int *\$options* [, string *\$arg*]])

Object oriented style (constructor):

finfo

__construct ([int *\$options* [, string *\$magic_file*]])

This function opens a magic database and returns its resource.

Parameters

options

One or disjunction of more [Fileinfo constants](#).

magic_file

Name of a magic database file, usually something like */path/to/magic.mime*. If not specified, the *MAGIC* environment variable is used. If this variable is not set either, */usr/share/misc/magic* is used by default. A *.mime* and/or *.mgc* suffix is added if needed.

Return Values

Returns a magic database resource on success or **FALSE** on failure.

Examples

Example #3 - Object oriented style

<pre><?php \$info = new finfo(FILEINFO_MIME, "/usr/share/misc/magic"); // return mime type ala mimetype extension</pre>
--

```
if (!$finfo) {
    echo "Opening fileinfo database failed";
    exit();
}

/* get mime-type for a specific file */
$filename = "/usr/local/something.txt";
echo $finfo->file($filename);

/* close connection */
$finfo->close();
?>
```

Example #4 - Procedural style

```
<?php
$finfo = finfo_open(FILEINFO_MIME, "/usr/share/misc/magic"); // return mime
type ala mimetype extension

if (!$finfo) {
    echo "Opening fileinfo database failed";
    exit();
}

/* get mime-type for a specific file */
$filename = "/usr/local/something.txt";
echo finfo_file($finfo, $filename);

/* close connection */
finfo_close($finfo);
?>
```

The above example will output:

text/plain

See Also

- [finfo_close\(\)](#)

finfo_set_flags

finfo_set_flags -- Set libmagic configuration options

Description

bool **finfo_set_flags** (resource \$finfo, int \$options)

finfo

bool **set_flags** (int \$options)

This function sets various Fileinfo options. Options can be set also directly in [finfo_open\(\)](#) or other Fileinfo functions.

Parameters

finfo

Fileinfo resource returned by [finfo_open\(\)](#).

options

One or disjunction of more [Fileinfo constants](#).

Return Values

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