

Laborator 07 - Extragerea datelor dintr-un document XML (PHP->XML), partea a treia

Cuprins

[Utilizarea DOM pentru citirea de documente](#)

[Exemplu de utilizare a DOM pentru încărcarea unui document XML dintr-un fișier și afișarea sa integrală în browser](#)

[Exemplu de utilizare a DOM pentru încărcarea unui document XML dintr-un fișier și parcurgerea sa nod cu nod](#)

[SAX Parser](#)

[Exemplu de utilizare a SAX Parser pentru citirea unui document XML](#)

În laboratorul precedent am văzut cum poate fi citit un document XML utilizând bibliotecile SimpleXML și XML Expat Parser. Laboratorul de față conține utilizarea bibliotecii DOM pentru aceleași operații.

Utilizarea DOM pentru citirea de documente

Parserul DOM este un parser bazat pe arbori (încarcă tot documentul XML într-o structură stocată în memoria RAM iar ulterior structura respectivă poate fi parcursă în mod iterativ de așa natură încât să se poată obține informația din oricare nod / toate nodurile documentului XML).

Exemplu de utilizare a DOM pentru încărcarea unui document XML dintr-un fișier și afișarea sa integrală în browser

Pregătiți în prealabil un fișier amplasat în aceeași locație cu script-ul PHP și denumit note.xml care să conțină documentul XML (acest document ar putea să existe deja de la laboratoarele precedente):

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

Scriptul PHP corespunzător este:

```
<?php
//Crearea obiectului în memorie și încărcarea datelor din fișier în obiect
$xmlDoc = new DOMDocument();
$xmlDoc->load("note.xml");

//Afișarea conținutului obiectului în browser
print $xmlDoc->saveXML();
```

?>

Obs. Dând clic dreapta oriunde pe zona de afișare din browser și selectând opțiunea View Source se poate observa că sursa a ceea ce este afișat în browser este documentul XML original care a fost încărcat din fișier. Tragem concluzia ca browser-ele obișnuite sunt capabile să interpreteze nu numai HTML ci și XML (de fapt cele două limbaje sunt apropiate conceptual).

Exemplu de utilizare a DOM pentru încărcarea unui document XML dintr-un fișier și parcurgerea sa nod cu nod

Pregătiți în prealabil un fișier amplasat în aceeași locație cu script-ul PHP și denumit books.xml care să conțină documentul XML (acest document ar putea să existe deja de la laboratoarele precedente):

```
<?xml version="1.0" encoding="utf-8"?>
<bookstore>
  <book category="COOKING">
    <title lang="en">Everyday Italian</title>
    <author>Giada De Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>
  <book category="CHILDREN">
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
  <book category="WEB">
    <title lang="en-us">XQuery Kick Start</title>
    <author>James McGovern</author>
    <year>2003</year>
    <price>49.99</price>
  </book>
  <book category="WEB">
    <title lang="en-us">Learning XML</title>
    <author>Erik T. Ray</author>
    <year>2003</year>
    <price>39.95</price>
  </book>
</bookstore>
```

Scriptul PHP corespunzător este:

```

<?php
//Crearea obiectului în memorie și încărcarea datelor din fișier în obiect
$xmlDoc = new DOMDocument();
$xmlDoc->load("books.xml");

//Extragerea conținutului din documentul stocat în memorie
$x = $xmlDoc->documentElement;
//Parcurgerea iterativă a conținutului, cu afișare
foreach ($x->childNodes AS $item)
{
    print $item->nodeName . " = " . $item->nodeValue . "<br>";
}
?>

```

SAX Parser

SAX Parser este un parser bazat pe evenimente. Principalul avantaj oferit este faptul că nu stochează datele în memorie putând fi folosit pentru citirea de documente XML de dimensiuni foarte mari.

Exemplu de utilizare a SAX Parser pentru citirea unui document XML

Pregătiți în prealabil un fișier amplasat în aceeași locație cu script-ul PHP și denumit sax.xml care să conțină documentul XML:

```

<?xml version = "1.0" encoding = "utf-8"?>
<tutors>
    <course>
        <name>Android</name>
        <country>India</country>
        <email>contact@tutorialspoint.com</email>
        <phone>123456789</phone>
    </course>

    <course>
        <name>Java</name>
        <country>India</country>
        <email>contact@tutorialspoint.com</email>
        <phone>123456789</phone>
    </course>

```

```

<course>
    <name>HTML</name>
    <country>India</country>
    <email>contact@tutorialspoint.com</email>
    <phone>123456789</phone>
</course>
</tutors>

```

Scriptul PHP corespunzător este:

```

<?php

$tutors = array();
$elements = null;

//Functie apelata la deschiderea unui tag
function startElements($parser, $name, $attrs)
{
    global $tutors, $elements;

    if(!empty($name)) {
        if ($name == 'COURSE') {
            //Crearea matricii pentru stocarea informatiei
            $tutors []= array();
        }
        $elements = $name;
    }
}

//Functie apelata la inchiderea unui tag
function endElements($parser, $name)
{
    global $elements;

    if(!empty($name)) {
        $elements = null;
    }
}

//Functie apelata pe textul dintre deschiderea si inchiderea unui tag
function characterData($parser, $data)
{
    global $tutors, $elements;

    if(!empty($data)) {
        if ($elements == 'NAME' || $elements == 'COUNTRY' || $elements == 'EMAIL' || $elements ==
'PHONE')

```

```

        {
            $tutors[count($tutors)-1][$elements] = trim($data);
        }
    }
}

//Creaza un parser XML nou si returneaza o referinta la resursa creata de acest parser
$xml_parser = xml_parser_create();

xml_set_element_handler($xml_parser, "startElements", "endElements");
xml_set_character_data_handler($xml_parser, "characterData");

//Deschide fisierul XML
if (!($handle = fopen('sax.xml', "r")))
{
    die("Nu am putut deschide documentul XML.");
}
else
{
    echo "Am deschis documentul XML. Acest document contine: <br>";
}

//Citirea bucata cu bucata a datelor din fisier
while($data = fread($handle, 4096))
{
    //Parsarea documentului XML
    xml_parse($xml_parser, $data);
}

//Eliberarea memoriei ocupate de parser
xml_parser_free($xml_parser);
$i = 1;

//Afisarea datelor
foreach($tutors as $course)
{
    echo "course No - ".$i."<br/>";
    echo "course Name - ".$course['NAME']."<br/>";
    echo "Country - ".$course['COUNTRY']."<br/>";
    echo "Email - ".$course['EMAIL']."<br/>";
    echo "Phone - ".$course['PHONE']."<br/>";
    $i++;
}
?>

```