**Set-1**

**Palindrome:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <input type="text" id="i1">

    <button onclick="check()">Click</button>

    <script>

        function check() {

            const number = document.getElementById('i1').value;

            const num = number.toString();

            const reversedNum = num.split('').reverse().join('');

            if (num === reversedNum) {

                alert(`${number} is a palindrome.`);

            } else {

                alert(`${number} is not a palindrome.`);

            }

        }

    </script>

</body>

</html>

**Simple Order Form(JSP):**

<html>

<body>

<%! String Items[]={"DVD","CD","Diskette"};

double price [] = {19.99,12.99,1.99};

int Quantity [] = {2,9,24};

double TotalPrice [] =new double [Items.length]; %>

<table border="1px">

<tr>

<td>Item</td>

<td> Price </td>

<td>Quantity</td>

<td>Total Price </td>

</tr>

<% for (int i=0;i<Items.length;i++) {

TotalPrice[i]=price[i]\*Quantity[i]; %>

<tr>

<td><% out.println(Items[i]); %></td>

<td><% out.println(price[i]); %></td>

<td><% out.println(Quantity[i]); %></td>

<td><% out.println(TotalPrice[i]); %></td>

</tr>

<%} %>

</table>

</body>

</html>

**Set-2**

**Angular JS**

<!DOCTYPE html>

<html lang="en" ng-app="myApp">

<head>

<meta charset="UTF-8">

<title>AngularJS Current Time</title>

<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>

</head>

<body>

<div ng-controller="timeController">

<h2>Current Time: {{ currentTime }}</h2>

</div>

<script>

var app = angular.module('myApp', []);

app.controller('timeController', function($scope, $interval) {

function updateTime() {

$scope.currentTime = new Date().toLocaleTimeString();

}

updateTime();

$interval(updateTime, 1000);

});

</script>

</body>

</html>

**Factorial**

**<!DOCTYPE html>**

**<html lang="en">**

**<head>**

**<meta charset="UTF-8">**

**<meta name="viewport" content="width=device-width, initial-scale=1.0">**

**<title>Document</title>**

**</head>**

**<body>**

**<input type="number" id="i1" placeholder="Enter a number">**

**<button onclick="calculateFactorial()">Calculate Factorial</button>**

**<script>**

**function calculateFactorial() {**

**const number = parseInt(document.getElementById('i1').value);**

**if (!isNaN(number)) {**

**let factorial = 1;**

**for (let i = 1; i <= number; i++) {**

**factorial \*= i;**

**}**

**alert(`The factorial of ${number} is: ${factorial}`);**

**} else {**

**alert("Please enter a valid number.");**

**}**

**}**

**</script>**

**</body>**

**</html>**

**Set-3**

**XML**

<?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">XQuery Kick Start</title>

<author>James McGovern</author>

<author>Per Bothner</author>

<author>Kurt Cagle</author>

<author>James Linn</author>

<author>Vaidyanathan Nagarajan</author>

<year>2003</year>

<price>49.99</price>

</book><book category="web">

<title lang="en">Learning XML</title>

<author>Erik T. Ray</author>

<year>2003</year>

<price>39.95</price>

</book> </bookstore>

**XSLT**

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

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="bookstore">

<xs:complexType>

<xs:sequence>

<xs:element name="book" maxOccurs="unbounded">

<xs:complexType>

<xs:sequence>

<xs:element name="title" type="xs:string"/>

<xs:element name="author" type="xs:string" minOccurs="0"

maxOccurs="unbounded"/>

<xs:element name="year" type="xs:integer"/>

<xs:element name="price" type="xs:decimal"/>

</xs:sequence>

<xs:attribute name="category" type="xs:string" use="required"/>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

**Set-4**

**abc.html**

<html>

<body>

<h1>ABC</h1>

<h4>This is abc.html file </h4>

<p>abc</p>

</body>

</html>

**xyz.html**

<html>

<body>

<h1>XYZ</h1>

<h4> This is xyz.html file </h4>

<p>xyz</p>

</body>

</html>

**Fileserver.js**

var http = require('http');

var url = require('url');

var fs = require('fs');

http.createServer(function (req, res) {

var q = url.parse(req.url, true);

var filename = "." + q.pathname;

fs.readFile(filename, function(err, data) {

if (err) {

res.writeHead(404, {'Content-Type': 'text/html'});

return res.end("404 Not Found");

}

res.writeHead(200, {'Content-Type': 'text/html'});

res.write(data);

return res.end();

});

}).listen(8080);

**Set-6**

**home.html**

<html>

<body>

<h1>Welcome student database</h1>

<p>choose an action:</p>

<ul><li><a href="insert.html"> Insert</a></li>

<li><a href ="update.html"> Update</a></li>

<li><a href="delete.html"> Delete</a></li></ul>

</body>

</html>

**insert.html**

<html>

<body><form action="insert.jsp" method="Post">

<h1 align="center">Insert</h1>

<table align="center">

<tr><td>Name :</td>

<td><input type="text" name="sname"/></td></tr>

<tr><td>Roll no:</td>

<td><input type="text" name="srno"/></td></tr>

<tr><td>cgpa :</td>

<td><input type="text" name="cgpa"/></td></tr>

<tr><td><input type="submit" value="insert" /></td>

<td><input type="reset" value="clear"/></td></tr>

</table>

</form>

</body>

</html>

**insert.jsp**

<%@page import="java.sql.\*" %>

<%

String name=request.getParameter("sname");

int rno=Integer.parseInt(request.getParameter("srno"));

String cgpa=request.getParameter("cgpa");

//out.println(name+" "+rno+""+cgpa);

try{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/cai", "root", "root");

PreparedStatement ps=con.prepareStatement("insert into std values(?,?,?)");

ps.setInt(1, rno);

ps.setString(2, name);

ps.setString(3,cgpa);

int i=ps.executeUpdate();

if (i!=0){

out.println("inserted..");

}

else{

out.println("not inserted..");

}

}

catch (Exception e){out.print(e);}

session.setAttribute("sname", name);

session.setAttribute("srno",rno);%>

**update.html**

<body>

<form action="update.jsp" method="post">

<h1 align="center"> Update </h1>

<table align="center">

<tr><td>Name:</td>

<td><input type="text" name="sname"/></td>

</tr>

<tr><td>Roll no: </td>

<td><input type="number" name="srno"/></td></tr>

<tr><td>cgpa :</td>

<td><input type="text" name="cgpa"/></td></tr>

<tr><td><input type="submit" value="Update" /></td></tr>

</table>

</form>

</body>

**update.jsp**

<%@page import="java.sql.\*" %>

<%

String name=request.getParameter("sname");

int rno=Integer.parseInt(request.getParameter("srno"));

String cgpa=request.getParameter("cgpa");

try{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/cai", "root", "root");

PreparedStatement ps = con.prepareStatement("UPDATE std SET rno=?, cgpa=? WHERE name=?");

ps.setInt(1, rno);

ps.setString(2, cgpa);

ps.setString(3,name);

int i=ps.executeUpdate();

if (i!=0){

out.println("updated..");

}

else{

out.println("not updated..");

}

}

catch (Exception e){out.print(e);}%>

**delete.html**

<body>

<form action="del.jsp" method="post">

<h1 align="center" >Delete</h1>

<table align="center">

<tr><td>Name :</td>

<td><input type="text" name="sname"/></td></tr>

<tr><td><input type="submit" value="Delete"/></td></tr>

</table>

</form>

</body>

**delete.jsp**

<%@page import="java.sql.\*" %>

<% String name=request.getParameter("sname");

try{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/cai", "root",

"root");

PreparedStatement ps = con.prepareStatement("DELETE FROM `std` WHERE name=?");

ps.setString(1,name);

int i=ps.executeUpdate();

if (i!=0){

out.println("Deleted..");

}

else{

out.println("not Deleted..");

}

}

catch (Exception e){out.print(e);}%>

**Set-7**

**Simple JSP for date and time**

<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Date and Time Example</title>

</head>

<body>

<h1>Date and Time Example</h1>

<%-- Import necessary Java classes --%>

<%@ page import="java.util.Date" %>

<%-- Create a Date object to get the current date and time --%>

<% Date currentDate = new Date(); %>

<p>Current Date and Time: <%= currentDate %></p>

</body>

</html>

**Set-8**

**abc.html**

<html>

<body>

<h1>ABC</h1>

<h4>This is abc.html file </h4>

<p>abc</p>

</body>

</html>

**xyz.html**

<html>

<body>

<h1>XYZ</h1>

<h4> This is xyz.html file </h4>

<p>xyz</p>

</body>

</html>

**Fileserver.js**

var http = require('http');

var url = require('url');

var fs = require('fs');

http.createServer(function (req, res) {

var q = url.parse(req.url, true);

var filename = "." + q.pathname;

fs.readFile(filename, function(err, data) {

if (err) {

res.writeHead(404, {'Content-Type': 'text/html'});

return res.end("404 Not Found");

}

res.writeHead(200, {'Content-Type': 'text/html'});

res.write(data);

return res.end();

});

}).listen(8080);

**Css internal, external, inline**

**Inline**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Inline CSS Example</title>

</head>

<body>

<p style="line-height: 1.5; margin-top: 20px; font-family: Arial, sans-serif; background-color: #f0f0f0; color: #333; text-align: center;">This is a paragraph with inline styles applied directly to the HTML element. It has a line-height of 1.5 and a top margin of 20px.</p>

</body>

</html>

**Internal**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Internal CSS Example</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #f0f0f0;

color: #333;

text-align: center;

}

p {

line-height: 1.5;

margin-top: 20px;

}

</style>

</head>

<body>

<p>This is a paragraph with styles applied using internal CSS. It has a line-height of 1.5 and a top margin of 20px.</p>

</body>

</html>

**External**

**html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>External CSS Example</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<p>This is a paragraph with styles applied using an external CSS file. It has a line-height of 1.5 and a top margin of 20px.</p>

</body>

</html>

**Css**

/\* styles.css \*/

body {

font-family: Arial, sans-serif;

background-color: #f0f0f0;

color: #333;

text-align: center;

}

p {

line-height: 1.5;

margin-top: 20px;

}