ImdbScraper (class)

Updated: 2011-11-22

About

Scrapes information about MOVIEmeter top-10 movies of the week from imdb.com.

Dependencies

Scraper has been tested and should work on PHP versions 5.2 and 5.3.

DOMDocument extension.
http://se2.php.net/manual/en/intro.dom.php

cURL must be included in your PHP-installation.
http://se2.php.net/manual/en/curl.installation.php

Install

Upload the files to wished location and enjoy!

Usage

Simply include the *ImdbScraper.php*-file and instantiate the class and call the *getTop10Data()*-function to retrieve information about the “MOVIEmeter weekly top 10” movies found on imdb.com.

Settings

chartUrl: the imdb webpage to scrape, where the top 10 list is found.
imdbUrl: url to imdb site root without trailing slash.
cacheFile: where to save and locate the scraped cache file.
cacheTime: time in seconds to load cache before scraping fresh data.

Caching

The scraper will store scraped data in a text file on the web server. Default time to cache before fetching fresh data is 24 hours. Default location to store cache file is “cache/imdb_cache.txt” relative to where the ImdbScraper.php is located. The folder must exist initially before using the scraper, but the cache-file will be created if not found.

Methods

public *getTop10Data()*

Extracts movie title with year and link to the movie on imdb for each movie in the weekly top 10 list. This is the method you want to call to retrieve the toplist.

Returns a two dimensional array holding information about each movie (utf8 decoded).

Array elements

title: The title and year of the movie.
link: The relative url to the movies imdb-page.

Example:
$imdbScraper->getTop10Data() will return something like:

```
array( 0 => array( "title" => "Immortals
(2011)", "link" => "title/tt1253864/" ), 1 =>
array( "title" => "11-11-11 (2011)", "link" => "title/
tt1712159/" ), 2 => ... )
```

private *scrapeImdbToplist()*

Scrapes the Imdb MOVIEmeter top 10 movies of the week toplist.

Returns information, as an array, about the scraped document and the scraped data itself.

Array elements

error: boolean, false if no error during scraping.
status: http-response code (200 if ok).
content: string, the scraped document source code.

private *getDocumentSource()*

Gets the scraped data from the cache if one exists and has not expired. Otherwise a fresh scraping will be performed from calling the scrapeImdbToplist-function, which will be cached.

Returns the scraped document source code as a string.