

PETICION GET A UN SERVICIO O API

Para realizar una petición de tipo get a un servicio o una api, desde php se utiliza **file_get_contents**

file_get_contents(\$url): La url es la dirección a el recurso sobre el que realizamos una petición get, habitualmente esta url incluye la información con la que solicitamos la información, y esta información debe de tener un formato correcto (por ejemplo los espacios se sustituyen por un %20). Para asegurarnos de que la url está bien codificada se utiliza la función **urlencode()**

Supongamos que nuestro recurso es una API cuya URL es: http://ejemplo.es?nombre=paco%20lopez. A Esta api le enviamos un nombre de persona y nos devuelve la información de esa persona. Pero si ese nombre se recoge de un formulario habrá que asegurarse de que los espacios y los caracteres especiales se gestionan correctamente por tanto se construirá la url así:

```
$url="http://ejemplo.es?nombre=".urlencode($_POST['nombre']);
```

USO DE JSON

Las peticiones a servicios o a APIs devuelven los resultados según estos estén programados, en la actualidad el formato más usado es JSON, este formato tiene una estructura muy definida y php tiene instrucciones para codificarlo y decodificarlo

json_encode(\$value)

Devuelve un string con la representación JSON de \$value.

```
<?php
$arr = array('a' => 1, 'b' => 2, 'c' => 3, 'd' => 4, 'e' =>
5);

echo json_encode($arr);
?>
```

json_decode(\$variable,\$assoc)

Convierte un string codificado en JSON a una variable de PHP.

\$assoc: Cuando es TRUE, los object devueltos serán convertidos a array asociativos. Si es FALSE o se omite se convierten a objetos.

```
<?php
$json = '{"a":1,"b":2,"c":3,"d":4,"e":5}';

var_dump(json_decode($json));
var_dump(json_decode($json, true));

?>
```

El resultado del ejemplo sería:

```
object(stdClass)#1 (5) {
```

```
    ["a"] => int(1)
    ["b"] => int(2)
    ["c"] => int(3)
    ["d"] => int(4)
    ["e"] => int(5)
}
```

```
array(5) {
    ["a"] => int(1)
    ["b"] => int(2)
    ["c"] => int(3)
    ["d"] => int(4)
    ["e"] => int(5)
}
```

En el ejemplo de la petición a una API visto anteriormente sería de la siguiente manera

```
$datos=json_decode(file_get_contents($url), true);
```

De esta manera `$datos` contendrá un array asociativo con toda la información devuelta, y generalmente con varias dimensiones.