

Curs 7

2019/2020

Programarea aplicațiilor web

- Programarea aplicațiilor web
 - An V RC
 - 2C/1L/1P

Nota

- An V
 - ~~33%~~ E: 40%
 - ~~66%~~ Aplicatii
 - ~~33%~~ L (0%)
 - ~~33%~~ P (60%)

Memorarea datelor

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

```
$produse = array ( 'carte' => array ("descr" => "mai multe pagini scrise legate", "pret" => 100, "cant" => 0),  
    'caiet' => array ("descr" => "mai multe pagini goale legate", "pret" => 50, "cant" => 0),  
    'penar' => array ("descr" => "loc de depozitat instrumente de scris", "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");
```

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

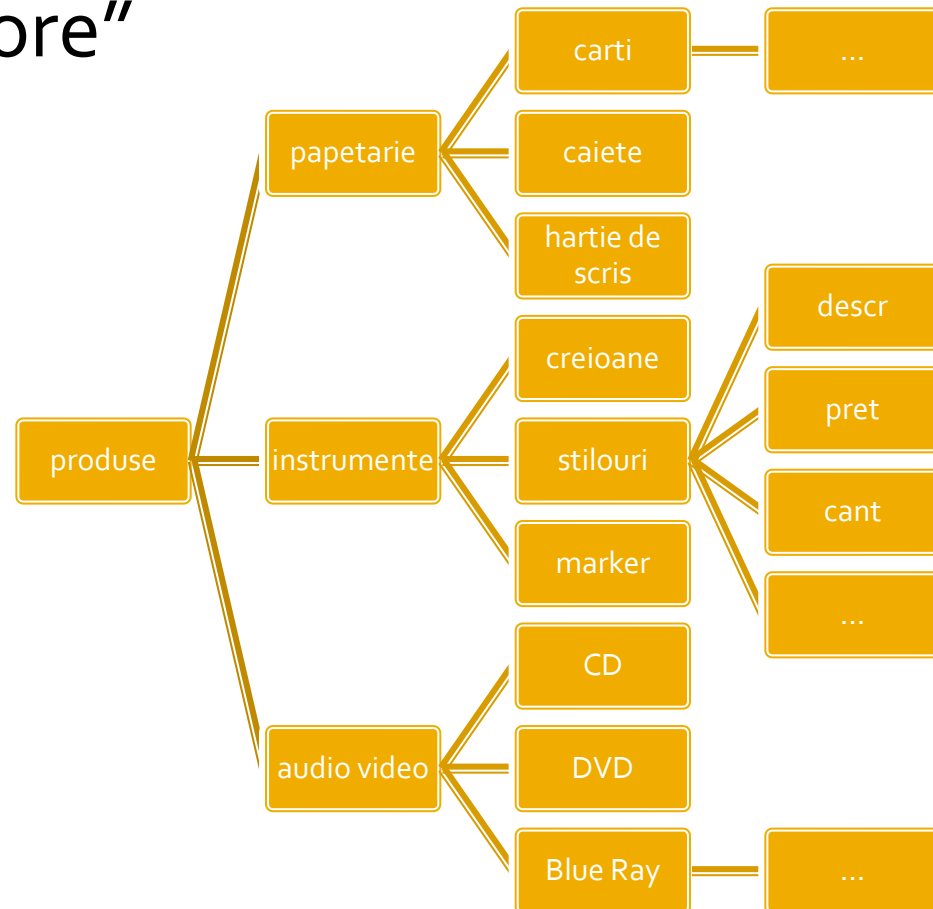
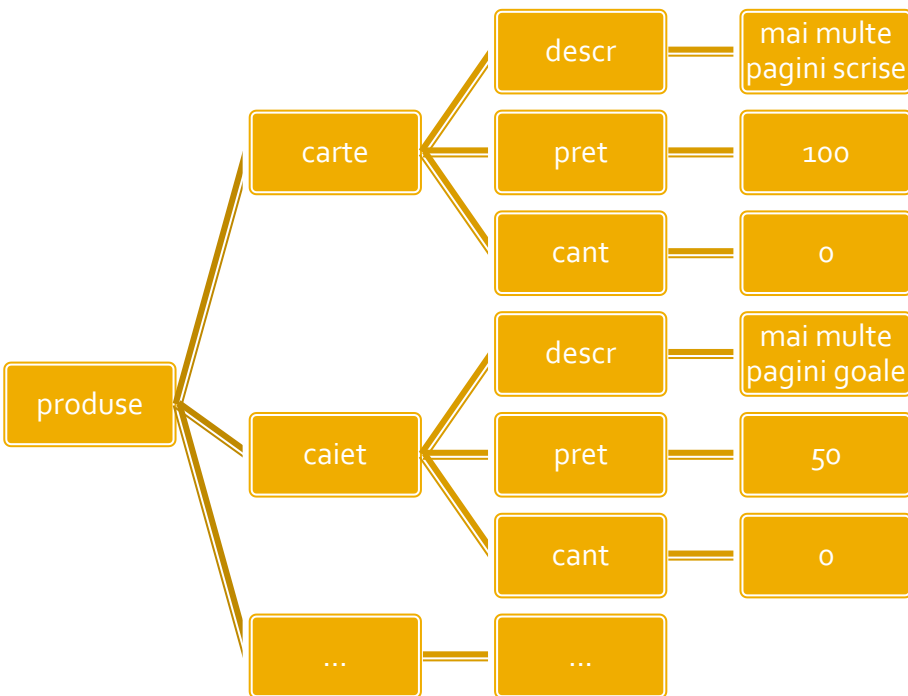
carte	mai multe pagini scrise legate	100	0	
caiet	mai multe pagini goale legate	75	0	
penar	loc de depozitat instrumente de scris		150	0
stilou	instrument de scris albastru	125	0	
creion	instrument de scris gri	25	0	

Scrierea datelor pe disc

- Structura datelor este total dependentă de algoritmul utilizat
- Modificările sunt dificil de realizat și implică schimbarea fisierului de date și schimbarea codului în întregime
- Exemplu: gruparea elementelor pe categorii
 - `$produse["produs"] = ["caract."]`
 - `$produse["categorie"]["produs"] = ["caract."]`
- index-ul suplimentar introdus transformă o matrice bidimensională în una tridimensională
 - dificil de implementat cu fișiere text (caract/linii – 2D)

Matrici

- adaugare nivel in "arbore"



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

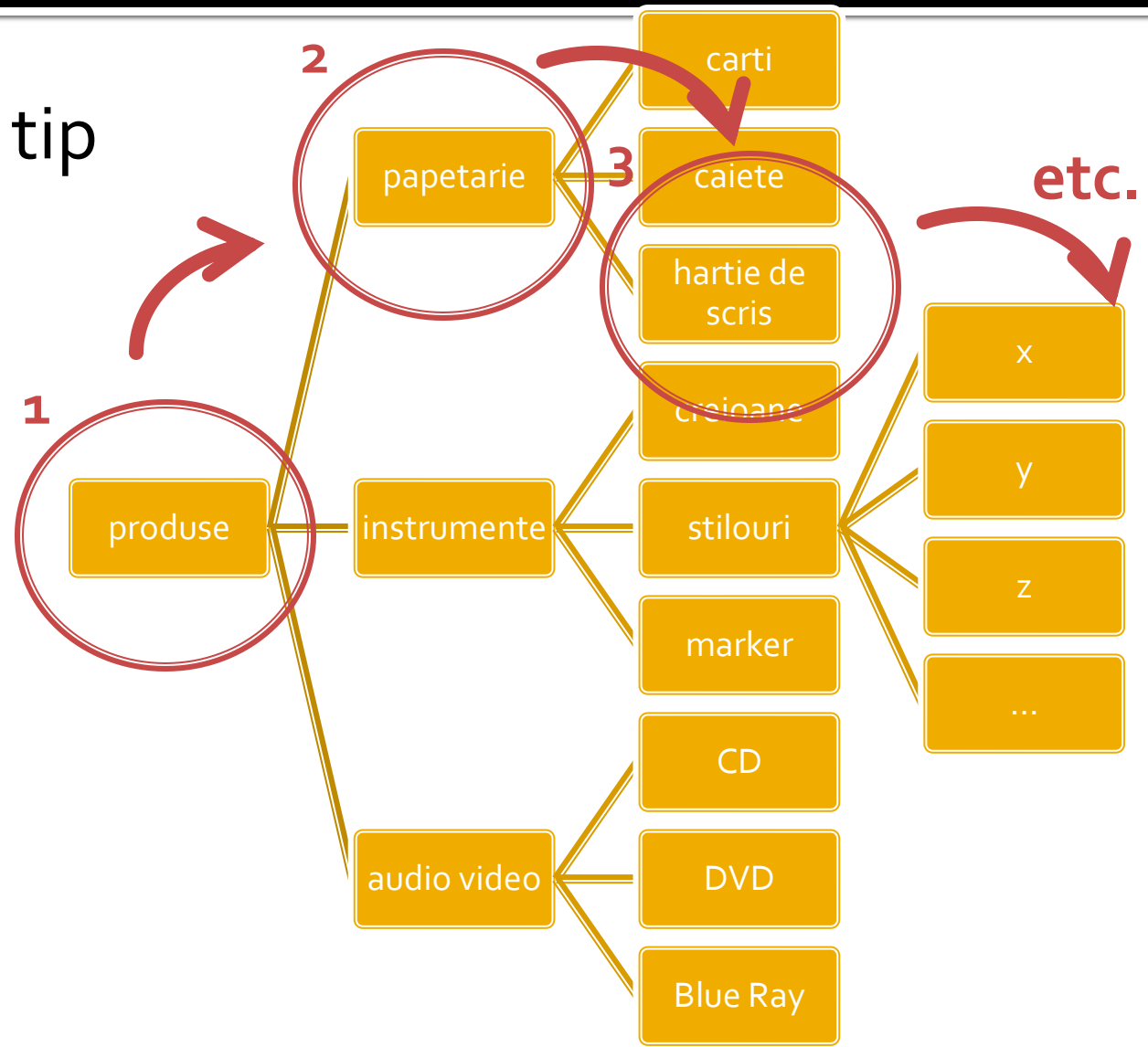
Lista Produse

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

[Comanda](#)

Matrici

- structura tip "arbore"



Accesul la metode externe de stocare eficiente a datelor

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

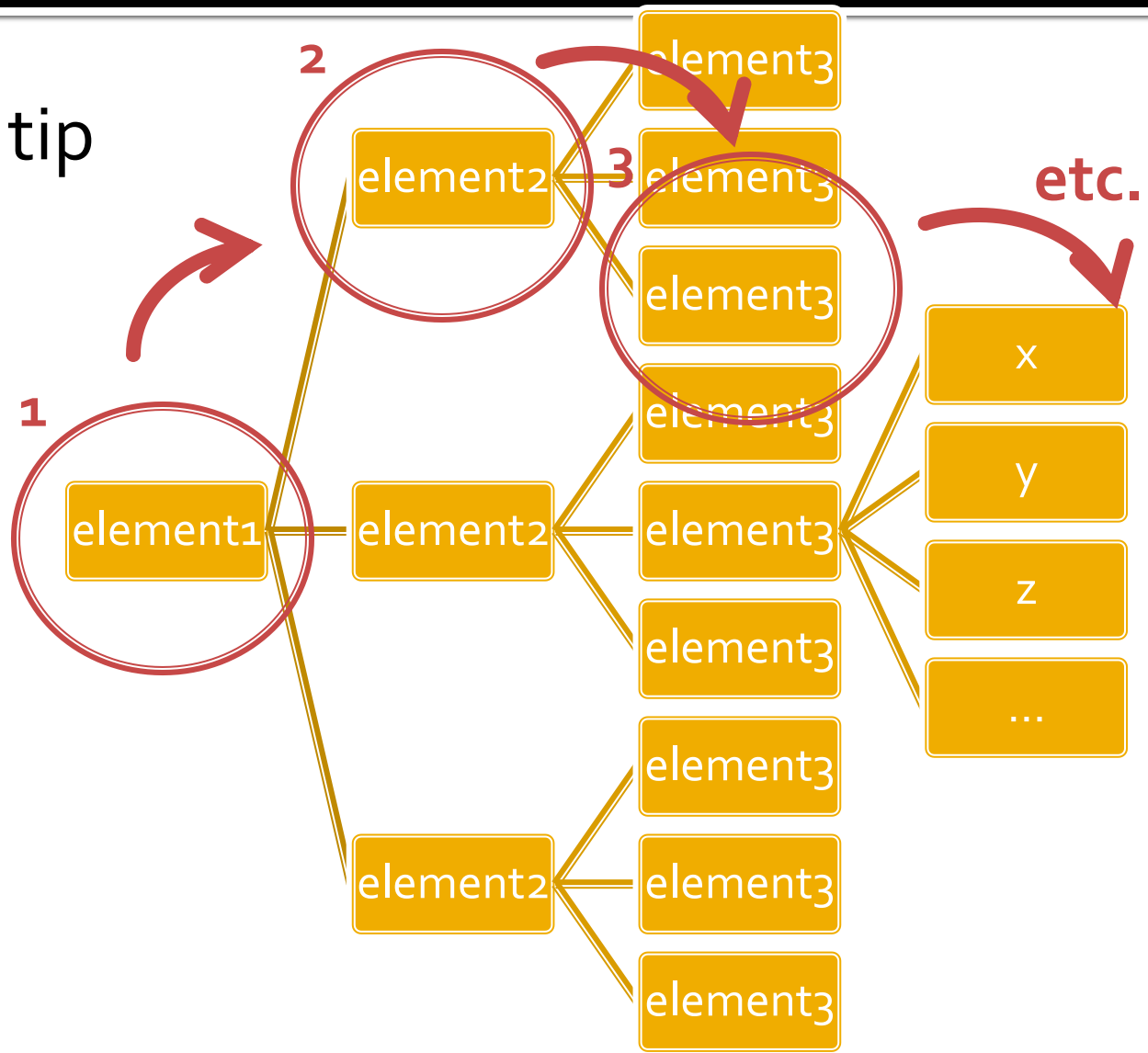
- In conceptie asemanator cu XHTML
 - etichete XHTML ("tag" - EN)
 - elemente XML ("element" - EN) descrise de etichete ("tag" - EN)
- Elementele XML accepta attribute (similar XHTML)
- Conceptele de scriere a documentului similar XHTML
- Diferenta majora:
 - HTML – etichetele si attributele sunt predefinite si orientate spre modalitatea de afisare a datelor
 - XML – etichetele de identificare a elementelor si attributele sunt la latitudinea creatorului documentului, introducand **structura** in date

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>
```

Avantaje

- **Redundanta**
 - fiecare element XML trebuie introdus complet
 - aceasta permite detectia si corectarea facila a erorilor
- **Auto descriptiv**
 - XML este un limbaj bazat pe text, insesi elementele si attributele descriu datele
 - usor de citit/corectat pentru utilizatori umani
- **Generalitate**
 - orice fisier XML poate fi citit de orice aplicatie XML
 - anumite aplicatii necesita o anumita structura a datelor

Reguli XML

- Aproape orice caracter UNICODE poate fi utilizat
- 107000 caractere, 90 scrieri diferite
- exceptii:
 - < <
 - > >
 - & &
 - " "
 - ' `

Reguli XML

- etichetele de definire a elementelor **trebuie** închise sau elementul declarat ca vid
 - `<descriere> ... </descriere>`
 - `<descriere></descriere>`
 - `<descriere />`
- atributele **trebuie** scrise între ghilimele
 - `<categorie nume="papetarie">`
- etichetele și atributele sunt **Case Sensitive**
 - **gresit (tehnic)** -> `<descriere> ... </Descriere>`
 - **gresit (logic)** -> `<descriere> ... </descriere>`
`<DESCRIERE> ... </DESCRIERE>`

Reguli XML

- Documentele XML creaza o structura ierarhica foarte stricta
- Nu sunt permise etichete suprapuse
 - `<x><y></y></x>` → permis
 - `<x><y></x></y>` → interzis
- Trebuie sa existe un singur element radacina care sa le cuprinda pe toate celelalte
 - similar cu `<html></html>`

Structura unui document XML

- prima linie – definitia tipului de document
 - `<?xml ... ?>`
 - `<?xml version="1.0" encoding="utf-8"?>`
- element radacina
 - `<radacina> ...[elemente]... </radacina>`

XML Concepte

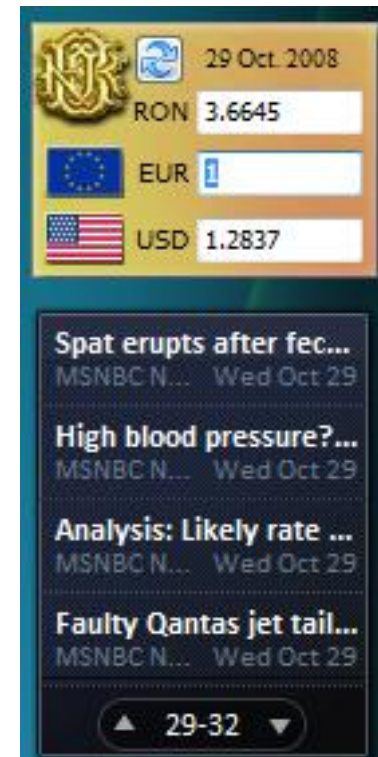
- comentariile pot fi introduse oriunde in interiorul documentului cu conditia sa fie in exteriorul oricarui element
 - similare cu comentariile HTML: intre `<-- si -->`
- Sectiuni de date neinterpretate
 - intre `<![CDATA[si]]>`
 - pentru a putea introduce date care ar putea contine caracterele interzise
 - cod
 - date binare oarecare

Exemple XML


RSS

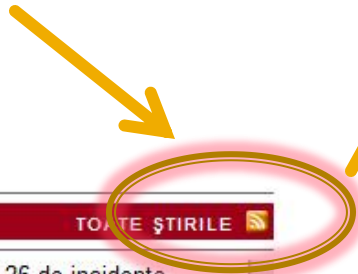


- RSS – Really Simple Syndication
- Format Web utilizat pentru a face disponibile utilizatorilor sau altor server-e informatii frecvent modificate
 - stiri
 - postari pe forum



RSS

- Accesul la fluxurile de date
 - identificarea canalelor 
 - inscrierea la canal ("Subscribe to this feed")
 - aplicatie pentru afisarea informatiilor



The screenshot shows a web browser window with several tabs. The active tab is 'Adevarul.ro'. The main content area displays a news article with a 'Subscribe to this feed' button circled in red. An arrow points from the RSS icon to this button. Another arrow points from the button to a sidebar widget on the right. The sidebar widget displays a calendar for November 2009, showing the 23rd as Monday. Above the calendar, there are several news snippets, including 'Antonescu: "Institute...', 'Geoană a mulțumit ...', 'Românii își doresc p...', and 'Băsescu: "Încă nu a...'. The bottom of the page shows a news ticker with the following items:

- 08:42 | Poliția a înregistrat 26 de incidente electorale în județ-Revista presei locale
- 08:41 | Rezultatele etapei a 13-a din Italia

BNR Politică monetară Stabilitate financiară Supraveghere Sisteme de plăți Monede și bancnote Publicații **Statistică** Legislație

Seturi de date
Cursul de schimb
Serii statistice
Info financiar
Rate dobânzi de politică monetară
Operațiuni de piață monetară
Titluri de stat
Statistică monetară și financiară
Raport inflație
Rapoarte asupra stabilității financiare

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 austriac	AUD	3,2325	3,2325	3,2434	3,2434
 Leva bulgărească	BGN	2,3283	2,3272	2,3228	2,3228
 Dolarul canadian	CAD	3,1725	3,1801	3,1779	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

Seturi de date
Cursul de schimb
Serii statistice
Info financiar
Rate dobânzi de politică monetară
Operațiuni de piață monetară
Titluri de stat
Statistică monetară și financiară
Raport inflație
Rapoarte asupra stabilității financiare
Conturile naționale financiare
Indicatori plăți
Balanța de plăți - BPM6
Finanțe publice
Situatiile financiare ale BNR
Bază de date interactivă
Centrala Incidentelor de Plăți (CIP)
Centrala Riscului de Credit (CRC)
SDDS
Raportări către BNR

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>
```

SERII DE DATE ÎN FORMAT XML

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

XML in PHP

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>`**

Simple XML

- fiecare **tip** de element descendent creaza o proprietate a clasei cu acel nume
 - proprietatea clasei (descendent) e o matrice de obiecte **SimpleXMLElement** daca nu e unicul descendent
 - matricea e caracterizata de elementele implicite de indexare (index intreg, pornind de la 0) dar e recomandata iteratia folosind `foreach()`

Exemplul anterior

- elementul <gallery> contine un numar mare de descendenti de un singur tip, <category>
- obiectul radacina a fost redenumit prin operatia de deschidere a fisierului
 - `$xml = simplexml_load_file('cale');`
- obiectul `$xml` va contine ca proprietate
 - `$xml->category`
 - `$xml->category` este o matrice deoarece nu este o singura categorie
 - descendentii se pot accesa cu
 - `$xml->category[0], $xml->category[1], ...`
 - `foreach ($xml->category as $categ_curent) { // $categ_curent obiect de tip SimpleXMLElement, aceleasi reguli }`

Simple XML

- attributele XML pot fi accesate in obiectul corespunzator ca si cum ar fi elementele unei matrici cu indice text, indicele fiind acelasi cu numele atributului cautat
- In exemplul anterior fiecare element `<category>` are un atribut `nume`
 - `<category nume="VIATA">`
 - ```
foreach ($xml->category as $categ_curent)
{
 echo $categ_curent['nume'];
}
```

# Recursivitate

- Fiecare proprietate a clasei **SimpleXMLElement** este de asemenea un obiect din clasa **SimpleXMLElement** daca mai are descendenti
  - `$xml->category[0]` este de tip obiect:  
`$xml->category[0]->image ...`
- Proprietatea clasei contine datele elementului daca nu mai sunt alti descendenti
  - `echo $xml->category[0]->image[0]->img // foto33.jpg`
- Pentru ca proprietatea sa fie tratata ca fiind de tip string (implicit la afisare) trebuie fortata conversia
  - `if ((string)$xml->category[0]->image[0]->img=="foto33.jpg") ...`

# 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"`

# Contact

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