

Objective = understand what is an input list of GPS points

In this tutorial you will learn

how to use an input list of GPS points with PHP

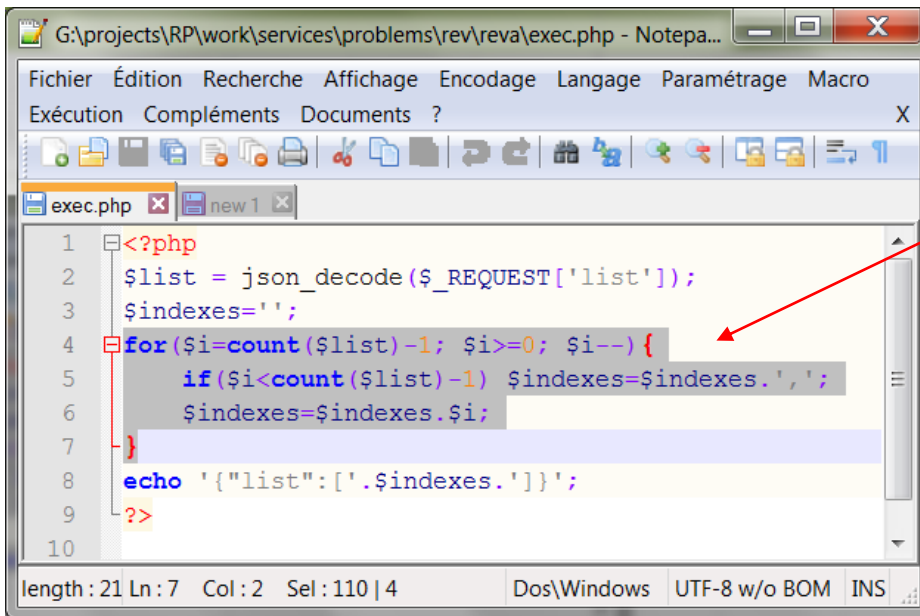


We suppose you are using Firefox, Chrome or IE.

We suppose you have a running, and well-configured,
apache server on your PC.



Remember what we did with the "rev" problem.



```
1 <?php
2 $list = json_decode($_REQUEST['list']);
3 $indexes='';
4 for($i=count($list)-1; $i>=0; $i--){
5     if($i<count($list)-1) $indexes=$indexes.',';
6     $indexes=$indexes.$i;
7 }
8 echo '{"list":['.$indexes.']}';
9 ?>
10
```

length: 21 Ln: 7 Col: 2 Sel: 110 | 4 Dos\Windows UTF-8 w/o BOM INS

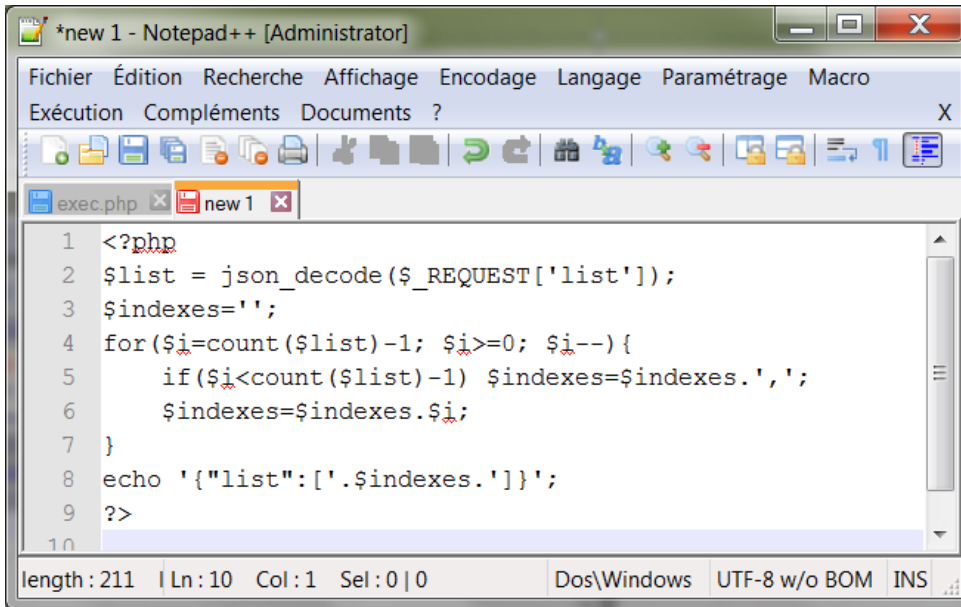
We reversed the list !

This was extremely simple.

But a list of GPS points is much more than just an array.



Make a copy of the PHP file.



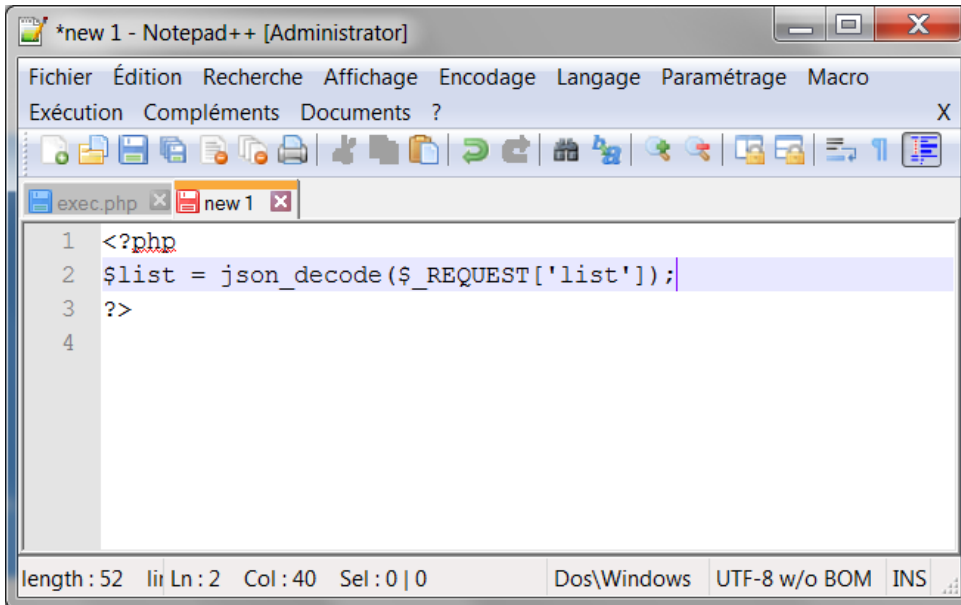
The screenshot shows a Notepad++ window titled '*new 1 - Notepad++ [Administrator]'. The menu bar includes 'Fichier', 'Édition', 'Recherche', 'Affichage', 'Encodage', 'Langage', 'Paramétrage', and 'Macro'. The toolbar contains various icons for file operations and editing. The active tab is 'new 1', and the code editor contains the following PHP code:

```
1 <?php
2 $list = json_decode($_REQUEST['list']);
3 $indexes='';
4 for($i=count($list)-1; $i>=0; $i--){
5     if($i<count($list)-1) $indexes=$indexes.',';
6     $indexes=$indexes.$i;
7 }
8 echo '{"list":['.$indexes.']}';
9 ?>
10
```

The status bar at the bottom indicates 'length: 211 | Ln: 10 Col: 1 Sel: 0 | 0' and 'Dos\Windows UTF-8 w/o BOM INS'.

We will avoid to pollute the original exec.php file.

Let's restart from the beginning.



The screenshot shows a Notepad++ window titled '*new 1 - Notepad++ [Administrator]'. The menu bar includes 'Fichier', 'Édition', 'Recherche', 'Affichage', 'Encodage', 'Langage', 'Paramétrage', and 'Macro'. The toolbar contains various icons for file operations and editing. The main text area shows the following PHP code:

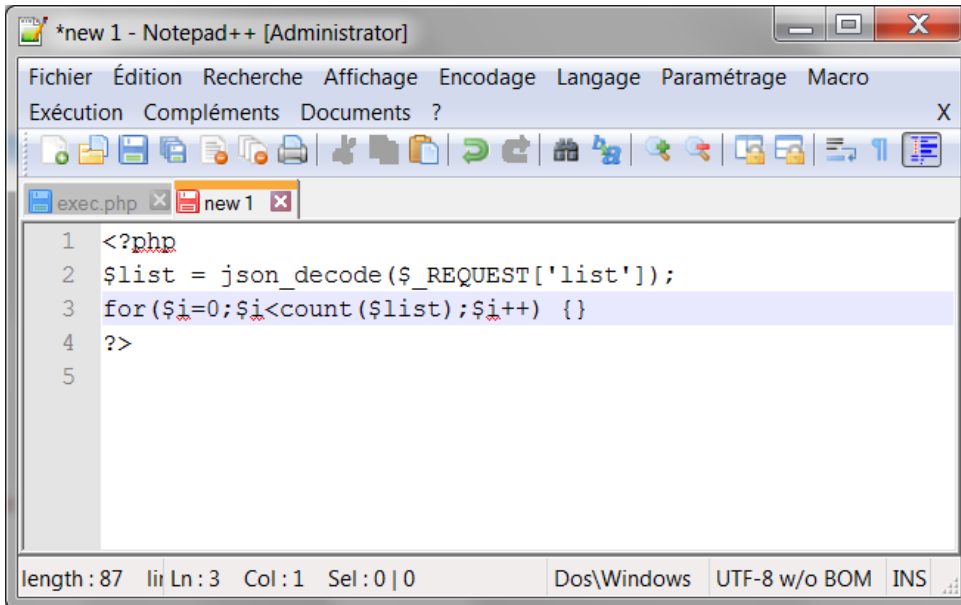
```
1 <?php
2 $list = json_decode($_REQUEST['list']);
3 ?>
4
```

The status bar at the bottom indicates 'length: 52', 'Ln: 2', 'Col: 40', 'Sel: 0 | 0', 'Dos\Windows', 'UTF-8 w/o BOM', and 'INS'.

Now, keep the `json_decode` line

A `list` is a JSON-coded input parameter, and must be decoded. The `list` PHP variable now holds the list of GPS points.

Scan the array of GPS points.



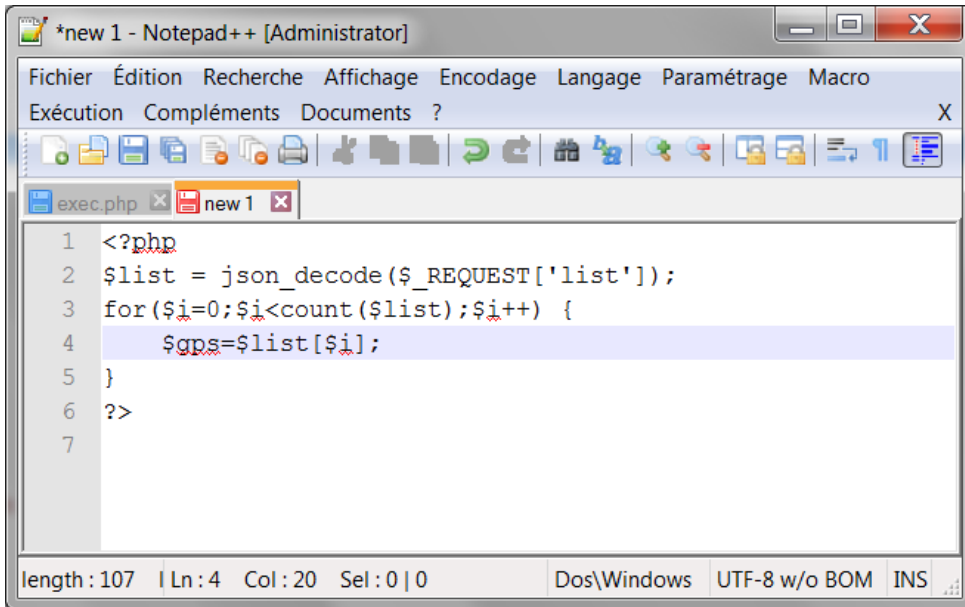
```
*new 1 - Notepad++ [Administrator]
Fichier  Édition  Recherche  Affichage  Encodage  Langage  Paramétrage  Macro
Exécution  Compléments  Documents  ?
exec.php  new 1
1  <?php
2  $list = json_decode($_REQUEST['list']);
3  for($i=0;$i<count($list);$i++) {}
4  ?>
5
length: 87  lln: 3  Col: 1  Sel: 0|0  Dos\Windows  UTF-8 w/o BOM  INS
```

A list is an array.
`count($list)` is its size.

We scan all the elements of `$list` with a `for` loop.
`$i` is the index of each element of that array.
We currently do nothing.



Access each element of the array.



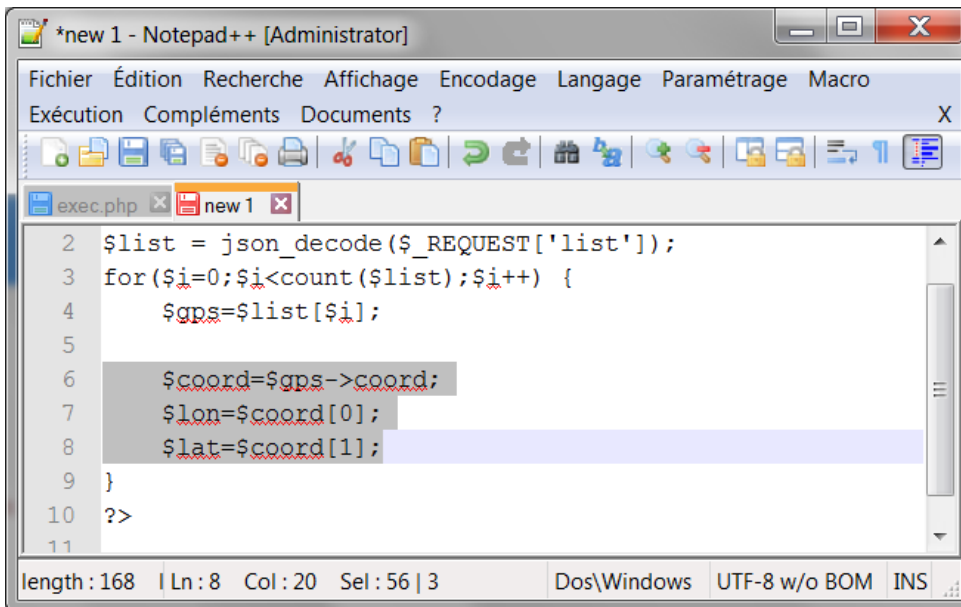
```
*new 1 - Notepad++ [Administrator]
Fichier  Édition  Recherche  Affichage  Encodage  Langage  Paramétrage  Macro
Exécution  Compléments  Documents  ?
exec.php  new 1
1  <?php
2  $list = json_decode($_REQUEST['list']);
3  for($i=0;$i<count($list);$i++) {
4      $gps=$list[$i];
5  }
6  ?>
7

length: 107 | Ln: 4 Col: 20 Sel: 0 | 0 Dos\Windows UTF-8 w/o BOM INS
```

`$gps` is the current element.

Now we can look into each element.

Get the coordinates of a GPS Point.



```
*new 1 - Notepad++ [Administrator]
Fichier  Édition  Recherche  Affichage  Encodage  Langage  Paramétrage  Macro
Exécution  Compléments  Documents  ?
exec.php  new 1
2 $list = json_decode($_REQUEST['list']);
3 for($i=0;$i<count($list);$i++) {
4     $gps=$list[$i];
5
6     $coord=$gps->coord;
7     $lon=$coord[0];
8     $lat=$coord[1];
9 }
10 ?>
11
length: 168 | Ln: 8 Col: 20 Sel: 56 | 3 Dos\Windows UTF-8 w/o BOM INS
```

A GPS Point has coordinates.
\$gps->coord are these ones.

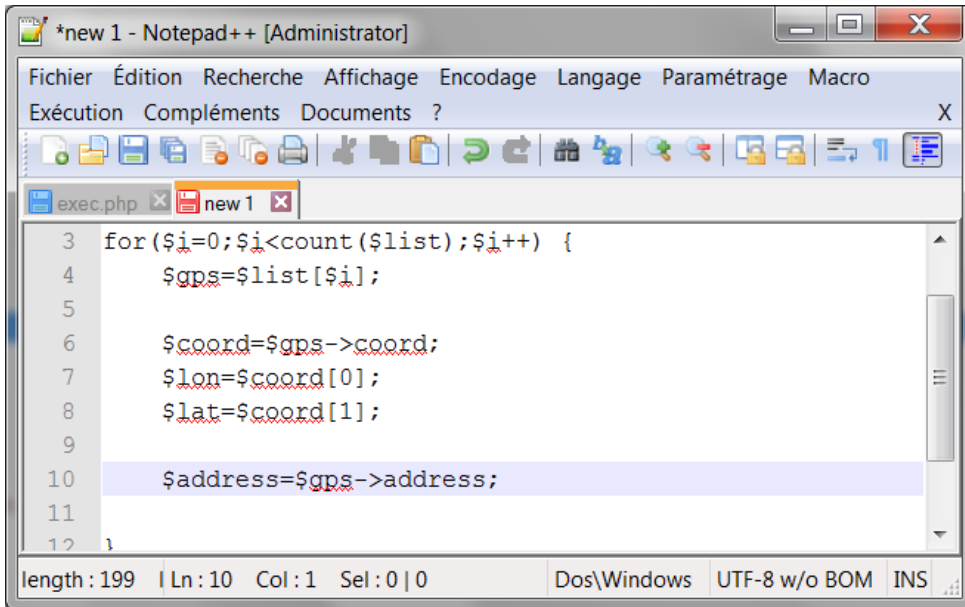
Now \$coord contains them:

the longitude is \$coord[0]

the latitude is \$coord[1]



Get the address of a GPS point.



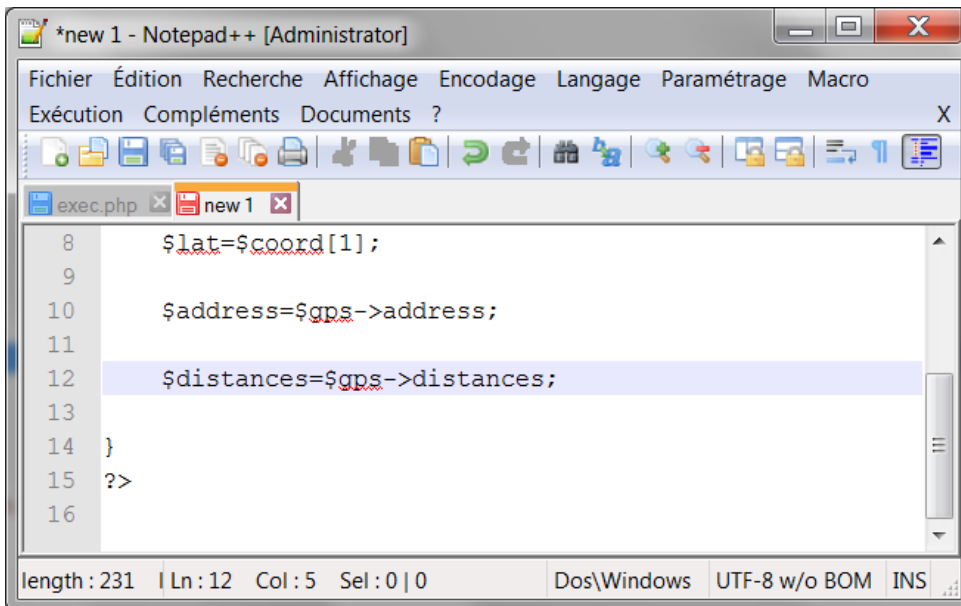
```
*new 1 - Notepad++ [Administrator]
Fichier  Édition  Recherche  Affichage  Encodage  Langage  Paramétrage  Macro
Exécution  Compléments  Documents  ?
exec.php  new 1
3  for($i=0;$i<count($list);$i++) {
4      $gps=$list[$i];
5
6      $coord=$gps->coord;
7      $lon=$coord[0];
8      $lat=$coord[1];
9
10     $address=$gps->address;
11
12 }
```

length: 199 | Ln: 10 Col: 1 Sel: 0 | 0 Dos\Windows UTF-8 w/o BOM INS

A GPS Point has an address.
\$gps->address is this address.

Now \$address contains the address.

Get the array of distances of a GPS point.



```
*new 1 - Notepad++ [Administrator]
Fichier  Édition  Recherche  Affichage  Encodage  Langage  Paramétrage  Macro
Exécution  Compléments  Documents  ?
exec.php  new 1
8   $lat=$coord[1];
9
10  $address=$gps->address;
11
12  $distances=$gps->distances;
13
14  }
15  ?>
16

length: 231 | Ln: 12 Col: 5 Sel: 0 | 0  Dos\Windows UTF-8 w/o BOM  INS
```

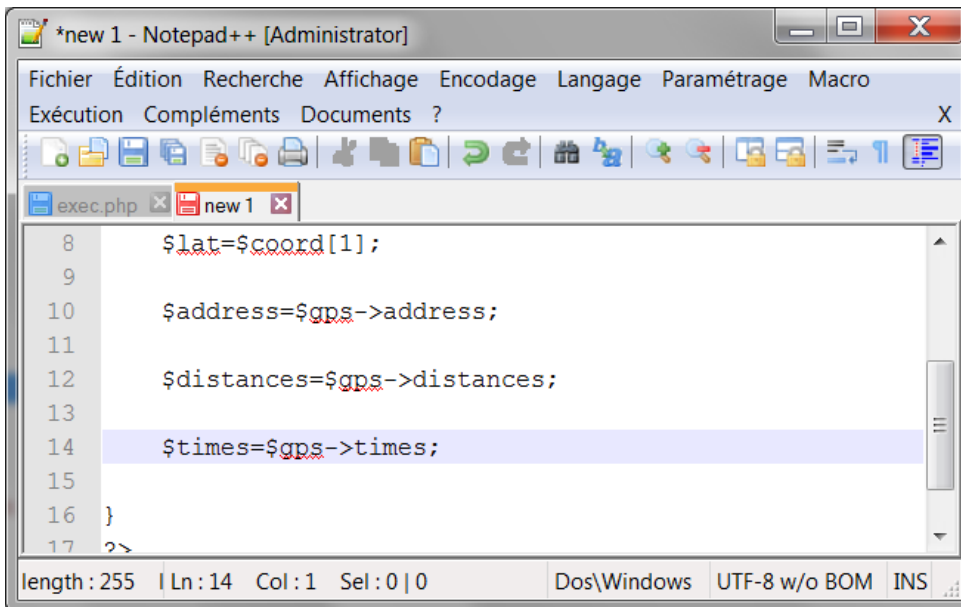
A GPS Point has an array of distances to all the other GPS points.
`$gps->distances` is this array.

Now `$distances` contains these distances.

Be very careful here, *array are numbered from 0 to N-1* (not from 1 to N) and the order is the same than the browser's.



Get the array of times of a GPS point.



```
*new 1 - Notepad++ [Administrator]
Fichier  Édition  Recherche  Affichage  Encodage  Langage  Paramétrage  Macro
Exécution  Compléments  Documents  ?
exec.php  new 1
8   $lat=$coord[1];
9
10  $address=$gps->address;
11
12  $distances=$gps->distances;
13
14  $times=$gps->times;
15
16  }
17  ?>
length: 255 | Ln: 14 Col: 1 Sel: 0 | 0 Dos\Windows UTF-8 w/o BOM INS
```

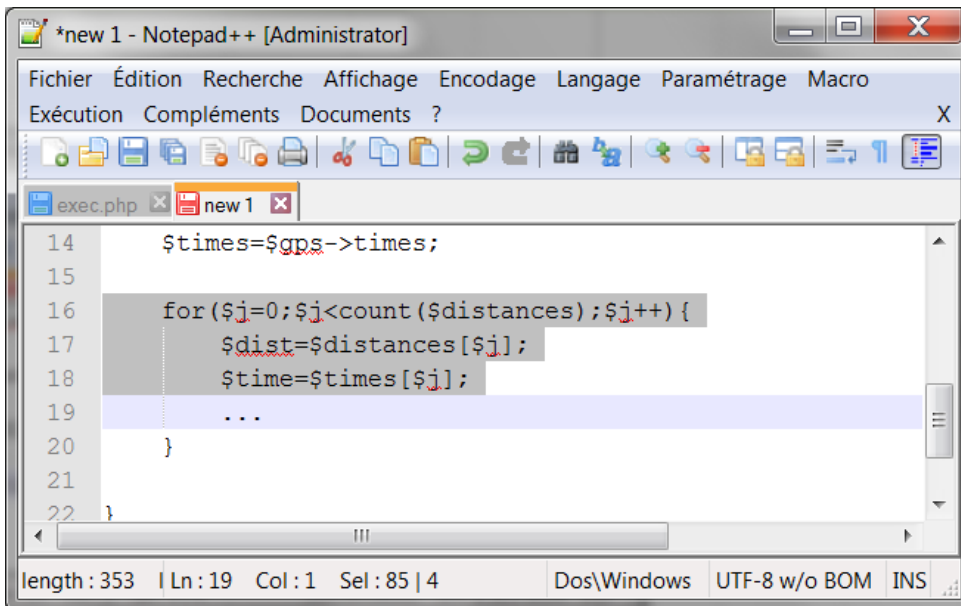
A GPS Point has an array of times to all the other GPS points.
`$gps->times` is this array.
Its size is the same than the list's.

Now `$times` contains these distances.

Be very careful here, *array are numbered from 0 to N-1* (not from 1 to N) and the order is the same than the browser's.



Scan the distances and the times.



```
*new 1 - Notepad++ [Administrator]
Fichier  Édition  Recherche  Affichage  Encodage  Langage  Paramétrage  Macro
Exécution  Compléments  Documents  ?
exec.php  new 1
14  $times=$gps->times;
15
16  for($j=0;$j<count($distances);$j++){
17      $dist=$distances[$j];
18      $time=$times[$j];
19      ...
20  }
21
22 }
```

length: 353 | Ln: 19 Col: 1 Sel: 85 | 4 Dos\Windows UTF-8 w/o BOM INS

`$dist` is the distance from `$i` to `$j`
`$time` is the time from `$i` to `$j`

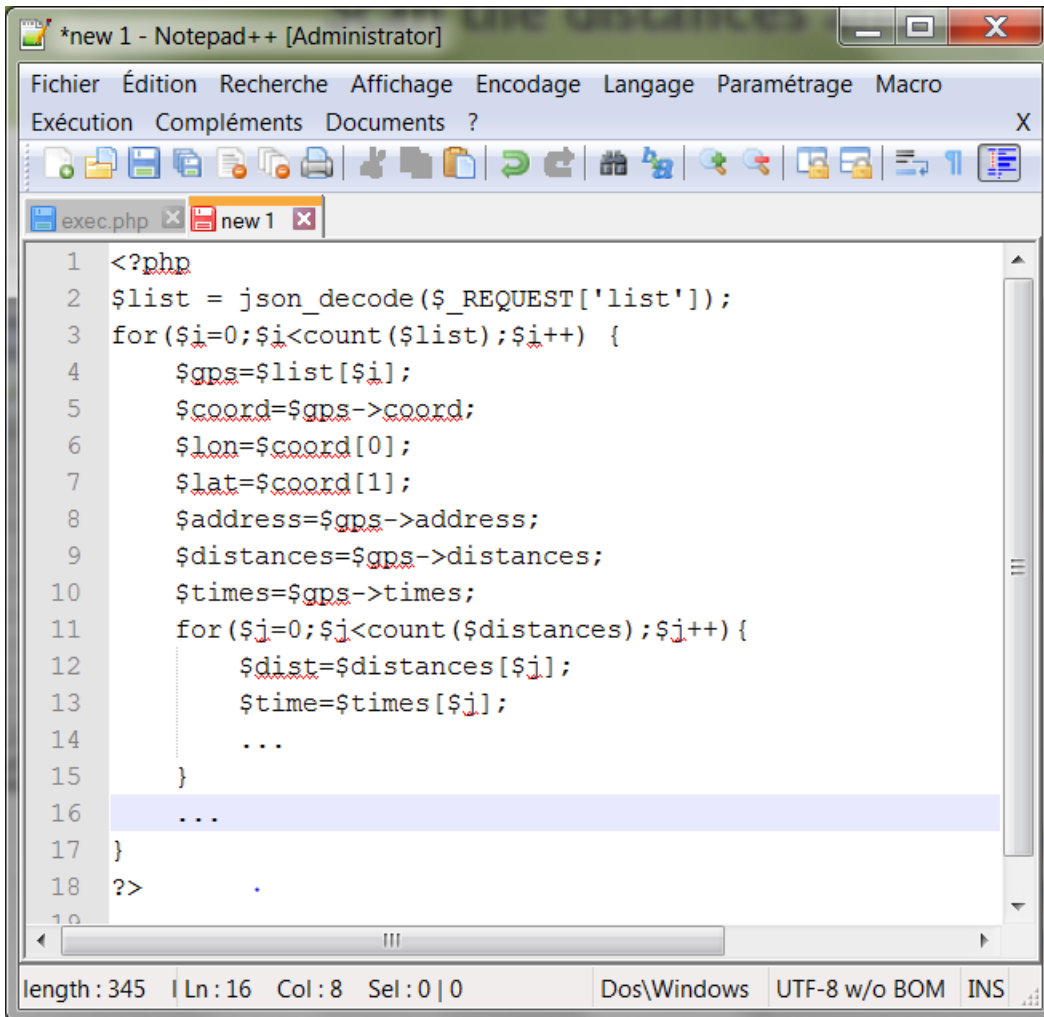
`count($distances)`
`count($times)`
`count($list)`

is the size of the array of distances
is the size of the array of times
is the size of the array of GPS points

And they are all equal !



Scan the distances and the times.



```
*new 1 - Notepad++ [Administrator]
Fichier  Édition  Recherche  Affichage  Encodage  Langage  Paramétrage  Macro
Exécution  Compléments  Documents  ?
exec.php  new 1
1  <?php
2  $list = json_decode($_REQUEST['list']);
3  for($i=0;$i<count($list);$i++) {
4      $gps=$list[$i];
5      $coord=$gps->coord;
6      $lon=$coord[0];
7      $lat=$coord[1];
8      $address=$gps->address;
9      $distances=$gps->distances;
10     $times=$gps->times;
11     for($j=0;$j<count($distances);$j++){
12         $dist=$distances[$j];
13         $time=$times[$j];
14         ...
15     }
16     ...
17 }
18 ?>
```

length : 345 | Ln : 16 | Col : 8 | Sel : 0 | 0 | Dos\Windows | UTF-8 w/o BOM | INS

Here is a skeleton of code.
It illustrates all you need to know:
- to use an input list of GPS points
- and create new web-services.



Conclusion



Congratulations !

Now you can use an input list
of GPS point to develop your own web-service.

