

Get data from string with JSON object

This tutorial shows you **how to get data from a string that contains an JSON object**, and use it in PHP, with `json_decode()`.

That object (the JSON string) can be retrieved from an external resource. For example, from an URL address (with `file_get_contents()`, or cURL) that outputs a string with data in valid JSON object, or from a text file on server, or from database.

Let's work with a simple example; a JSON object saved into a text file on server.

The file, named "sitedata.txt" contains this string, a simple JSON object with 3 properties (url, category, nr_pages):

```
{"url": "http://coursesweb.net/", "category": "Web Development", "nr_pages": 400}
```

We use `file_get_contents()` to get data (the string) from text file, than, with `json_decode()` converts the string into a PHP Object.

```
<?php
// From: http://coursesweb.net/php-mysql/
$file = 'sitedata.txt'; // path and name of the file

// check if the file exists
if(file_exists($file)) {
    $filedata = file_get_contents($file);

    /* Or, you can add directly the string:
    $filedata = '{"url": "http://coursesweb.net/", "category": "Web
Development", "nr_pages": 400}';
    */

    // converts the string into an Object of the stdClass class
    $objjson = json_decode($filedata, true);

    // traverse the object, and outputs each property and its value
    foreach($objjson AS $prop => $val) {
        echo '<br/>'. $prop .' - '. $val;
    }
}
else echo $file .' not exists';
```

Outputs:

```
url - http://coursesweb.net/
category - Web Development
nr_pages - 400
```

Getting data from PHP object

- The `json_decode()` converts the JSON string into an Object of the `stdClass`. To get the value associated to a certain property, use this syntax:

```
$value = $object->property_name;
```

Or, if the "property_name" is stored into a variable, use this syntax:

```
$property = 'property_name';
$value = $object->{$property};
```

Here's another example;

```

<?php
// string with JSON object
$strjson = '{"url": "http://coursesweb.net/", "category": "Web Development",
"nr_pages": 400}';

// converts the string into an Object of the stdClass class
$objjson = json_decode($strjson);

// getting the value of a certain property
$url = $objjson->url;
echo '<br/>'. $url;

// getting the value of a property with the name stored in variable
$prop = 'category';
$val = $objjson->{$prop};
echo '<br/>'. $val;

```

This code will output:

<http://coursesweb.net/>
Web Development

It is important that the string to be a valid JSON object /array, otherwise, the `json_decode()` function will return False or Null.

Converting JSON object into an Associative Array

If you want to convert the string with JSON object into an associative Array in PHP, add True as the second argument in `json_decode()`.

Syntax:

```
$arr = json_decode($json_string, true);
```

Example:

```

<?php
// string with JSON object
$strjson = '{"url": "http://www.marplo.net/", "category": "Diverse",
"nr_pages": 1500}';

// converts the string into an Associative array
$arrjson = json_decode($strjson, true);

// Test
var_dump($arrjson);

// getting the value of a certain element
$url = $arrjson['url'];
echo '<br/>'. $url;

// getting the value of an element with the name stored in variable
$elm = 'category';
$val = $arrjson[$elm];
echo '<br/>'. $val;

```

The above example will output:

array(3) { ["url"]=> string(18) "http://marplo.net/" ["category"]=> string(7) "Diverse" ["nr_pages"]=> int(1500) }
<http://www.marplo.net/>
Diverse

- The JSON string can contain more complex data, with a structure that has other objects and arrays. In this case you must know how to work with arrays in PHP. Here's an example:

```
<?php
// string with JSON object, containing another Object, and an Array
$strjson = '{
  "url": "http://www.marplo.net/",
  "category": {"ct1": "Web Development", "ct2": "Web Programming"},
  "courses": ["PHP-MySQL", "JavaScript", "HTML", "CSS"]
}';

// converts the string into an Associative array
$arrjson = json_decode($strjson, true);

// Outputs array structure
echo '<pre>';
var_export($arrjson);
echo '</pre>';

// gets first category "ct1", and the second element in "courses"
$ct1 = $arrjson['category']['ct1'];
$course2 = $arrjson['courses'][1];

echo $ct1 . '<br/>' . $course2;
```

The above example will output:

```
array (
  'url' => 'http://www.marplo.net/',
  'category' =>
  array (
    'ct1' => 'Web Development',
    'ct2' => 'Web Programming',
  ),
  'courses' =>
  array (
    0 => 'PHP-MySQL',
    1 => 'JavaScript',
    2 => 'HTML',
    3 => 'CSS',
  )
)

Web Development
JavaScript
```

- The JSON format is used especially in Development Applications, like Facebook App, Google Apps, Youtube Api. If the string with data from that Appi isn't a valid JSON format, you must process the string to get a valid JSON before using the json_decode(). For example, the "info" from Youtube Api contains the JSON object with data in this string (which is not a valid JSON format):

```
info = { "title" : "DaOne - povestea noastra (feat. Ewa)", "image" :
"http://i.ytimg.com/vi/6UjfTKHQ4vo/default.jpg", "length" : "4", "status" : "serving",
"progress_speed" : "", "progress" : "", "ads" : "", "pf" :
"http://acef.u.aclst.com/ping.php?video_id=6UjfTKHQ4vo&h=2717", "h" :
```

```
"a148abf7ef86bab307d50d0d9d6177f0" };
```

Here's how to get data from this string.

```
<?php
// From: http://coursesweb.net/php-mysql/
// string with info from Youtube App
$yinfo = 'info = {
  "title" : "DaOne - povestea noastra (feat. Ewa)",
  "image" : "http://i.ytimg.com/vi/6UjfTKHQ4vo/default.jpg",
  "length" : "4",
  "status" : "serving",
  "progress_speed" : "",
  "progress" : "",
  "ads" : "",
  "pf" : "http://acef.u.aclst.com/ping.php?video_id=6UjfTKHQ4vo&h=2717",
  "h" : "a148abf7ef86bab307d50d0d9d6177f0"
}';

/*
  You can get the string from Youtube Api using this code:
  $url = 'http://www.youtube-
mp3.org/api/itemInfo/?video_id=6UjfTKHQ4vo&adloc=&r={{datetime}}';
  $yinfo = file_get_contents($url);
*/

// gets only the substring with valid JSON
if(preg_match('/{(.*)}/i', $yinfo, $infojson)) {
  // converts the valid JSON string into an Associative array
  $arrjson = json_decode($infojson[0], true);

  // gets and outputs title and image data
  $title = $arrjson['title'];
  $imgsrc = $arrjson['image'];

  echo $title . '<br/>'. $imgsrc;
}
```