

Laborator 5

2020/2021

# Programarea aplicațiilor web

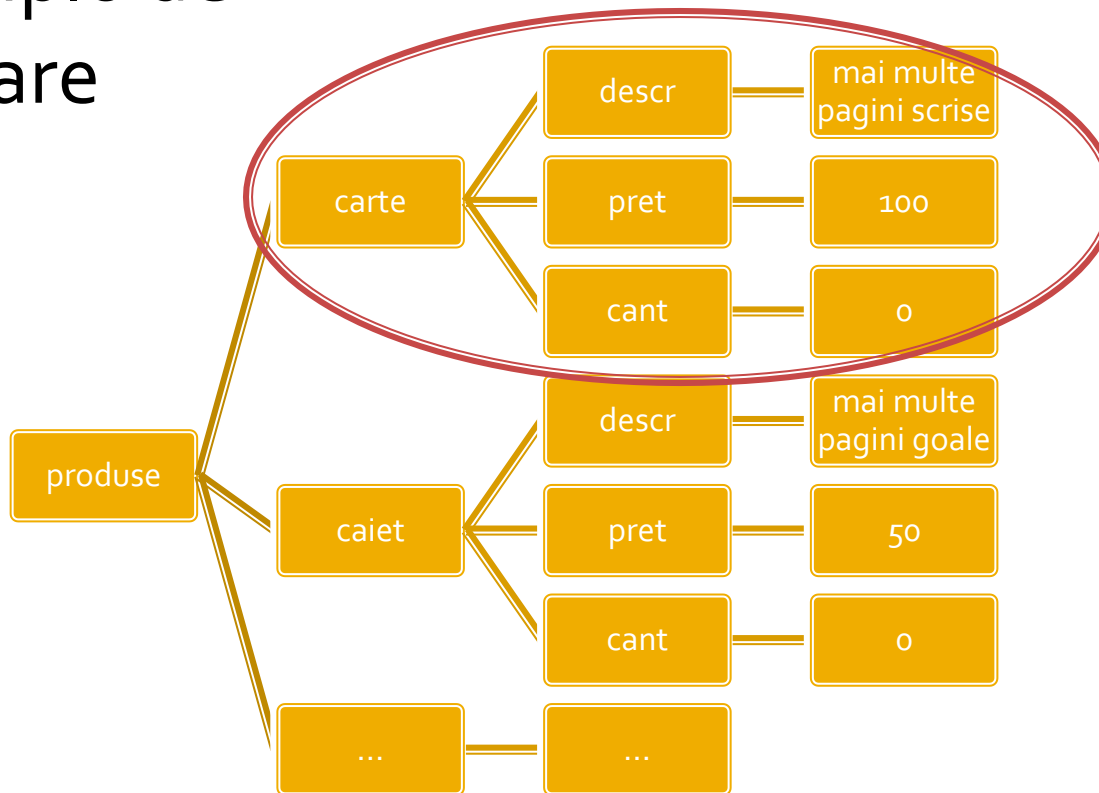
# Laborator 4

# Laborator L4

- Sa se creeze un magazin simplu virtual care:
  - sa prezinte utilizatorului o lista de produse si preturi (constanta – maxim 5 produse)
  - sa preia de la acesta numarul de produse dorit
  - sa calculeze suma totala
  - sa adauge TVA 19%
  - sa prezinte un raport care sa contina:
    - total de plata
    - ora comenzii

# Laborator 4 – Matrice produse

- exemplu de grupare



# Laborator – L4 - rezultat

## Magazin online Firma X SRL

### Lista Produse

Nr.	Produs	Pret
1	Carti	100
2	Caiete	50
3	Penare	150
4	Stilouri	125
5	Creioane	25
<a href="#">Comanda</a>		

## Magazin online Firma X SRL

### Realizati comanda

Nr.	Produs	Pret	Cantitate
1	Carti	100	<input type="text" value="1"/>
2	Caiete	50	<input type="text" value="2"/>
3	Penare	150	<input type="text" value="1"/>
4	Stilouri	125	<input type="text" value="0"/>
5	Creioane	25	<input type="text" value="0"/>

## Magazin online Firma X SRL

### Rezultate comanda

Pret total (fara TVA): 350

Pret total (cu TVA): 416.5

Comanda receptionata la data: 17/03/2010 ora 08:24

# Lista produse

antet.php

```
<html>
<head>
<title>Magazin online Firma X
SRL</title>
</head>
<body bgcolor="#CCFFFF"><?php
define('PRET_CARTE',100);
define('PRET_CALET',50);
define('PRET_PENAR',150);
define('PRET_STILOU',125);
define('PRET_CREION',25);
//orice cod comun PHP
?><table width="600" border="0"
align="center">
<tr><td></td></tr>
<tr><td height="600" valign="top"
bgcolor="#FFFFCC">
<h1>Magazin online Firma X SRL</h1>
```

subsol.php

```
</td></tr>
</table>
</body>
</html>
```

\*.php

```
<?php require('antet.php');?>
<h2>Lista Produse</h2>
<table border="1">
...
</table>
<?php require('subsol.php');?>
```

# Tablou produse

- una din structurile posibile

```
$produse = array ( 'carte' => array ("descr" => "mai multe pagini scrise", "pret" => 100, "cant" => 0),  
                  'caiet' => array ("descr" => "mai multe pagini goale", "pret" => 50, "cant" => 0),  
                  'penar' => array ("descr" => "loc de depozitat instrumente", "pret" => 150, "cant" => 0),  
                  'stilou' => array ("descr" => "instrument de scris albastru", "pret" => 125, "cant" => 0),  
                  'creion' => array ("descr" => "instrument de scris gri", "pret" => 25, "cant" => 0)  
                );
```

- se urmareste obtinerea unei structuri clare
  - usor de modificat/adaugat date
  - usor de utilizat
- daca definitia se introduce in fisierul antet va fi accesibila in toate fisierele individuale

# Laborator 5

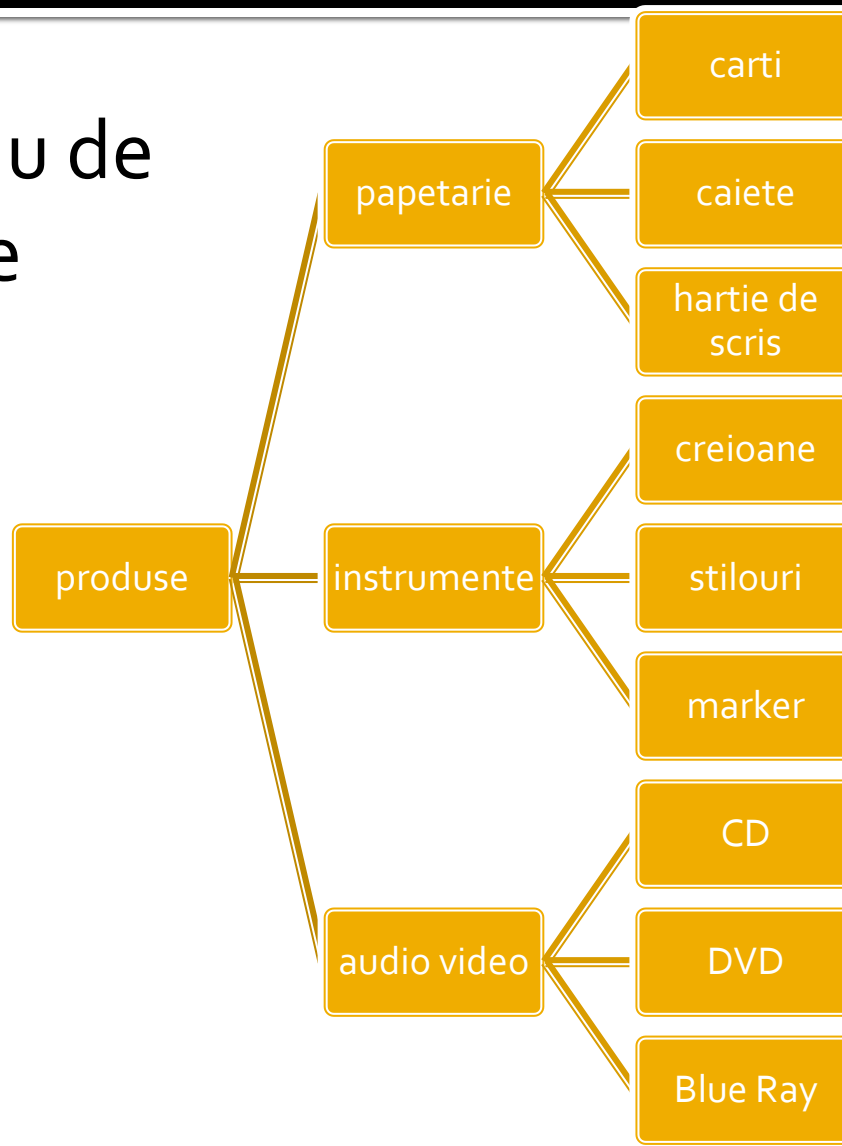


# Laborator 5

- Sa se continue magazinul virtual cu:
  - produsele sunt grupate pe **categorii** de produse
  - sa prezinte utilizatorului o lista de categorii de produse pentru a alege
  - sa prezinte utilizatorului o lista de produse si preturi in categoria aleasa
  - lista de produse si preturi se citeste dintr-un **fisier**
  - se preia comanda si se calculeaza suma totala
- Optional
  - se creaza o pagina prin care vanzatorul poate **modifica** preturile si produsele
  - fisierul care contine date in format **XML**

# Laborator 4

- exemplu de grupare



# Rezultat

## Categorii Produse

Alegeti categoria:

Nr.	Categorie	Total Produse
1	<a href="#">Papetarie</a>	3
2	<a href="#">Instrumente</a>	3
3	<a href="#">Audio-video</a>	3
4	<a href="#">Calculatoare</a>	3
5	<a href="#">Jucarii</a>	2

Total produse: 14

## Magazin online Firma X SRL

### Realizati comanda

Nr.	Produs	Pret	Cantitate
1	Carti	100	<input type="text" value="1"/>
2	Caiete	50	<input type="text" value="2"/>
3	Penare	150	<input type="text" value="1"/>
4	Stilouri	125	<input type="text" value="0"/>
5	Creioane	25	<input type="text" value="0"/>

## Magazin online Firma X SRL

### Rezultate comanda

Pret total (fara TVA): 350

Pret total (cu TVA): 416.5

Comanda receptionata la data: 17/03/2010 ora 08:24

Mod de lucru

# Laborator 5

# Tablou produse

- una din structurile posibile

```
$produse = array ( 'carte' => array ("descr" => "mai multe pagini scrise", "pret" => 100, "cant" => 0),  
                  'caiet' => array ("descr" => "mai multe pagini goale", "pret" => 50, "cant" => 0),  
                  'penar' => array ("descr" => "loc de depozitat instrumente", "pret" => 150, "cant" => 0),  
                  'stilou' => array ("descr" => "instrument de scris albastru", "pret" => 125, "cant" => 0),  
                  'creion' => array ("descr" => "instrument de scris gri", "pret" => 25, "cant" => 0)  
                );
```

- se urmareste obtinerea unei structuri clare
  - usor de modificat/adaugat date
  - usor de utilizat
- daca definitia se introduce in fisierul antet va fi accesibila in toate fisierele individuale

# Scrierea datelor pe disc

- Pentru a oferi posibilitatea beneficiarului aplicatiei (vanzator) sa poata adauga/sterge/modifica produse
  - din interfata browser
  - fara sa aiba cunostinte de programare
- E necesar ca tabloul **\$produse** sa fie creat in timpul rularii plecand de la un suport extern de date, accesibil pentru scriere vanzatorului
- Ulterior se va implementa aplicatia ce utilizeaza baze de date – momentan se vor scrie datele pe disc

# Utilizarea fisierelor – Functii

- `pointer = fopen(cale,mod)` deschide un fisier pentru operatii descrise de "mod". Se returneaza un pointer spre fisier de tip resursa care va fi folosit la operatiile urmatoare
- `fwrite(pointer,date)` – scrie datele in fisier (date – de tip string)
- `string = fread(pointer,cantitate)` citeste "cantitate" octeti din fisier
- `$matr = file(cale)` deschide fisierul identificat cu "cale" si citeste fiecare linie (incluzand \n) intr-un element distinct in matrice. \$matr de tip array, matrice de siruri

# Crearea fisierului

- crearea initiala se poate face prin modificarea **o singura data** a fisierului antet.php existent astfel incat sa scrie datele pe disc

```
$produse = array ( 'carte' => array ("descr" => "mai multe pagini scrise", "pret" => 100, "cant" => 0),  
                  'caiet' => array ("descr" => "mai multe pagini goale", "pret" => 50, "cant" => 0),  
                  'penar' => array ("descr" => "loc de depozitat instrumente", "pret" => 150, "cant" => 0),  
                  'stilou' => array ("descr" => "instrument de scris albastru", "pret" => 125, "cant" => 0),  
                  'creion' => array ("descr" => "instrument de scris gri", "pret" => 25, "cant" => 0)  
                );  
$handle = fopen("produse.txt", "wb");  
foreach ($produse as $prod => $detalii)  
    fwrite($handle,$prod."\t".$detalii['descr']."\t".$detalii['pret']."\t".$detalii['cant']."\r\n");
```



# Citirea fisierului pentru crearea matricii

```
$matr=file("produse.txt");  
echo "<pre>";  
print_r ($matr);  
echo "</pre>";  
foreach ($matr as $linie)  
{  
    $valori=explode("\t",$linie,4);  
    $produse[$valori[0]]=array ("descr" => $valori[1], "pret" => $valori[2], "cant" => $valori[3]);  
}
```

```
Array  
(  
    [0] => carte          mai multe pagini scrise legate  100    0  
    [1] => caiet          mai multe pagini goale legate   50    0  
    [2] => penar          loc de depozitat instrumente de scris 150    0  
    [3] => stilou         instrument de scris albastru   125    0  
    [4] => creion         instrument de scris gri 25      0
```

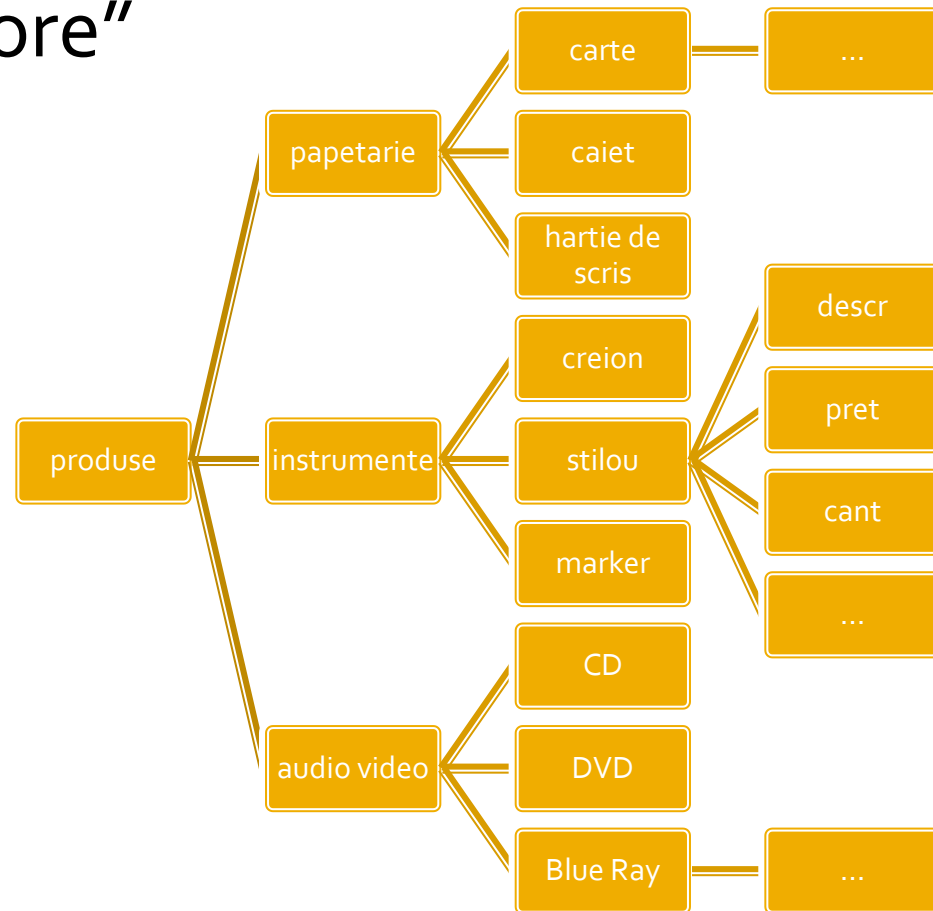
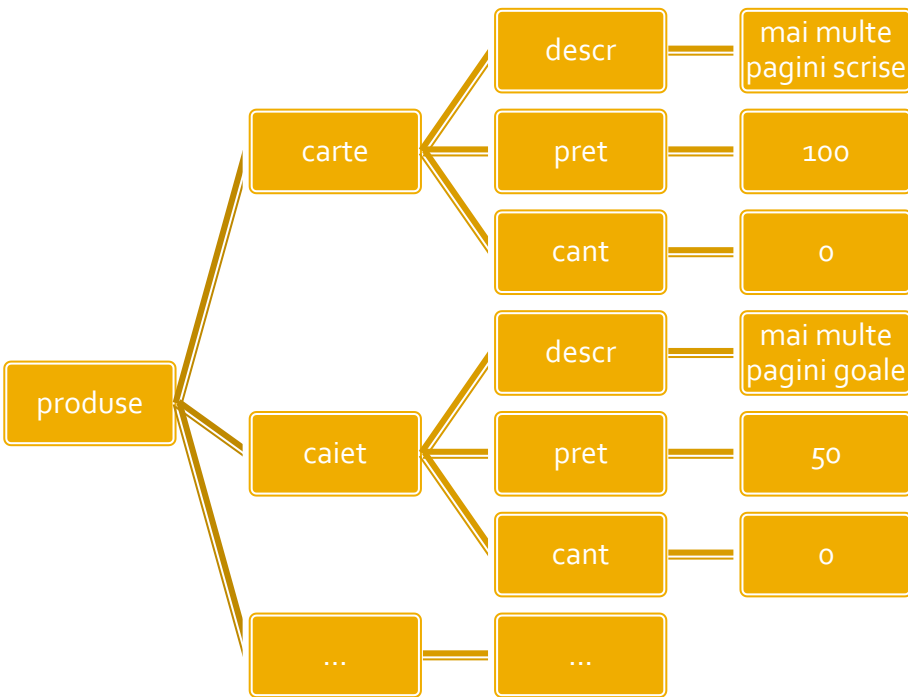
# produse.txt

- se pot utiliza si alte caractere pentru separare
  - esential: sa nu apara in date
  - TAB are efect vizual si in fisiere text

carte	mai multe pagini scrise legate	100	0
caiet	mai multe pagini goale legate	75	0
penar	loc de depozitat instrumente de scris	150	0
stilou	instrument de scris albastru	125	0
creion	instrument de scris gri	25	0

# Adaugare categorii

- adaugare nivel in "arbore"



# Scrierea datelor pe disc

- Structura datelor este total dependenta de algoritmul utilizat
- Modificarile sunt dificil de realizat si implica schimbarea fisierului de date si schimbarea codului in intregime
- Exemplu: gruparea elementelor pe categorii
  - `$produse["produs"] = ["caract."]`
  - `$produse["categorie"]["produs"] = ["caract."]`
- index-ul suplimentar introdus transforma o matrice bidimensionala in una tridimensionala
  - dificil de implementat cu fisiere text (caract/linii – 2D)

# Rezolvare

- modificarea fisierului text cu introducerea categoriei ca prim parametru in fisier

carte	mai multe pagini scrise legate	100	0	
caiet	mai multe pagini goale legate	75	0	
penar	loc de depozitat instrumente de scris	150	0	
stilou	instrument de scris albastru	125	0	
papetarie	carte	mai multe pagini scrise legate	100	0
papetarie	caiet	mai multe pagini goale legate	75	0
instrumente	penar	loc de depozitat instrumente de scris	150	0
instrumente	stilou	instrument de scris albastru	125	0

# Dezavantaj

- date redundante: numele categoriei apare listat pentru fiecare produs in parte
- sensibilitate la erori si nevoie de precizie in realizarea fisierului
  - numele categoriei trebuie scris exact de fiecare data
  - o mica greseala ("instrument" in loc de "instrumente") are ca efect crearea unei categorii suplimentare
- Daca numarul de produse e mare schimbarea e dificil de realizat manual, si complicat de realizat prin cod
- Produsele care nu apartin nici unei categorii nu vor putea fi diferite

# Cod

- Codul ramane in mare parte acelasi
- Cateva modificari sunt necesare

```
$matr=file("produse.txt");
foreach ($matr as $linie)
{
    $valori=explode("\t",$linie,4);
    $produse[$valori[0]]=array ("descr" => $valori[1], "pret" => $valori[2], "cant" => $valori[3]);
}
```


```
$matr=file("produse.txt");
foreach ($matr as $linie)
{
    $valori=explode("\t",$linie,5);
    $produse[$valori[0]] [$valori[1]]=array ("descr" => $valori[2], "pret" => $valori[3], "cant" =>
$valori[4]);
}
```

# Acces la date

## ■ Utilizare, cu doua bucle foreach

```
$index=1;
foreach ($produse as $prod => $detalii) //primul indice in $produse imi da produsul
{
    <tr><td><?php echo $index;?></td><td><?php echo ucfirst(strtolower($prod));?></td><td><?php echo
    $detalii['descr'];?></td><td align="center"><?php echo $detalii['pret'];?></td></tr>
    <?php $index++;
}
```

```
$index=1;
foreach ($produse as $categ => $lista_categ) //primul indice in $produse imi da categoria
    foreach ($lista_categ as $prod => $detalii) //al doilea indice in $produse imi da produsul
        //din categoria stabilita cu primul indice
        {
            <tr><td><?php echo $index;?></td><td><?php echo $categ;?></td><td><?php echo
            ucfirst(strtolower($prod));?></td><td><?php echo $detalii['descr'];?></td><td align="center"><?php
            echo $detalii['pret'];?></td></tr>
            <?php $index++;
        }
```





# Rezultat

**Magazin**

**Firma X SRL**

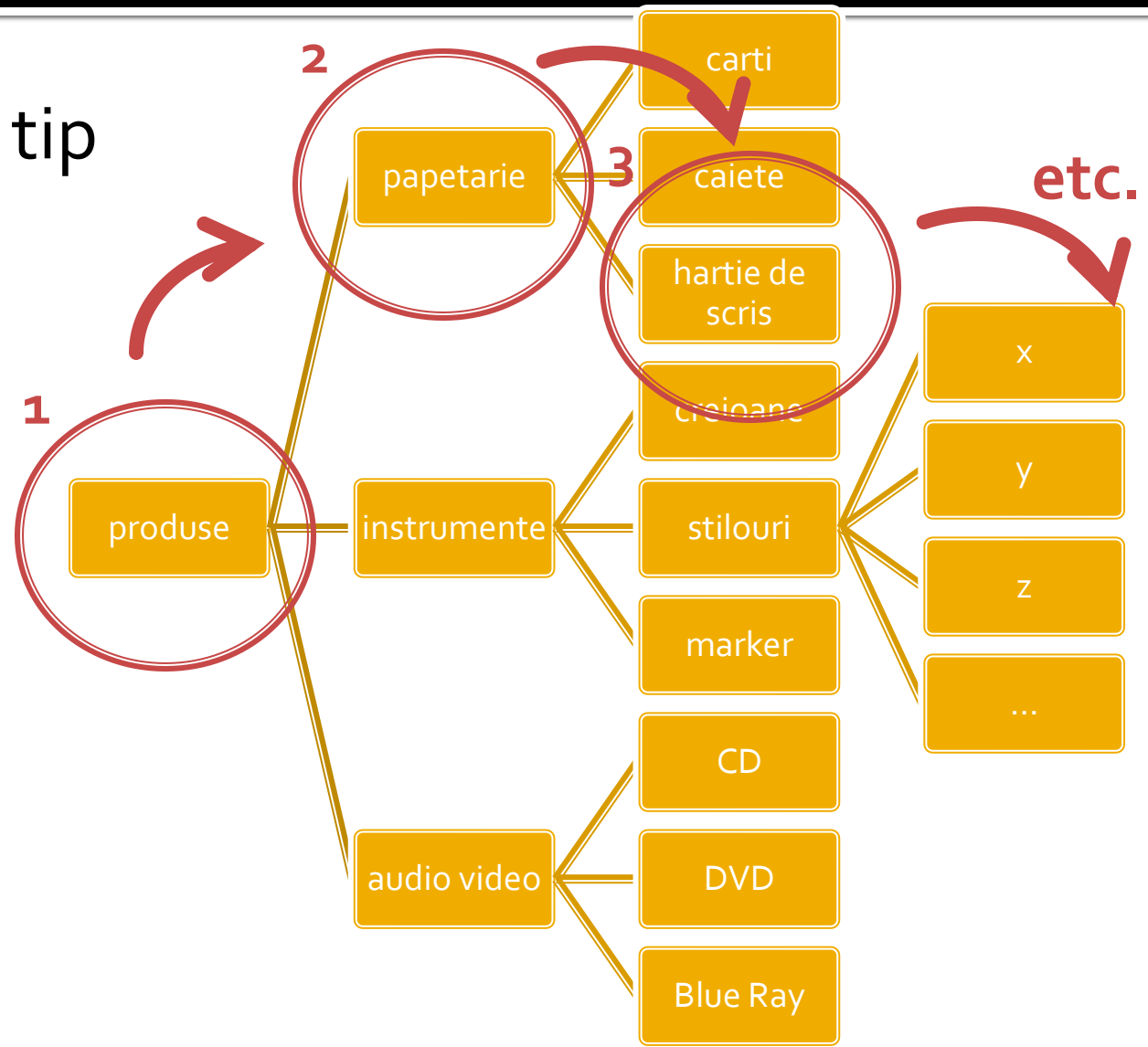
## Magazin online Firma X SRL

### Lista Produse

Nr.	Categorie	Produs	Descriere	Pret
1	papetarie	Carte	mai multe pagini scrise legate	100
2	papetarie	Caiet	mai multe pagini goale legate	75
3	papetarie	Hartie scris	mai multe pagini goale NElegate	50
4	instrumente	Penar	loc de depozitat instrumente de scris	150
5	instrumente	Stilou	instrument de scris albastru	125
6	instrumente	Creion	instrument de scris gri	25
7	instrumente	Marker	instrument de scris verde	25
8	audio-video	Cd	canta	50
9	audio-video	Dvd	vizual	100
10	audio-video	Blue ray	vizual extrem	500
<a href="#">Comanda</a>				

# Matrici

- structura tip “arbore”



# Lista categorii

- Aparitia categoriilor aduce necesitatea transmiterii de catre utilizator a informatiei despre categoria selectata

## Categorii Produse

Alegeti categoria:

Nr.	Categorie	Total Produse
1	<a href="#">Papetarie</a>	3
2	<a href="#">Instrumente</a>	3
3	<a href="#">Audio-video</a>	3
4	<a href="#">Calculatoare</a>	3
5	<a href="#">Jucarii</a>	2

Total produse: 14

## Magazin online Firma X SRL

### Realizati comanda

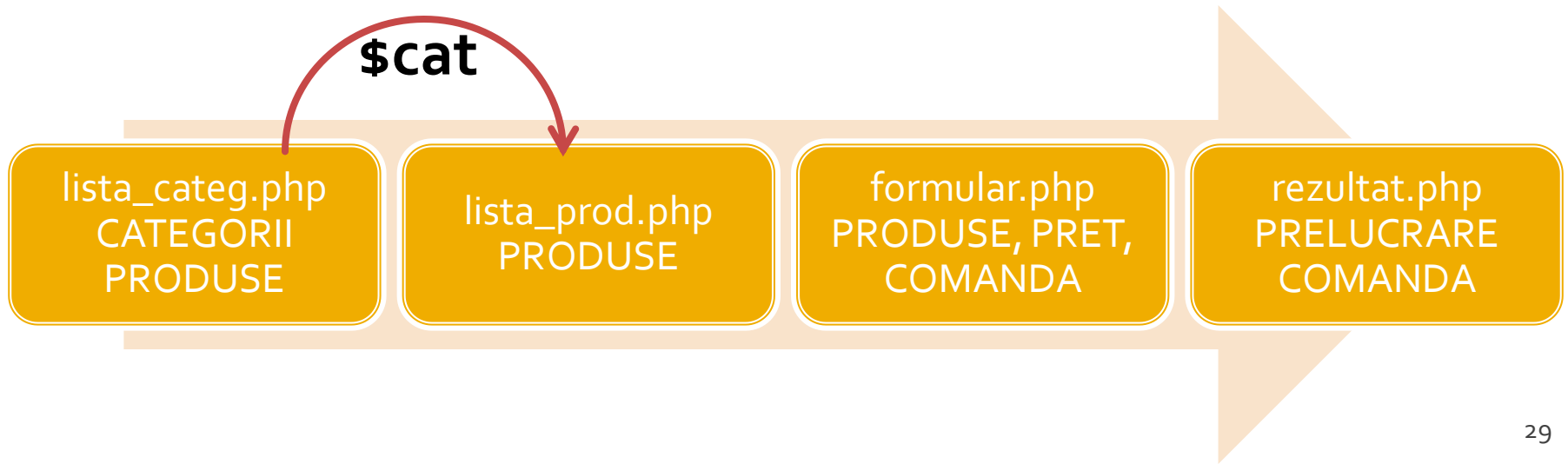
Nr.	Produs	Pret	Cantitate
1	Carti	100	<input type="text" value="1"/>
2	Caiete	50	<input type="text" value="2"/>
3	Penare	150	<input type="text" value="1"/>
4	Stilouri	125	<input type="text" value="0"/>
5	Creioane	25	<input type="text" value="0"/>

# Metode de transmitere

- **post** datele sunt transmise in bloc
- **get** datele sunt atasate adresei documentului de procesare : results.php?prob=81&an=2009
- se poate simula realizarea unei forme (**get**) prin scrierea corespunzatoare a link-urilor

# Transmitere prin GET

- in `lista_categ.php`
  - `<a href="lista_prod.php?categ=?php echo $cat;?>"> <?php echo $cat;?> </a>`
- are efect in `lista_prod.php`
  - `$_GET['categ']="valoarea $cat corespunzatoare"`



# Lista produse

- Nu se afiseaza toate produsele din toate categoriile ci numai produsele din categoria selectata

**Magazin online Firma X SRL**

**Realizati comanda**

Nr.	Produs	Pret	Cantitate
1	Carti	100	<input type="text" value="1"/>
2	Caiete	50	<input type="text" value="2"/>
3	Penare	150	<input type="text" value="1"/>
4	Stilouri	125	<input type="text" value="0"/>
5	Creioane	25	<input type="text" value="0"/>

# Lista produse

- in loc de doua bucle foreach
  - categorii
  - produse
- ramane una singura numai pentru produsele din categoria receptionata

```
$cat=$_GET['categ'];
$index=1;
foreach ($produse[$cat] as $prod => $detalii) //primul indice in $produse[$cat] imi da produsul
{
    <tr><td><?php echo $index;?></td><td><?php echo ucfirst(strtolower($prod));?></td><td><?php echo
    $detalii['descr'];?></td><td align="center"><?php echo $detalii['pret'];?></td></tr>
    <?php $index++;
}
```

# Lista produse din categorii

## Magazin online Firma X SRL

### Realizati comanda

Nr.	Produs	Pret	Cantitate
1	Carti	100	<input type="text" value="1"/>
2	Caiete	50	<input type="text" value="2"/>
3	Penare	150	<input type="text" value="1"/>
4	Stilouri	125	<input type="text" value="0"/>
5	Creioane	25	<input type="text" value="0"/>

Trimite



# XML

- XML - eXtensible Markup Language
- O forma a SGML - Standard Generalized Markup Language (ISO 8879:1986 SGML)
- O metoda de a descrie structura si importanta datelor si continutul lor fara a da indicatii despre afisare
- XSLT - XSL Transformations (Extensible Stylesheet Language) limbaj de conversie a XML in alte tipuri de documente XML cu sau fara reprezentare grafica

# HTML/XHTML vs XML

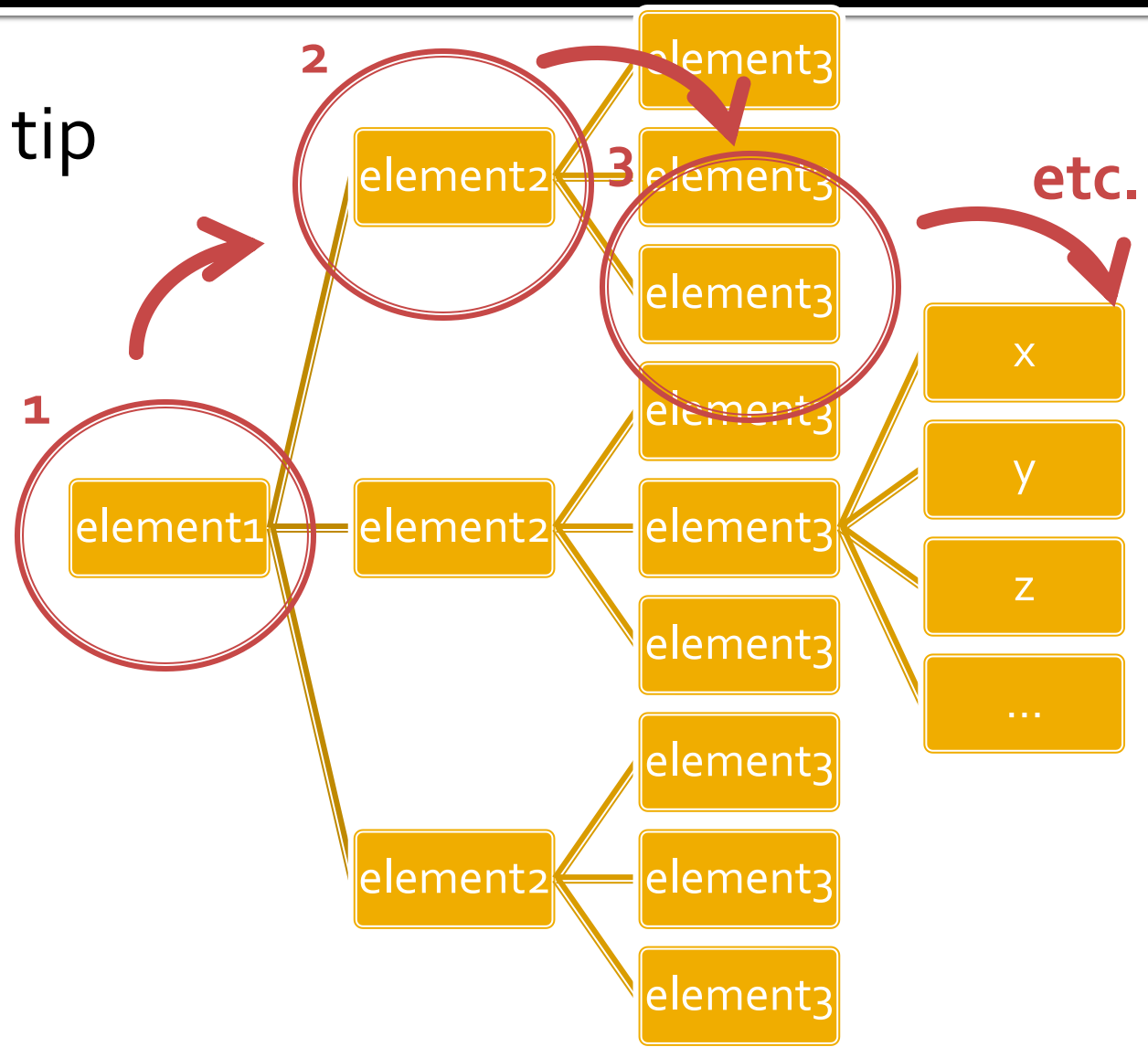
- XML
  - proiectat pentru a **descrie** datele
  - orientat spre **continutul** datelor respective
  - o metoda de a transmite informatiile **independent** de platforma si hardware
- HTML/XHTML
  - proiectat pentru a **afisa** datele
  - orientat spre **forma** pe un ecran a datelor respective
  - o metoda de a **afisa uniform** datele indiferent de platforma si hardware

# XML

- `<element1 atribut1 = "valoare" ... >`
  - `<element2 atribut2 = "valoare">`
    - `<element3>...</element3>`
    - ...
  - `</element2>`
  - ...
- `</element1>`
- ...
- `<element1>`
- `</element1>`

# XML

- structura tip  
"arbore"



# Exemplu XML

```
<?xml version="1.0" encoding="utf-8"?>
<gallery titlu="Photomagic" thumbDir="./fotografii/thumbnails/" imageDir="./fotografii/">
  <category nume="VIATA">
    <image>
      <desc>curiozitate</desc>
      <img>foto33.jpg</img>
      <thumb>foto33TH.jpg</thumb>
    </image>
  </category>
  <category nume="NUNTA">
    <image>
      <desc>asteptare</desc>
      <img>foto132.jpg</img>
      <thumb>foto132TH.jpg</thumb>
    </image>
  </category>
</gallery>
```

Statistică > Seturi de date > Cursul de schimb

**Cursul de schimb**  
Leu (RON)

		13 mar.2017	14 mar.2017	15 mar.2017	16 mar.2017
Dolarul australian	AUD	3,2525	3,2525	3,2494	
Leva bulgărească	BGN	2,3283	2,3272	2,3228	
Dolarul canadian	CAD	3,1725	3,1801	3,1779	

## Link-uri utile

- Serii zilnice - ultimele 10 înregistrări
- Serii lunare, trimestriale și anuale
- Baza de date interactivă (permite vizualizarea și exportul seriilor istorice - zilnice, lunare, anuale)
- Pentru programatori: Preluarea cursului valutar prin formate XML

Statistică > Seturi de date > Cursul de schimb > Cursurile în format XML

## Cursurile pieței valutare în format XML

Pentru diseminarea cursurilor valutare, BNR pune la dispoziția programatorilor fișiere de date în format XML. Actualizarea acestora se face în timp real, în fiecare zi bancară, imediat după ora 13:00.

```
<?xml version="1.0" encoding="UTF-8"?>
<DataSet xsi:schemaLocation="http://www.bnr.ro/xsd nbrfxrates.xsd">
  <Header>
    <Publisher>National Bank of Romania</Publisher>
    <PublishingDate>2017-03-17</PublishingDate>
    <MessageType>DR</MessageType>
  </Header>
  <Body>
    <Subject>Reference rates</Subject>
    <OrigCurrency>RON</OrigCurrency>
    <Cube date="2017-03-17">
      <Rate currency="AED">1.1542</Rate>
      <Rate currency="AUD">3.2609</Rate>
      <Rate currency="BGN">2.3342</Rate>
      <Rate currency="BRL">1.3600</Rate>
      <Rate currency="CAD">3.1807</Rate>
      <Rate currency="CHF">4.2598</Rate>
      <Rate currency="CNY">0.6140</Rate>
      <Rate currency="CZK">0.1690</Rate>
      <Rate currency="DKK">0.6141</Rate>
      <Rate currency="EGP">0.2335</Rate>
```

Conținutul sau structura acestora se poate modifica fără a fi necesară o notificare prealabilă. IP-urile care generează un volum de trafic excesiv sau care cauzează trafic redundant, IP-urile care generează un volum de trafic excesiv vor fi blocate temporar pentru remedierea situației pe adresa webmaster.

Se permite accesul la informații autorizate să efectueze operațiuni pe piața valutară, operațiuni efective de schimb valutar sau în înregistrări contabile.

## FIȘIERE DE DATE ÎN FORMAT XML

- Seturi de înregistrări zilnice:
  - cea mai recentă
  - ultimele 10 înregistrări
- Seturi de înregistrări anuale:
  - 2017
  - 2016
  - 2015
  - 2014
  - 2013
  - 2012
  - 2011
  - 2010
  - 2009
  - 2008
  - 2007
  - 2006
  - 2005

# Acces prin cod PHP la date XML

- **Biblioteci** PHP de acces la XML
  - DOM XML
  - XML Parser (Reader, Writer)
  - Simple XML

# Simple XML

- `$xml = simplexml_load_file('cale');`
- `$xml = simplexml_load_string($string);`
- Functiile care creaza un obiect din clasa predefinita **SimpleXMLElement** prin incarcarea si interpretarea documentului sau sirului XML
- obiectul creat (`$xml`) contine elementul radacina
  - exemplul anterior → `$xml` va contine **`<gallery></gallery>`**



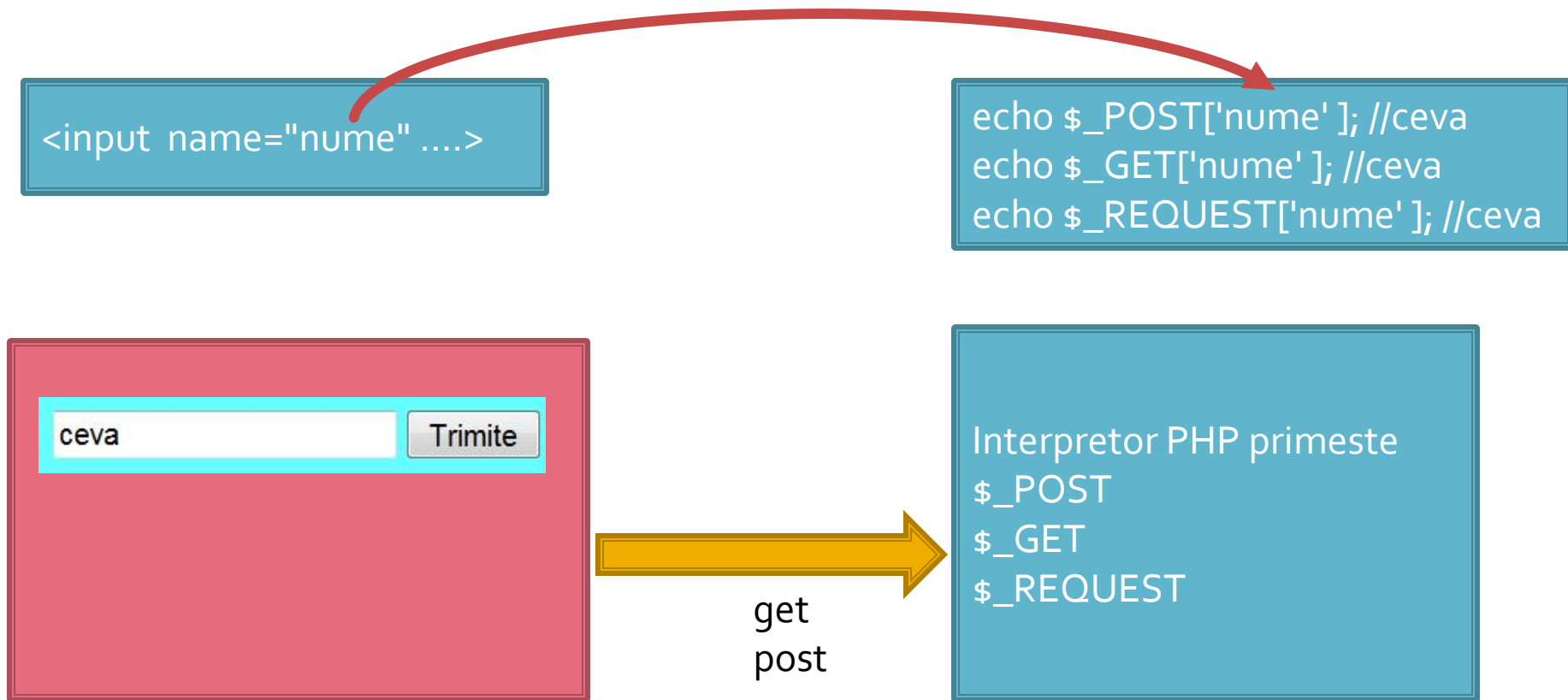
# Modificare de cod XML

- incepand cu PHP 5.1.3.
- SimpleXMLElement::addChild(string nume, string valoare)
- SimpleXMLElement::addAttribute(string nume, string valoare)
- SimpleXMLElement::asXML('filename') scrie in fisierul 'filename' rezultatul sau il returneaza ca un sir
- schimbarea continutului unui element
  - `$xml->category[0]->image[0]->img = "foto33.jpg"`

# Laborator 5 – Mod de lucru

- 1. Se introduce in surse facilitatea template
- 2. Se modifica sursele pentru lucru cu matrici
- 3. Se modifica sursele pentru a citi datele de pe disc
  - **anterior** se creaza fisierul text sau:
  - **o singura data** se salveaza datele (S16)
- 4. Se introduce structura suplimentara, categorie
  - se **creaza pagina** de selectie a categoriei, din care se va merge in lista de produse (utilizare \$\_GET – S29)
- 5. Optional: Se creaza o pagina care sa permita modificarea fisierului
  - numai pret/descriere, fara adaugare/schimbare produse
- 6. Optional (secundar): se schimba codul pentru a lucra cu fisier de date in format **XML**

# Client / Server



# Depanare

```
echo "<pre>";  
print_r($_POST);  
echo "</pre>";
```

```
<p>temp <?php echo  
"a=";echo $a; ?> </p>
```

# Contact

- Laboratorul de microunde si optoelectronica
- <http://rf-opto.etti.tuiasi.ro>
- [rdamian@etti.tuiasi.ro](mailto:rdamian@etti.tuiasi.ro)