**PHP XML Tutorial: Create, Parse, Read with Example**

**What is XML?**

XML is the acronym for Extensible Markup Language.

XML is used to structure, store and transport data from one system to another.

XML is similar to HTML.

It uses opening and closing tags.

Unlike HTML, XML allows users to define their own tags.

In this tutorial, you will learn-

* [What is DOM?](https://www.guru99.com/php-and-xml.html#2)
* [XML Parsers](https://www.guru99.com/php-and-xml.html#3)
* [Why use XML?](https://www.guru99.com/php-and-xml.html#4)
* [XML Document Example](https://www.guru99.com/php-and-xml.html#5)
* [How to Read XML using PHP](https://www.guru99.com/php-and-xml.html#6)
* [How to Create an XML document using PHP](https://www.guru99.com/php-and-xml.html#8)

**What is DOM?**

DOM is the acronym for Document Object Model.

It’s a cross platform and language neutral standard that defines how to access and manipulate data in;

* HTML
* XHTML
* XML

DOM XML is used to access and manipulate XML documents. It views the XML document as a tree-structure.

**XML Parsers**

An XML parser is a program that translates the XML document into an XML Document Object Model (DOM) Object.

The XML DOM Object can then be manipulated using JavaScript, Python, and PHP etc.

The keyword CDATA which is the acronym for (Unparsed) Character Data is used to ignore special characters such as “<,>” when parsing an XML document.

**Why use XML?**

* Web services such as SOAP and REST use XML format to exchange information. Learning what XML is and how it works will get you competitive advantage as a developer since modern applications make heavy use of web services.
* XML documents can be used to store configuration settings of an application
* It allows you to create your own custom tags which make it more flexible.

**XML Document example**

Let’s suppose that you are developing an application that gets data from a web service in XML format.

Below is the sample of how the XML document looks like.

<?xml version="1.0" encoding="utf-8"?>

<employees status = "ok">

        <record man\_no = "101">

            <name>Joe Paul</name>

            <position>CEO</position>

        </record>

        <record man\_no = "102">

            <name>Tasha Smith</name>

            <position>Finance Manager</position>

        </record>

</employees>

  HERE,

* “<?xml version="1.0" encoding="utf-8"?>” specifies the xml version to be used and encoding
* “<employees status = "ok">” is the root element.
* “<record…>…</record>” are the child elements of administration and sales respectively.

**How to Read XML using PHP**

Let’s now write the code that will read the employees XML document and display the results in a web browser. *Index.php*

<?php

$xml = simplexml\_load\_file('employees.xml');

echo '<h2>Employees Listing</h2>';

$list = $xml->record;

for ($i = 0; $i < count($list); $i++) {

    echo '<b>Man no:</b> ' . $list[$i]->attributes()->man\_no . '<br>';

    echo 'Name: ' . $list[$i]->name . '<br>';

    echo 'Position: ' . $list[$i]->position . '<br><br>';

}

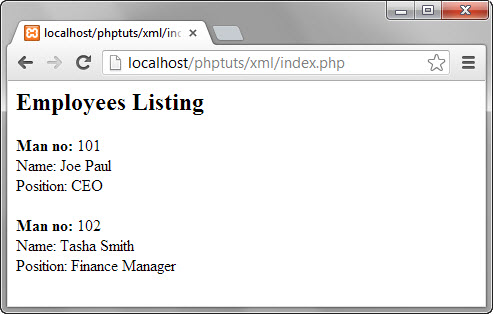
?>

  HERE,

* “$xml = simplexml\_load\_file('employees.xml');” uses the simplexml\_load\_file function to load the file name employees.xml and assign the contents to the array variable $xml.
* “$list = $xml->record;” gets the contents of the record node.
* “for ($i = 0; $i < count(…)…” is the for loop that reads the numeric array and outputs the results
* “$list[$i]->attributes()->man\_no;” reads the man\_no attribute of the element
* “$list[$i]->name;” reads the value of the name child element
* “$list[$i]->position;” reads the value of the position child element

**Testing our application**

Assuming you saved the file index.php in phptus/xml folder, browse to the URL**http://localhost/phptuts/xml/index.php**

[](https://www.guru99.com/images/2013/04/employees_listing.jpg)

**How to Create an XML document using PHP**

We will now look at how to create an XML document using PHP.

We will use the example above in the DOM tree diagram.

The following code uses the PHP built in class DOMDocument to create an XML document.

<?php

$dom = new DOMDocument();

$dom->encoding = 'utf-8';

$dom->xmlVersion = '1.0';

$dom->formatOutput = true;

$xml\_file\_name = 'movies\_list.xml';

$root = $dom->createElement('Movies');

$movie\_node = $dom->createElement('movie');

$attr\_movie\_id = new DOMAttr('movie\_id', '5467');

$movie\_node->setAttributeNode($attr\_movie\_id);

$child\_node\_title = $dom->createElement('Title', 'The Campaign');

$movie\_node->appendChild($child\_node\_title);

$child\_node\_year = $dom->createElement('Year', 2012);

$movie\_node->appendChild($child\_node\_year);

$child\_node\_genre = $dom->createElement('Genre', 'The Campaign');

$movie\_node->appendChild($child\_node\_genre);

$child\_node\_ratings = $dom->createElement('Ratings', 6.2);

$movie\_node->appendChild($child\_node\_ratings);

$root->appendChild($movie\_node);

$dom->appendChild($root);

$dom->save($xml\_file\_name);

echo "$xml\_file\_name has been successfully created";

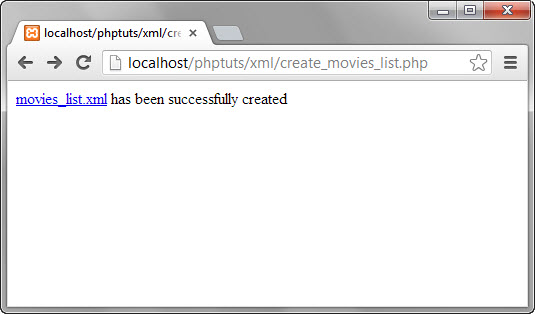
?>

  HERE,

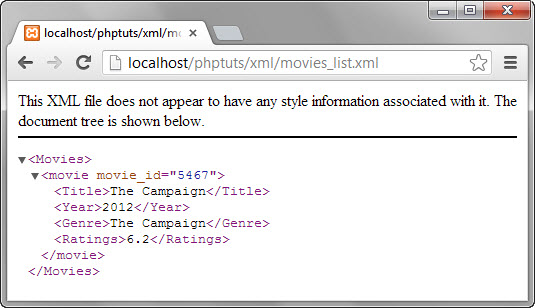
* “$dom = new DOMDocument();” creates an instance of DOMDocument class.
* “$dom->encoding = 'utf-8';” sets the document encoding to utf-8
* “$dom->xmlVersion = '1.0';” specifies the version number 1.0
* “$dom->formatOutput = true;” ensures that the output is well formatted
* “$root = $dom->createElement('Movies');” creates the root node named Movies
* “$attr\_movie\_id = new DOMAttr('movie\_id', '5467');” defines the movie id attribute of Movies node
* “$child\_node\_element\_name = $dom->createElement('ElementName', 'ElementValue')” creates the child node of Movies node. ElementName specifies the name of the element e.g. Title. ElementValue sets the child node value e.g. The Campaign.
* “$root->appendChild($movie\_node);” appends the movie\_node elements to the root node Movies
* “$dom->appendChild($root);” appends the root node to the XML document.
* “$dom->save($xml\_file\_name);” saves the XML file in the root directory of the web server.
* “echo '<a href= "'.$xml\_file\_name.'">' . $xml\_file\_name . '</a> has been successfully created';” creates the link to the XML file.

**Testing our application**

Assuming you saved the file create\_movies\_list in phptuts/xml folder, browse to the URL **http://localhost/phptuts/xml/create\_movies\_list.php**

[](https://www.guru99.com/images/2013/04/create_xml.jpg)

Click on movies\_list\_xml link

[](https://www.guru99.com/images/2013/04/view_xml.jpg)

**Summary**

* XML is the acronym for Extensible Markup Language
* XML can be used for exchanging information between systems or store configuration settings of an application etc.
* DOM is the acronym for Document Object Model. XML DOM views the XML document as a tree-structure
* An XML Parser is a program that translates XML an XML document into a DOM tree-structure like document.
* CDATA is used to ignore special characters when parsing XML documents.
* PHP uses the simplexml\_load\_file to read XML documents and return the results as a numeric array
* PHP DOMDocument class to create XML files.