

## Assignment No 2

Set A :

1) Write a script to create XML file named "Item.xml"

```
<Item>
    <ItemName>..... </ItemName>
    <ItemPrice>..... </ItemPrice>
    <Quantity>..... </Quantity>
</Item>
```

Store the details of 5 Items of different Types

=>

i) Item.php

```
<?php
```

```
$items = [
    ["Laptop", 800, 10],
    ["Smartphone", 500, 20],
    ["Tablet", 300, 15],
    ["Headphones", 100, 25],
    ["Smartwatch", 200, 30]
];
```

```
$xml = new SimpleXMLElement("<Items/>");
```

```
foreach ($items as $item) {
    $itemElement = $xml->addChild("Item");
    $itemElement->addChild("ItemName", $item[0]);
    $itemElement->addChild("ItemPrice", $item[1]);
    $itemElement->addChild("Quantity", $item[2]);
}
```

```
$xml->asXML("Item.xml");
echo "Item.xml file created successfully!";
?>
```

## OUTPUT :

<http://localhost/webPractical%20Ty%20Sem%206/Item.php>

Item.xml file created successfully!

**ii) Item.xml** ( this file are automatically created )

```
<?xml version="1.0" encoding="UTF-8"?>
<Items>
  <Item>
    <ItemName>Laptop</ItemName>
    <ItemPrice>800</ItemPrice>
    <Quantity>10</Quantity>
  </Item>
  <Item>
    <ItemName>Smartphone</ItemName>
    <ItemPrice>500</ItemPrice>
    <Quantity>20</Quantity>
  </Item>
  <Item>
    <ItemName>Tablet</ItemName>
    <ItemPrice>300</ItemPrice>
    <Quantity>15</Quantity>
  </Item>
  <Item>
    <ItemName>Headphones</ItemName>
    <ItemPrice>100</ItemPrice>
    <Quantity>25</Quantity>
  </Item>
  <Item>
    <ItemName>Smartwatch</ItemName>
    <ItemPrice>200</ItemPrice>
    <Quantity>30</Quantity>
  </Item>
</Items>
```

**2. Link “ Item. Xml” file to the CSS style sheet and get well formatted output as given below**

**i) ItemName :**

**Color : red; Font-family : copperplate Gothic Light;  
Font-size : 16pt; Font :bold;**

**ii) ItemPrice and Quantity**

**color:yellow; fontfamily:Arial**

**=>**

**i) Item.xml**

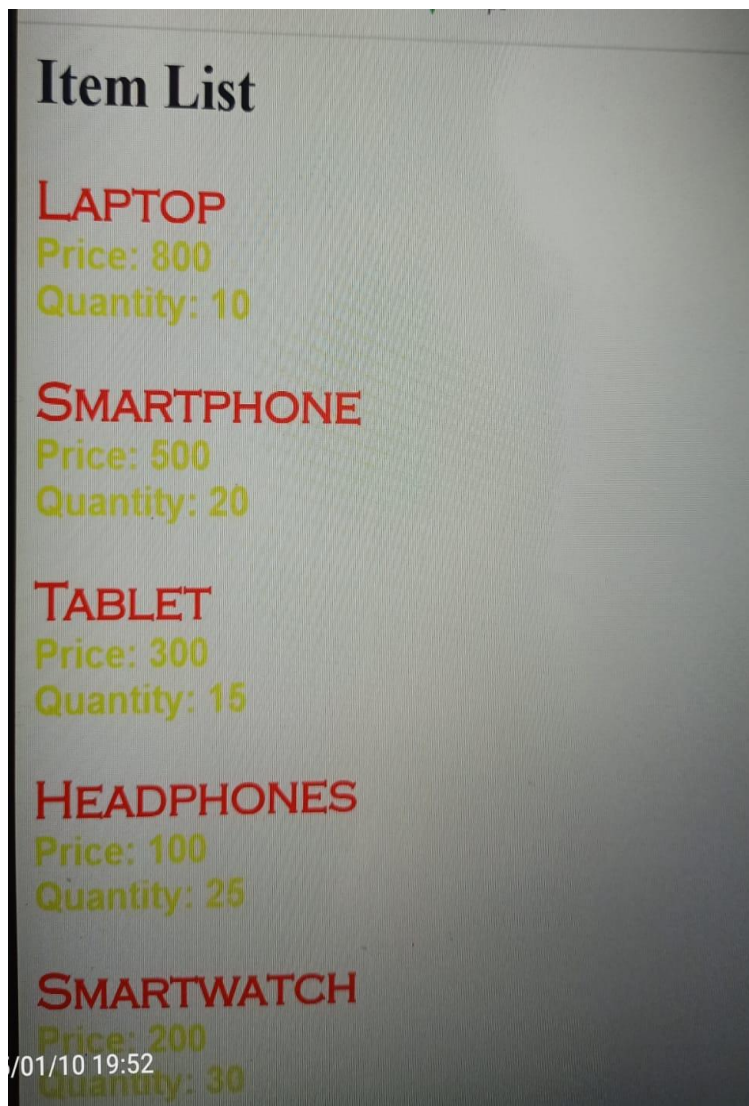
```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="Item.xsl"?>
<Items>
  <Item>
    <ItemName>Laptop</ItemName>
    <ItemPrice>800</ItemPrice>
    <Quantity>10</Quantity>
  </Item>
  <Item>
    <ItemName>Smartphone</ItemName>
    <ItemPrice>500</ItemPrice>
    <Quantity>20</Quantity>
  </Item>
  <Item>
    <ItemName>Tablet</ItemName>
    <ItemPrice>300</ItemPrice>
    <Quantity>15</Quantity>
  </Item>
  <Item>
    <ItemName>Headphones</ItemName>
    <ItemPrice>100</ItemPrice>
    <Quantity>25</Quantity>
  </Item>
  <Item>
    <ItemName>Smartwatch</ItemName>
    <ItemPrice>200</ItemPrice>
    <Quantity>30</Quantity>
  </Item>
</Items>
```

## ii) Item.xsl

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="html" indent="yes"/>
  <xsl:template match="/">
    <html>
      <head>
        <title>Item List</title>
        <style>
          .item-name { color: red; font-size: 16pt; font-weight:
bold; }
          .item-details { color: yellow; font-size: 12pt; font-weight:
bold; }
        </style>
      </head>
      <body>
        <h2>Item List</h2>
        <xsl:for-each select="Items/Item">
          <div>
            <span class="item-name"><xsl:value-of
select="ItemName"/></span><br/>
            <span class="item-details">Price: <xsl:value-of
select="ItemPrice"/></span><br/>
            <span class="item-details">Quantity: <xsl:value-of
select="Quantity"/></span><br/><br/>
          </div>
        </xsl:for-each>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```

## OUTPUT:

<http://localhost/webPractical%20Ty%20Sem%206/Item.xml>

A screenshot of a web application interface. At the top, the title 'Item List' is displayed in a large, bold, black serif font. Below the title, five product categories are listed in a red, bold, sans-serif font: 'LAPTOP', 'SMARTPHONE', 'TABLET', 'HEADPHONES', and 'SMARTWATCH'. Each category is followed by its price and quantity, displayed in a yellow, bold, sans-serif font. The items are listed vertically with a small gap between each entry. A timestamp '01/10 19:52' is visible in the bottom left corner of the screenshot.

Item List	
LAPTOP	Price: 800 Quantity: 10
SMARTPHONE	Price: 500 Quantity: 20
TABLET	Price: 300 Quantity: 15
HEADPHONES	Price: 100 Quantity: 25
SMARTWATCH	Price: 200 Quantity: 30

**3. Write a PHP script to generate an XML file in the following format in PHP.**

```
<?xml version="1.0" encoding="UTF-8"?>
<BookInfo>
<book>
<bookno>1</bookno>
<bookname>JAVA</bookname>
<authorname> Balguru Swami</authorname>
<price>250</price>
<year>2006</year>
</book>
<book>
<bookno>2</bookno>
<bookname>C</bookname>
<authorname> Denis Ritchie</authorname>
<price>500</price>
<year>1971</year>
</book>
</BookInfo>
=>
```

**I ) Item1.php**

```
<?php
$xml = new DOMDocument("1.0", "UTF-8");
$root = $xml->appendChild($xml->createElement("BookInfo"));

$books = [
    ["bookno" => 1, "bookname" => "JAVA", "authorname" => "Balguru
Swami", "price" => 250, "year" => 2006],
    ["bookno" => 2, "bookname" => "C", "authorname" => "Denis Ritchie",
"price" => 500, "year" => 1971],
];

foreach ($books as $book) {
    $bookElement = $root->appendChild($xml->createElement("book"));
    foreach ($book as $key => $value) {
        $bookElement->appendChild($xml->createElement($key, $value));
    }
}

$xml->save("book.xml");
echo "book.xml file created successfully!";
?>
```

## OUTPUT:

<http://localhost/webPractical%20Ty%20Sem%206/Item1.php>

book.xml file created successfully!

ii ) **book.xml** ( this file are automatically created )

```
<?xml version="1.0" encoding="UTF-8"?>
<BookInfo>
  <book>
    <bookno>1</bookno>
    <bookname>JAVA</bookname>
    <authorname>Balguru Swami</authorname>
    <price>250</price>
    <year>2006</year>
  </book>
  <book>
    <bookno>2</bookno>
    <bookname>C</bookname>
    <authorname>Denis Ritchie</authorname>
    <price>500</price>
    <year>1971</year>
  </book>
</BookInfo>
```

## Set B

1. Write PHP script to read above created "book.xml" file into simpleXML object. Display attributes and elements .(Hint L simple\_xml\_load\_file() function )

=>

i) book.php

```
<?php
```

```
$xml = simplexml_load_file("book.xml") or die("Failed to load XML file.");
```

```
foreach ($xml->book as $book) {  
    echo "Book No: {$book->bookno}, Name: {$book->bookname},  
    Author: {$book->authorname}, ";  
    echo "Price: {$book->price}, Year: {$book->year}<br><hr>";  
}  
?>
```

ii) previous(above) book.xml file

```
<?xml version="1.0" encoding="UTF-8"?>  
<BookInfo>  
    <book> <bookno>1</bookno>  
    <bookname>JAVA</bookname>  
    <authorname>Balguru Swami</authorname>  
    <price>250</price>  
    <year>2006</year></book>  
    <book>  
    <bookno>2</bookno>  
    <bookname>C</bookname>  
    <authorname>Denis Ritchie</authorname>  
    <price>500</price>  
    <year>1971</year> </book>  
</BookInfo>
```

## OUTPUT:

<http://localhost/webPractical%20Ty%20Sem%206/book.php>

Book No: 1, Name: JAVA, Author: Balguru Swami, Price: 250, Year: 2006

Book No: 2, Name: C, Author: Denis Ritchie, Price: 500, Year: 1971



**2. Write a PHP script to read “Movie.xml” file and print all MovieTitle and ActorName of file using DOMDocument Parser. “Movie.xml” file should contain following information with at least 5 records with values. MovieInfo MovieNo, MovieTitle, ActorName , ReleaseYear**

**=>**

**i) Movie.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<MovieInfo>
  <movie>
    <MovieNo>1</MovieNo>
    <MovieTitle>Inception</MovieTitle>
    <ActorName>Leonardo DiCaprio</ActorName>
    <ReleaseYear>2010</ReleaseYear>
  </movie>
  <movie>
    <MovieNo>2</MovieNo>
    <MovieTitle>The Matrix</MovieTitle>
    <ActorName>Keanu Reeves</ActorName>
    <ReleaseYear>1999</ReleaseYear>
  </movie>
  <movie>
    <MovieNo>3</MovieNo>
    <MovieTitle>Gladiator</MovieTitle>
    <ActorName>Russell Crowe</ActorName>
    <ReleaseYear>2000</ReleaseYear>
  </movie>
  <movie>
    <MovieNo>4</MovieNo>
    <MovieTitle>Forrest Gump</MovieTitle>
    <ActorName>Tom Hanks</ActorName>
    <ReleaseYear>1994</ReleaseYear>
  </movie>
  <movie>
    <MovieNo>5</MovieNo>
    <MovieTitle>The Dark Knight</MovieTitle>
    <ActorName>Christian Bale</ActorName>
    <ReleaseYear>2008</ReleaseYear>
  </movie>
</MovieInfo>
```

## ii) movie.php

```
<?php
$xml = simplexml_load_file("movie.xml") or die("Failed to load XML file.");

foreach ($xml->movie as $movie) {
    echo "MovieNo:{$movie->MovieNo},Movie Title: {$movie->MovieTitle}, Actor
Name: {$movie->ActorName},ReleaseYear:{$movie->ReleaseYear}<br><br>";
}
?>
```

## OUTPUT:

<http://localhost/webPractical%20Ty%20Sem%206/movie.php>

MovieNo:1,Movie Title: Inception, Actor Name: Leonardo  
DiCaprio,ReleaseYear:2010

MovieNo:2,Movie Title: The Matrix, Actor Name: Keanu  
Reeves,ReleaseYear:1999

MovieNo:3,Movie Title: Gladiator, Actor Name: Russell  
Crowe,ReleaseYear:2000

MovieNo:4,Movie Title: Forrest Gump, Actor Name: Tom  
Hanks,ReleaseYear:1994

MovieNo:5,Movie Title: The Dark Knight, Actor Name: Christian  
Bale,ReleaseYear:2008

## Set C

1. Create a XML file which gives details of movies available in “Movie CD Store “ from following categories a) Classical

b) Horror

c) Action Elements in each category are in the following format

<Category>

<MovieTitle> ..... </ MovieTitle >

<ActorName> .....</ActorName>

<ReleaseYear> ..... </ReleaseYear>

</Category>

Save the file with name “movies.xml”

=>

i) Movieinfo.php

```
<?php
```

```
$movies = [
```

```
    "Classical" => [
```

```
        ["Casablanca", "Humphrey Bogart", 1942],
```

```
        ["Gone with the Wind", "Clark Gable", 1939]
```

```
    ],
```

```
    "Horror" => [
```

```
        ["The Exorcist", "Linda Blair", 1973],
```

```
        ["Get Out", "Daniel Kaluuya", 2017]
```

```
    ],
```

```
    "Action" => [
```

```
        ["Die Hard", "Bruce Willis", 1988],
```

```
        ["Mad Max: Fury Road", "Tom Hardy", 2015]
```

```
    ]
```

```
];
```

```
$xml = new SimpleXMLElement('<MovieCDStore/>');
```

```
foreach ($movies as $category => $categoryMovies) {
```

```
    foreach ($categoryMovies as $movie) {
```

```
        $categoryElement = $xml->addChild($category);
```

```
        $categoryElement->addChild("MovieTitle", $movie[0]);
```

```
        $categoryElement->addChild("ActorName", $movie[1]);
```

```
        $categoryElement->addChild("ReleaseYear", $movie[2]);
```

```
    }
```

```
}
```

```
$xml->asXML("movies.xml");
```

```
echo "movies.xml file created successfully!";
```

```
?>
```

**ii) Movies.xml** (this file is automatically created)

```
<?xml version="1.0"?>
<MovieCDStore>
  <Classical>
    <MovieTitle>Casablanca</MovieTitle>
    <ActorName>Humphrey Bogart</ActorName>
    <ReleaseYear>1942</ReleaseYear>
  </Classical>
  <Classical>
    <MovieTitle>Gone with the Wind</MovieTitle>
    <ActorName>Clark Gable</ActorName>
    <ReleaseYear>1939</ReleaseYear>
  </Classical>
  <Horror>
    <MovieTitle>The Exorcist</MovieTitle>
    <ActorName>Linda Blair</ActorName>
    <ReleaseYear>1973</ReleaseYear>
  </Horror>
  <Horror>
    <MovieTitle>Get Out</MovieTitle>
    <ActorName>Daniel Kaluuya</ActorName>
    <ReleaseYear>2017</ReleaseYear>
  </Horror>
  <Action>
    <MovieTitle>Die Hard</MovieTitle>
    <ActorName>Bruce Willis</ActorName>
    <ReleaseYear>1988</ReleaseYear>
  </Action>
  <Action>
    <MovieTitle>Mad Max: Fury Road</MovieTitle>
    <ActorName>Tom Hardy</ActorName>
    <ReleaseYear>2015</ReleaseYear>
  </Action>
</MovieCDStore>
```

**OUTPUT:**

<http://localhost/webPractical%20Ty%20Sem%206/Movieinfo.php>

movies.xml file created successfully!