**Exemple\_Sarcini\_Programe**

**1. CUM Transmitem o variabila simpla din PHP in HTML?**

**1.sablon1.php**

**<?php**

**$hello = 'Salut!';**

**require\_once('index1.html');**

**?>**

**2. index1.html**

**<html>**

**<head>**

**</head>**

**<body>**

**<div>Variabila simpla din php:**

**<span><?php echo $hello.”</br>”; ?></span>**

**</div>**

**</body>**

**</html>**

**Variabila simpla din php:  
Salut!**

**2. CUM Transmiten un masiv din PHP in HTML? Prezentati rezultatul printr-o lista neordonata.**

**1.sablon2.php**

**<?php**

**$hellom = array('Salut!', "Privet!");**

**require\_once('index2.html');**

**?>**

**2. index2.html**

**<html>**

**<head>**

**</head>**

**<body>**

**<div>Masiv din php:**

**<ul>**

**<?php foreach ($hellom as $str) { ?>**

**<li>**

**<?php echo $str.”</br>”; ?>**

**</li>**

**<?php } ?>**

**</ul>**

**</div>**

**</body>**

**</html>**

**Masiv din php:**

* **Salut!**
* **Privet!**

**3. CUM Transmiten date din HTML in PHP? Folositi metoda POST**

**1. index3.html**

**<html>**

**<head>**

**</head>**

**<body>**

**<form method="POST">**

**Nume: <input type="text" name="login">**

**Parola: <input type="password" name="password">**

**<input type="submit" value="OK">**

**</form>**

**</body>**

**</html>**

**Nume:  Parola:  **

**3. CUM primim date din HTML in PHP? Folositi metoda POST**

**1. sablon3.php**

**<?php**

**if(!empty($\_POST))**

**{**

**echo '<pre>';**

**print\_r($\_POST);**

**die();**

**}**

**require\_once('index3.html');**

**?>**

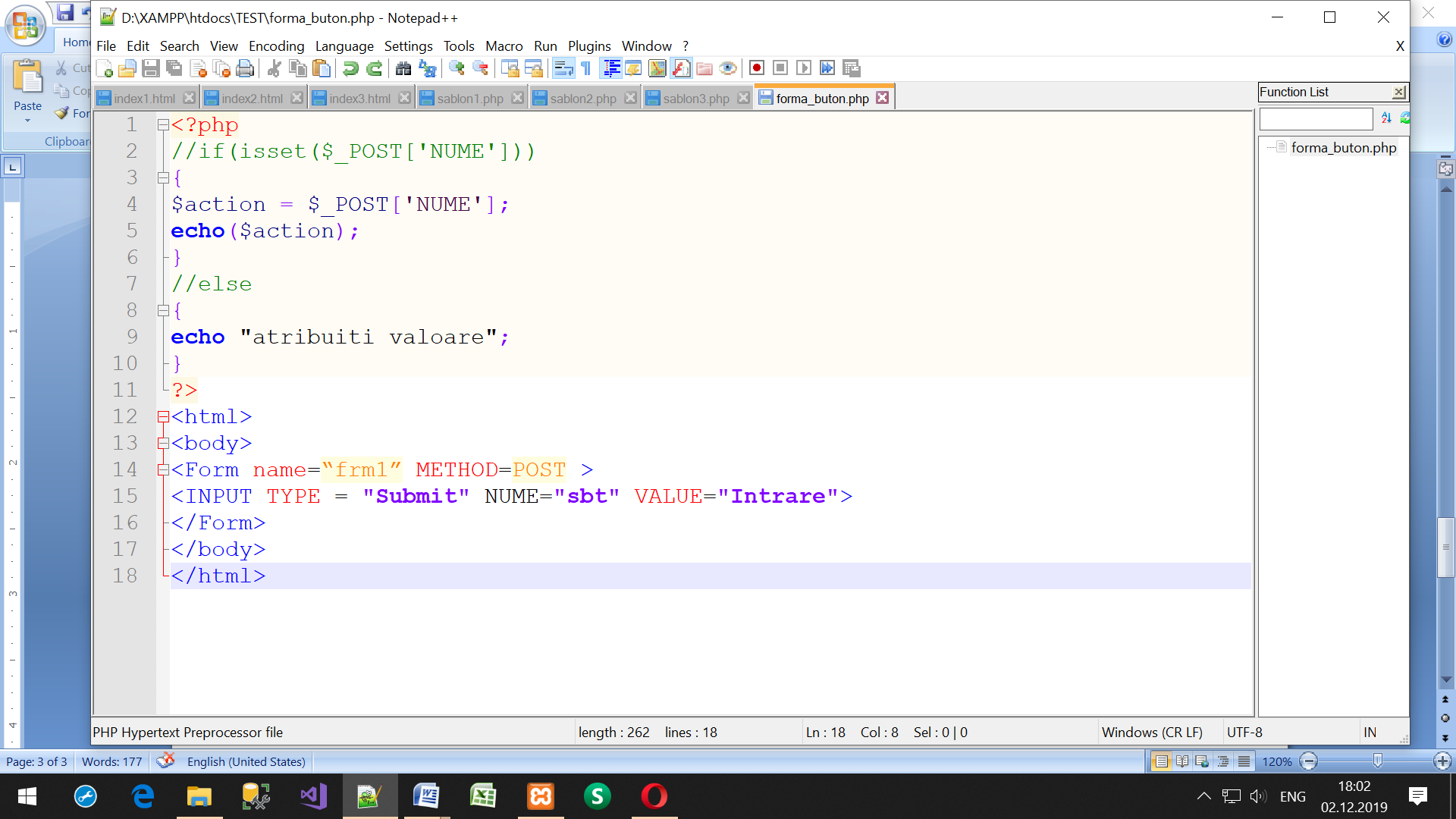
**Array**

**(**

**[login] => mihai**

**[password] => 123456789**

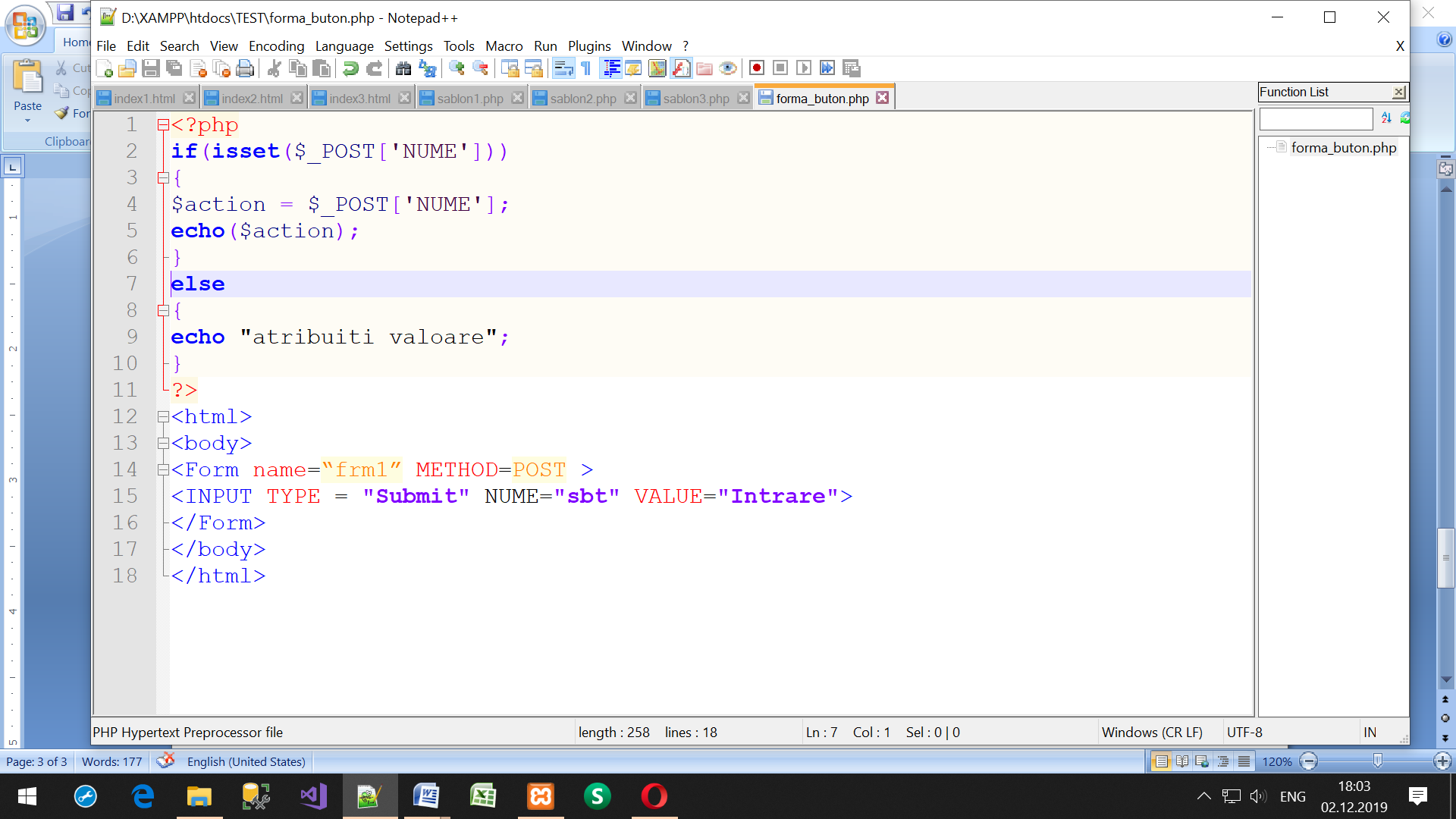
**)**

****

**Notice**: Undefined index: NUME in **D:\XAMPP\htdocs\TEST\forma\_buton.php** on line **4**  
atribuiti valoare

Top of Form

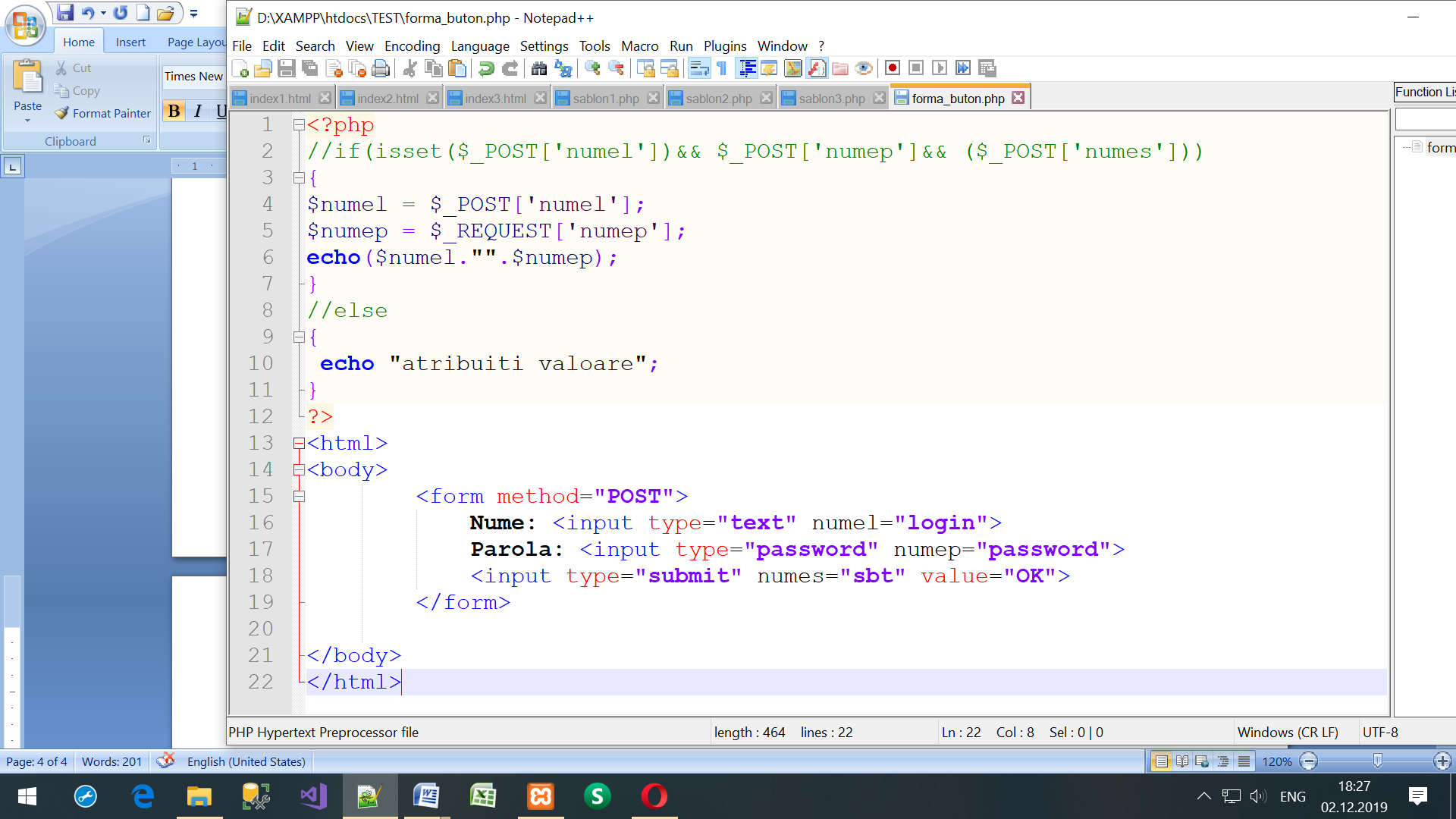




atribuiti valoare

Top of Form

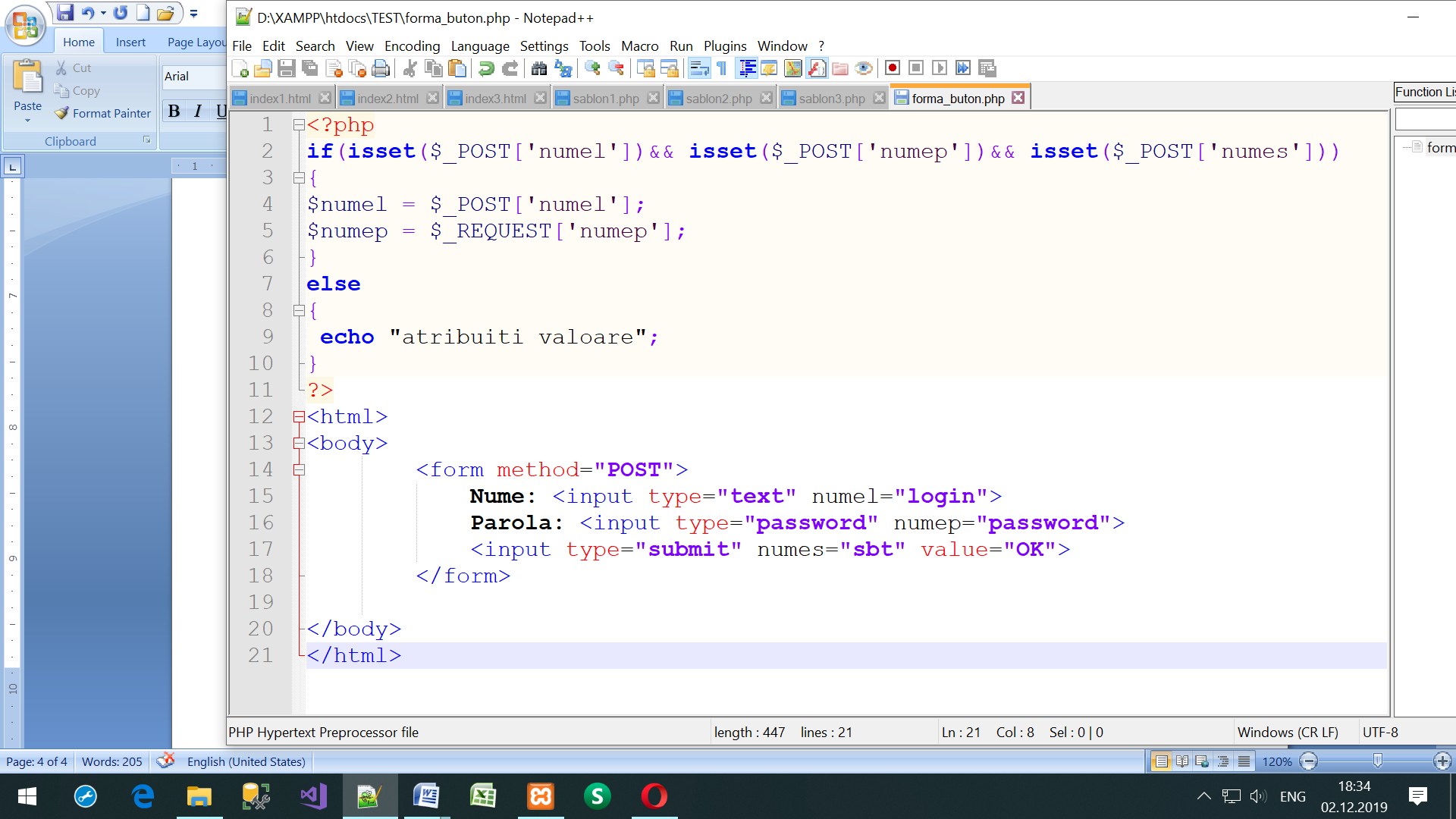


****

**Notice**: Undefined index: numel in **D:\XAMPP\htdocs\TEST\forma\_buton.php** on line **4**  
  
**Notice**: Undefined index: numep in **D:\XAMPP\htdocs\TEST\forma\_buton.php** on line **5**  
atribuiti valoare

Top of Form

Nume:  Parola:  



atribuiti valoare

Nume:  Parola:  

Isset() - ??

**SE RECOMANDA DE UTILIZAT ASTFEL**

**if(isset($\_POST['name']) && isset($\_POST['email']) &&....)**

**{**

**// If isset then assign data to variable $name = $\_POST['name'],...**

**}**

**else**

**{**

**// Redirect to the error page**

**}**

# ‘AND’ vs ‘&&’ as operator in PHP

**The AND** operator is called logical operator. It returns true if both the operands are true.

**The &&** operator is called logical operator. It returns true if both operands are true.

**Functia isset()**

**Funcția isset ()** **este utilizată pentru a verifica dacă o variabilă este setată sau nu**. Dacă o variabilă este deja **unset** cu funcția **unset (),** aceasta nu va mai fi setată. Funcția **isset ()** **returna false** dacă variabila de testare conține o valoare **NULL**.

**Versiunea:**

(PHP 4 in sus)

**Sintaxa:**

**isset(variable1, variable2......)**

**Parameter:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Description** | **Required /Optional** | **Type** |
| variable1 | The variable being checked | Required | Mixed\* |
| variable2 ...... | More variable to be checked. | Optional | Mixed\* |

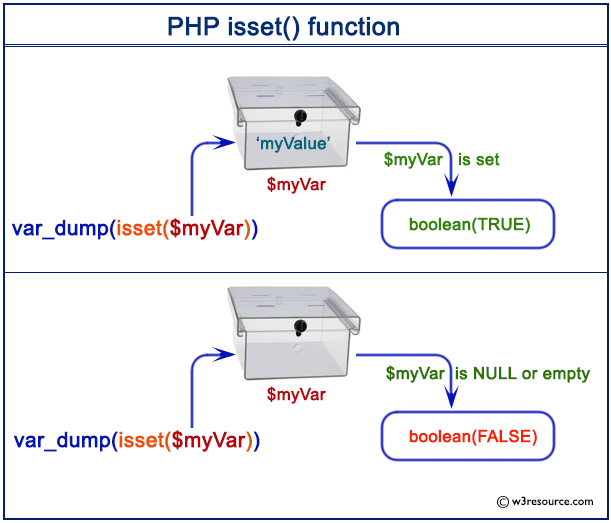
\*Mixed: Mixed indicates that a parameter may accept multiple (but not necessarily all) types.

**Return value:**

**TRUE** daca variabilile (variable1,variable2..) exista si au valori neegale cu NULL, FALSE.

**Value Type**: Boolean.

**Pictorial presentation of PHP isset() function**



**Example:**

<?php

$var1 = 'test'; //$var1 = 'test';

var\_dump(isset($var1));

?>

Output :

bool(true) //bool(false)

**MAI ESTE O FUNCTIE**

**Funcția empty()** este utilizată pentru a verifica dacă o variabilă este ***“goală/curata/nu are nimic in ea“*** sau ***nu***. Adesea, este posibil să întâlnim un scenariu, când trebuie să scrieti un cod dacă o variabilă este ***“goală/curata/nu are nimic in ea“***sau un alt lucru dacă nu este așa. În aceste situații, funcția **empty()** este utila.

**Versia:**

(PHP 4 in sus)

**Sintaxa:**

**empty(var\_name)**

**Return value:**

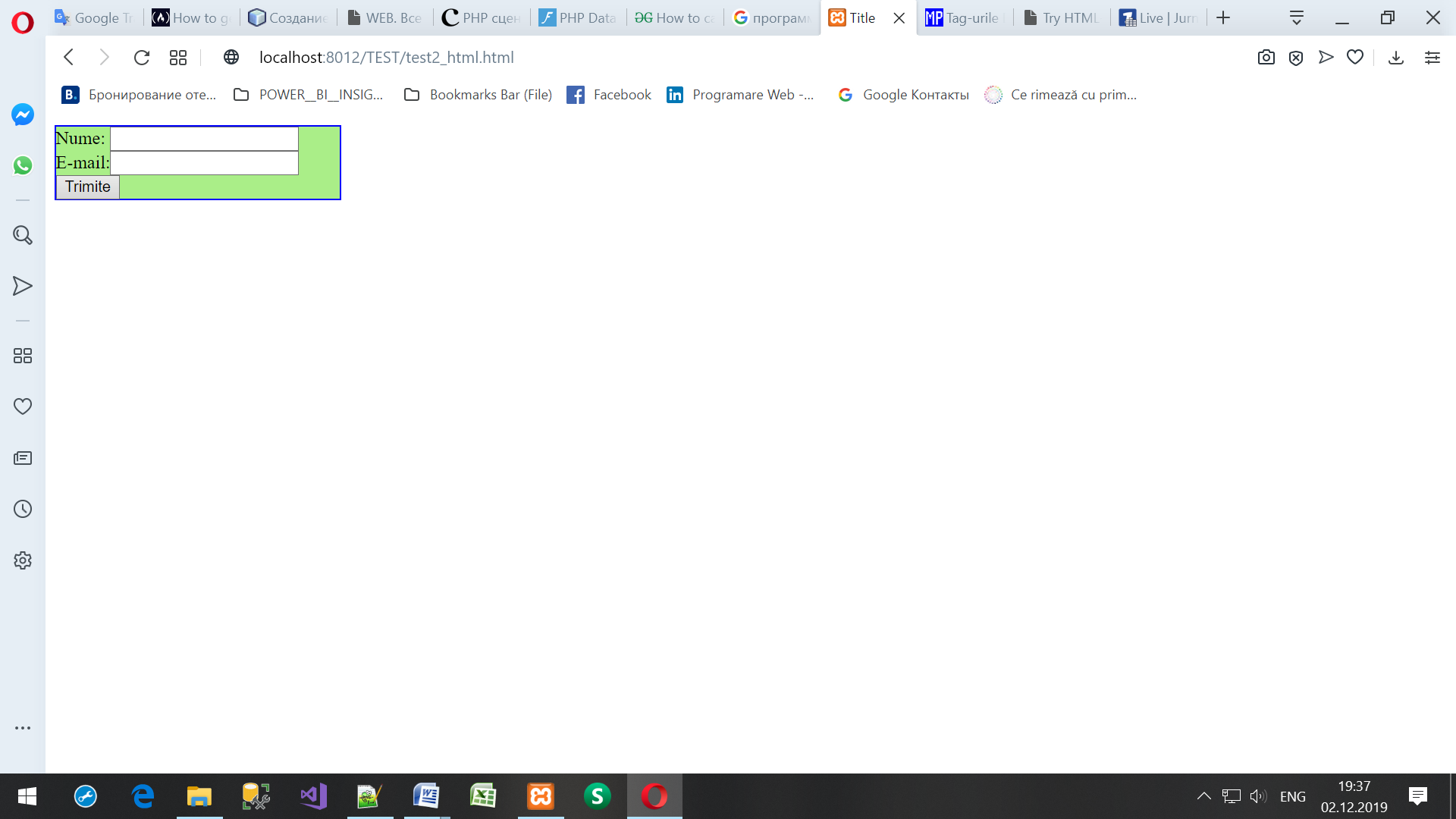
**FALSE** dacă **var\_name** are o valoare **non-goală** și valoare non-zero.

**Value Type**: Boolean

**List of empty things**:

* "0" (0 as a string)
* 0 (0 as an integer)
* "" (an empty string)
* NULL
* FALSE
* "" (an empty string)
* array() (an empty array)
* $var\_name; (a variable declared but without a value in a class)

**Sarcina!!**



<!doctype html>

<html>

<head>

<meta charset="utf-8" />

<title>Title</title>

<meta name="viewport" content="width=device-width, initial-scale=1" />

</head>

<body>

**<div style="width:250px; background:#aaee88; border:1px solid blue;">**

<form action="" method="**post**">

**Nume:** <input type="text"></input><br>

**E-mail:<**input type="text"></input><br>

<input type="submit" value="Trimite"></input>

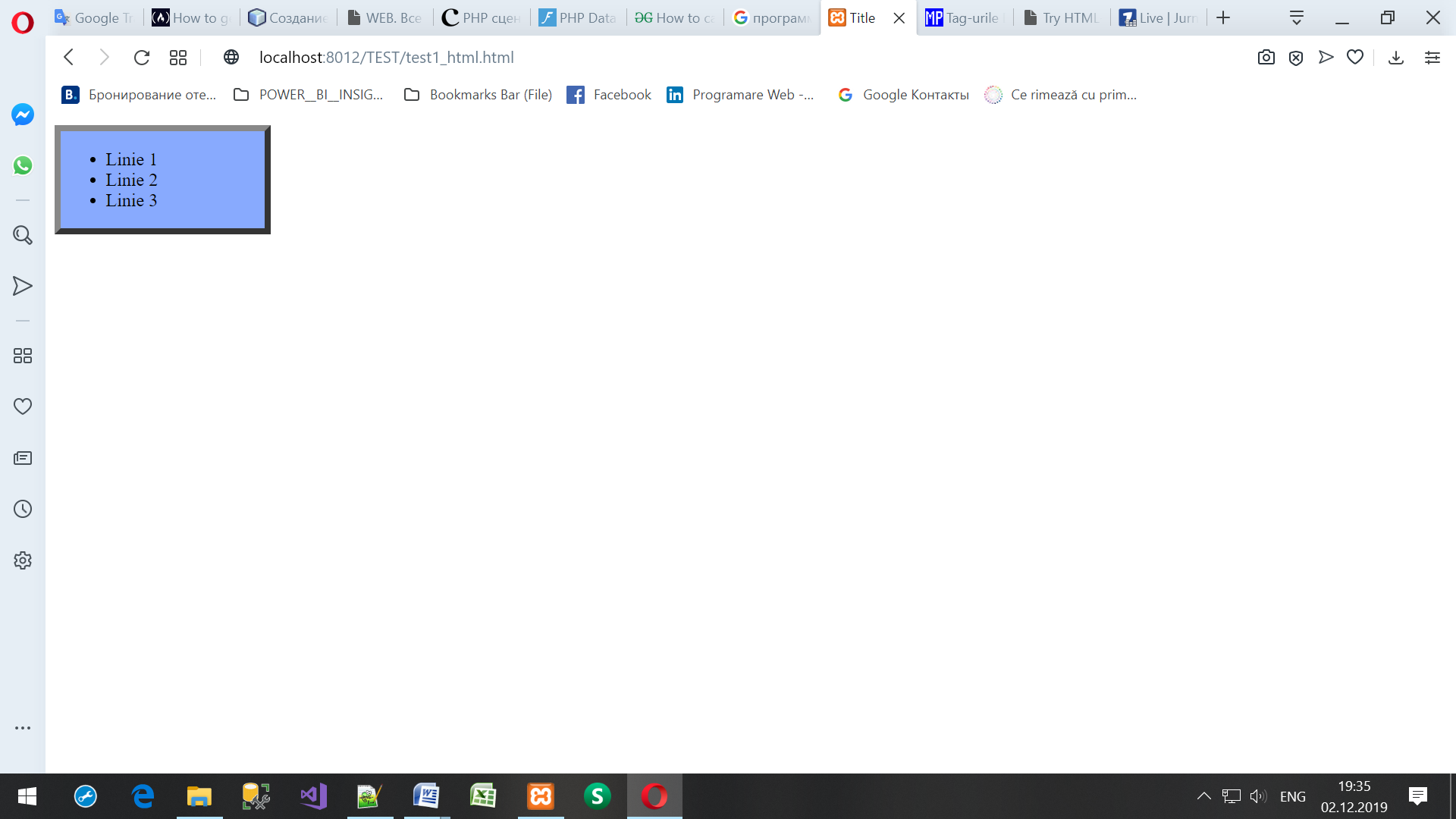
</form>

**</div>**

</body>

</html>

**Sarcina!!**



**<!doctype html>**

**<html>**

**<head>**

**<meta charset="utf-8" />**

**<title>Title</title>**

**<meta name="viewport" content="width=device-width, initial-scale=1" />**

**</head>**

**<body>**

**<div style="width:180px; background:#88aafe; border:5px outset #888888;">**

**<ul>**

**<li>Linie 1</li>**

**<li>Linie 2</li>**

**<li>Linie 3</li>**

**</ul>**

**</div>**

**</body>**

**</html>**

# Function array\_key\_exists()

**Array\_key\_exists ()** este o funcție a PHP și este utilizată pentru a verifica dacă o cheie sau un index specific este prezent în interiorul unui masiv sau nu. Funcția returnează **True** dacă cheia specificată se găsește în masiv, în caz contrar, returnează **fals.**

**Syntax**:

**array\_key\_exists($index, $array)**

**Type**

**boolean**

**Parametri: Această funcție are două argumente**

**1. $index (obligatoriu): acest parametru se referă la cheia care trebuie căutată într-un masiv de intrare.**

**2. $array (obligatoriu): Acest parametru se referă la masivul original în care dorim să căutăm indicele de cheie dat.**

**Valoare de returnare: Această funcție returnează o valoare booleană adică, TRUE sau FALSE, în funcție de faptul dacă cheia este prezentă în tablou sau nu.**

**Notă: Tastele amplasate vor returna rezultatul ca FALSE.**

**<!DOCTYPE html> test3\_html.php**

**<html>**

**<head>**

**<title>**

**Cum chemam o functie PHP printr-un clik de buton**

**</title>**

**</head>**

**<body style="text-align:center;">**

**<h4> Cum chemam o functie PHP printr-un clik de buton </h4>**

**<?php**

**if(array\_key\_exists('button1', $\_POST))**

**{ button1(); }**

**else**

**if(array\_key\_exists('button2', $\_POST))**

**{ button2(); }**

**function button1()**

**{ echo "A fost selectat Button1"; }**

**function button2()**

**{ echo "A fost selectat Button2 "; }**

**?>**

**<form method="post">**

**<input type="submit" name="button1"**

**class="button" value="Button1" />**

**<input type="submit" name="button2"**

**class="button" value="Button2" />**

**</form>**

**</head>**

**</html>**

**Cu rezultatele**

#### Cum chemam o functie PHP printr-un clik de buton

A fost selectat Button2

Top of Form

**PHP si pagina Web dinamica**

Ati realizat in Lucrarea de laborator nr 1 cind ati amplasat citeva pagini html (1.htm, 2.htm, 3.htm…) cu design similar in una singura cu acelasi aproape design – index.php – cu un continut aproximativ

<!—Antet paginii... -->  
<!—Meniul paginii -->  
<a href=’index.php?numPage=1’>Partea 1</a>  
<a href=’index.php?numPage=2’>Partea 2</a>  
<a href=’index.php?numPage=3’>Partea 3</a>  
**<?php**

/\* conectarea continutului paginii in dependenta de parametrul transmis – numarul partii paginii $numPage: \*/  
$numPage=$\_GET["numPage"];  
Include($numPage.".html");  
**?>**  
<!—Partea de jos a paginii... -->

**Transmiterea datelor din Javascript in PHP**

Exista 3 metode de transmitere a datelor din Javascript in PHP care corespund celor 3 metode de adresare a browser-ului la server

GET - datele sunt transmise în bara de adrese;

POST - trimiterea datelor formularului prin Submit/Trimite (reîncărcare pagină);

Cookie - trecerea cookie-urilor (cookie-urile) și citirea acestora la momentul încărcării scriptului PHP.

Exemplul precedent de pagina dinamica in PHP

