

Class

Symfony\Component\HttpFoundation\File\UploadedFile

class **UploadedFile** extends [File](#)

A file uploaded through a form.

Methods

__construct(string \$path, string \$originalName, string \$mimeType = null, int \$size = null, int \$error = null, bool \$test = false)
Accepts the information of the uploaded file as provided by the PHP global \$_FILES.

string|null [guessExtension](#)() from [File](#)
Returns the extension based on the mime type.

string|null [getMimeType](#)() from [File](#)
Returns the mime type of the file.

string [getExtension](#)() from [File](#)
Returns the extension of the file.

[File](#) [move](#)(string \$directory, string \$name = null)
Moves the file to a new location.

string|null [getClientOriginalName](#)()
Returns the original file name.

string [getClientOriginalExtension](#)()
Returns the original file extension

string|null [getClientMimeType](#)()
Returns the file mime type.

string|null [guessClientExtension](#)()
Returns the extension based on the client mime type.

int|null [getClientSize](#)()
Returns the file size.

int [getError](#)()
Returns the upload error.

bool [isValid](#)()
Returns whether the file was uploaded successfully.

static int [getMaxFileSize](#)()
Returns the maximum size of an uploaded file as configured in php.ini

string [getErrorMessage](#)()
Returns an informative upload error message.

Details

public **__construct**(string \$path, string \$originalName, string \$mimeType = null, int \$size = null, int \$error = null, bool \$test = false) *at line 92*

Accepts the information of the uploaded file as provided by the PHP global \$_FILES.

The file object is only created when the uploaded file is valid (i.e. when the [isValid\(\)](#) method returns true). Otherwise the only methods that could be called on an [UploadedFile](#) instance are:

- [getClientOriginalName](#),
- [getClientMimeType](#),
- [isValid](#),
- [getError](#).

Calling any other method on a non-valid instance will cause an unpredictable result.

Parameters

string \$path	The full temporary path to the file
string \$originalName	The original file name
string \$mimeType	The type of the file as provided by PHP

int	\$size	The file size
int	\$error	The error constant of the upload (one of PHP's <code>UPLOADERRXXX</code> constants)
bool	\$test	Whether the test mode is active

Exceptions

<code>FileException</code>	If <code>file_uploads</code> is disabled
<code>FileNotFoundException</code>	If the file does not exist

```
public string|null guessExtension()
```

in [File](#) at line 62

Returns the extension based on the mime type.

If the mime type is unknown, returns null.

This method uses the mime type as guessed by `getMimeType()` to guess the file extension.

Return Value

string|null The guessed extension or null if it cannot be guessed

See also

[getMimeType\(\)](#)

```
public string|null getMimeType()
```

in [File](#) at line 83

Returns the mime type of the file.

The mime type is guessed using a `MimeTypeGuesser` instance, which uses `finfo()`, `mimecontenttype()` and the system binary "file" (in this order), depending on which of those are available.

Return Value

string|null The guessed mime type (i.e. "application/pdf")

See also

[MimeTypeGuesser](#)

```
public string getExtension()
```

in [File](#) at line 99

Returns the extension of the file.

`\SplFileInfo::getExtension()` is not available before PHP 5.3.6

Return Value

string The extension

```
public File move(string $directory, string $name = null)
```

at line 232

Moves the file to a new location.

Parameters

string	\$directory	The destination folder
string	\$name	The new file name

Return Value

[File](#) A File object representing the new file

Exceptions

`FileException` if, for any reason, the file could not have been moved

```
public string|null getClientOriginalName()
```

at line 113

Returns the original file name.

It is extracted from the request from which the file has been uploaded. Then it should not be considered as a safe value.

Return Value

string|null The original name

```
public string getClientOriginalExtension()
```

at line 126

Returns the original file extension

It is extracted from the original file name that was uploaded. Then it should not be considered as a safe value.

Return Value

string The extension

```
public string|null getClientMimeType()
```

at line 146

Returns the file mime type.

The client mime type is extracted from the request from which the file was uploaded, so it should not be considered as a safe value.

For a trusted mime type, use `getMimeType()` instead (which guesses the mime type based on the file content).

Return Value

string|null The mime type

See also

[getMimeType](#)

```
public string|null guessClientExtension()
```

at line 168

Returns the extension based on the client mime type.

If the mime type is unknown, returns null.

This method uses the mime type as guessed by `getClientMimeType()` to guess the file extension. As such, the extension returned by this method cannot be trusted.

For a trusted extension, use `guessExtension()` instead (which guesses the extension based on the guessed mime type for the file).

Return Value

string|null The guessed extension or null if it cannot be guessed

See also

[guessExtension\(\)](#)

[getClientMimeType\(\)](#)

```
public int|null getClientSize()
```

at line 186

Returns the file size.

It is extracted from the request from which the file has been uploaded. Then it should not be considered as a safe value.

Return Value

int|null The file size

```
public int getError()
```

at line 201

Returns the upload error.

If the upload was successful, the constant `UPLOADERROK` is returned. Otherwise one of the other `UPLOADERRXXX` constants is returned.

Return Value

int The upload error

public bool **isValid()**

at line 213

Returns whether the file was uploaded successfully.

Return Value

bool True if the file has been uploaded with HTTP and no error occurred.

static public int **getMaxFileSize()**

at line 259

Returns the maximum size of an uploaded file as configured in php.ini

Return Value

int The maximum size of an uploaded file in bytes

public string **getErrorMessage()**

at line 291

Returns an informative upload error message.

Return Value

string The error message regarding the specified error code

Generated by [Sami](#), the API Documentation Generator.