ASSIGNMENT-04

CSA4307

INTERNET PROGRAMMING

- JAYA HARINI. J.
192371034.
BTech CSBIOS.

PHP chessboard Program.

Explanation

1. uses nested for loops : 8 rows x 8 columns.
2. Alternates cell colors using ($ row + $ col) % 2
3. Applies inline CSS to size cells (30px x 30px).
4. output is rendered in a browser using HTML table structure.

CODE (chessboard.php):

```
<!Doctype html>
<html>
<head>
    <title> PHP chessboard </title>.
    <style>
    {
        width : 400px;
        border- collapse : collapse;
    }.
    td
    {
        width : 30 px;
        height : 30 px;
    }
    </style>.
</head>
<body>
<h2> chessboard </h2>
<? php
echo " < table border = '1' > ";
for ($ row = 0; $ row < 8; $ row++)
{
```

```
    echo "<tr>";
    for ($col = 0; $col < 8; $col++)
    {
        $color = (($row + $col) % 2 == 0) ? "white" : "black";
        echo "<td style='background-color: $color;'></td>";
    }
    echo "</tr>";
}
echo "</table>";
?>
</body>
</html>
```

Output:

An 8X8 chessboard grid alternating black and white cells in a browser

2. DTD for Four-stroke Motorbike catalog.

Explanation.

1. Describes structure of XML data.
2. motorbike has elements: make, model, year etc.,
3. engine is a nested element with further sub-elements.
4. accessories uses attributes with fixed values (Yes/No)
5. Ensures XML data consistency through validation

DTD File (motorbike.dtd)

```
<!ELEMENT catalog (motorbike+)>
<!ELEMENT motorbike (make, model, year, color, engine>
<!ELEMENT make (#PCDATA)>.
<!ELEMENT model (#PCDATA)>
<!ELEMENT Year (#PDATA)>
<!ELEMENT color (#PCDATA)>.
<!ELEMENT chasis_number (#PCDATA)>
<!ELEMENT engine (engine_number, number_of_cylinders,
                  type_of_fuel)>
<!ELEMENT engine_number (#PCDATA)>
<!ELEMENT No_of_cylinders (#PCDATA)>
<!ELEMENT type_of_fuel (#PCDATA)>
<!ELEMENT accessories EMPTY>
<!ATTLIST accessories.
    disc_brake (Yes | no) # Required.
    auto_start (Yes | no) # Required.
    radio (Yes | no) # Required>.
```

Sample XML using the DTD'.

```
<?xml version = "1.0"?>
<!Doctype catalog system "motorbike.dtd">.
<catalog>
    <motorbike>
        <make> Honda. </make>
        <model> CBR </model>
```

<Year> 2023 </Year>
<color> Red </color>.

<engine>
<engine-number> ENG1123 </engine-number>
<No-of-cylinders> 2 </No-of-cylinders>.
<type-of-fuel> Petrol </type-of-fuel>.
</engine>
<chasis-number> CHSH456 </chasis-number>.
<type-of-fuel> petrol </type-of-fuel>
</motorbike>
</catalog>.

Output:

If validated in an XML validator.

Result: Document is valid.

3. XSLT for student Database to HTML Table.

Explanation.

1. Converts XML student data into an HTML table.
2. uses <xsl:for-each> to loop over <student> elements.
3. <xsl:value-of> extracts data for display.
4. Adds table headers and rows dynamically.
5. Output is a Styled, readable HTML representation.

XML File (Student.xml):

```
<?xml version = "1.0" encoding = "UTF-8"?>
<?xml-stylesheet type = "text/xsl" href = "Student.xsl"?>
<Students>
    <Student>
        <name> Hariit </name>
        <roll> 101 </roll>
        <department> CSE </department>
    </Student>
        <name> Arjun </name>
        <roll> 102 </roll>
        <department> ECE </department>
    </Students>
</Students>
```

XSLT File (students.xsl):

```
<?xml version = "1.0" encoding = "UTF-8"?>
<xsl:stylesheet version = "1.0"
    <xsl:template match = "/">
        <html>
        <head> <title> Student Table </title> </head>
        <body>
          <h2> Student Database </h2>
          <table border = "1">
              <tr bordcolor = "#cccccc">
                  <th> Name </th>
                  <th> Roll No </th>
```

Exp No: 08 — PHP Program to calculate Factorial of a Number - SEPARATE FILE.

DATE:

Aim: To write a PHP Program ... factorial of

```
<th> Department </th>.
</tr>.
<xsl: for-each select = "students / student">
    <tr>.
    <td> <xsl : value - of select = "name" /> </td>.
    <td> <xsl : value - of select = "roll" /> </td>.
    <td> <xsl : value - of select = "department" /> </td>
    </tr>.
    </xsl : for - each>
</table>
</body>
</html>
</xsl : template>
</xsl : stylesheet>.
```

output:

NAME	ROLL NO:	DEPARTMENT
Harini	101	CSE
Arjun	102	ECE