

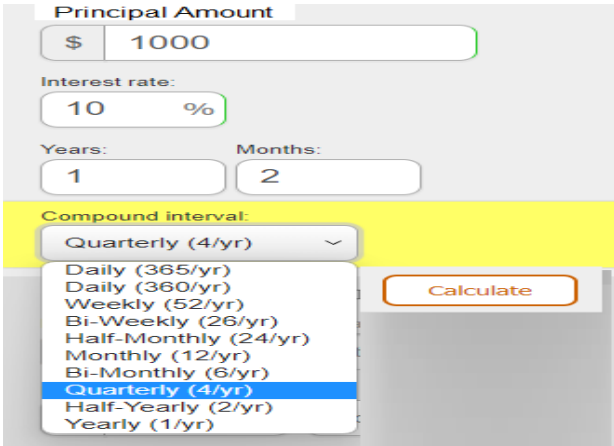
## Department of Computer Science, Gujarat Vidyapith, Ahmedabad-380009

Academic Year : 2025-26 Semester: 2<sup>nd</sup> Subject : Msc.IT-202: Web Technology

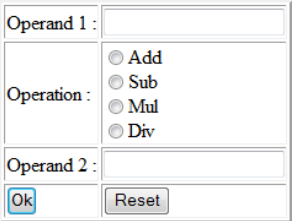
Enrollment No.: \_\_\_\_\_

Student Name: \_\_\_\_\_

Subject Teacher: Kamallesh Salunke

No.	Problem Definition	Assignment / Submission Date	Sign of Teacher	Grade	Remark
1.	<p>Create HTML form as follows.</p>  <p>Create a servlet that gets submitted values and calculate compound Interest.          Note: Write JavaScript to validate the input fields.</p> <p>Formula <math>A = P (1 + R/(n*100))^{(nt)}</math></p> <p>A = the future value of the investment, including interest. P = the principal investment amount (the initial deposit or loan amount). R = the annual interest rate (decimal). n = the number of times that interest is compounded per unit t. t = the time the money is invested for, where t = no. of months/12 (E.g. 1year and 4 months so t = 16/12 = 1.333). Note: 1) Validate all inputs using JavaScript 2) Assumptions can be made wherever necessary.</p>				
2.	<p>Write servlet which prints the name and value for all its init parameters, details of HTTP request headers, client/browser, and server.          Note: Assumptions can be made wherever necessary.</p>				
3.	<p>Design the registration page/form and write JavaScript to validate the following fields of the Registration page.</p> <ol style="list-style-type: none"> <li>1. First Name (Name should contain alphabets and the length should not be less than 6 characters).</li> <li>2. Last Name (Name should contain alphabets and the length should not be less than 6 characters).</li> <li>3. Password (Password should not be less than 6 characters in length).</li> </ol>				

	<p>4. Conform Password (Password and conform password must be same).</p> <p>5. E-mail (should not contain any invalid and must follow the standard pattern of email and it is unique and required field)</p> <p>6. Mobile Number (Phone number should contain 10 digits only).</p> <p>7. Address (should not be Empty).</p> <p>8. Select user role (admin, registered user, guest etc.) from dropdown (required field)</p> <p>8. Create a table user table to store the above details.</p> <p>9. Write a servlet to insert data into the table</p> <p>Note: Assumptions can be made wherever necessary.</p>				
4.	<p>1. Use user table (email, password, name, mobile) and create user-role table (email, role)</p> <p>2. Design a good-looking login page which accepts and submits username as email and password.</p> <p>3. Write JavaScript to validate inputs.</p> <p>3. Write servlet to verify whether the user exists or not.</p> <p>4. if exist, redirect to user's home page.</p> <p>Note: Assumptions can be made wherever necessary.</p>				
5.	<p>Write a CRUD application using servlet(s).</p> <p>Note: Assumptions can be made wherever necessary.</p>				
6.	<p>A web application using servlet that takes a name as input and on submit it shows a hello &lt;name&gt; page where name is taken from the request. It shows the start time at the right top corner of the page, displays all session information, displays the number of times a client has accessed it and provides a logout button. On clicking this button, it should show a logout page with Thank You &lt;name&gt; message with the duration of usage (hint: Use session to store name and time).</p>				
7.	<p>Design and develop JSP application to demonstrate</p> <p>1. JSP Scripting elements</p> <p>2. JSP Directives</p> <p>3. JSP Implicit Objects</p> <p>4. JSP Action tags</p>				
8.	<p>1. use user and user-role table that create in previous exercise.</p> <p>2. Create a good-looking login form which accepts and submit username and password.</p> <p>3. Validate all inputs using JavaScript.</p> <p>4. Incorporate JSP Scripting elements, JSP Directives, JSP Implicit Objects, and JSP Action tags in your development.</p> <p>4. Write JSP page to verify whether the user exists or not and perform following,</p> <p>1. If the user does not exist, back to login page.</p> <p>2. If user is exists and user-role is admin then redirect to "admin.jsp", else on "welcome.jsp"</p> <p>Note: Assumptions can be made wherever necessary. For redirection use &lt;jsp:forward/&gt;</p>				

9.	<p>1. Create Book table (bookId, title, author, price, Quantity, ISBN, publisher, edition year, catalogueId).</p> <p>2. Create CRUD application using JSP(s).</p>				
10.	<p>1. Create HTML form as following</p>  <p>2. Create a JavaBean for calculating and return results.</p> <p>3. Write a JSP page to use/call JavaBean and display result using standard action tags.</p> <p>4. Write JavaScript to validate the input fields.</p>				
11.	<p>Write XML file (planner.xml) according to the following information</p> <ul style="list-style-type: none"> <li>planner as root element.</li> <li>planner has one or more year element.</li> <li>year contains one attribute 'value' and one or more child element date.</li> <li>date has two attribute 'moth' and 'day' and one child element note.</li> </ul> <p>Write DTD that enforce above rules on XML</p>				
12.	<p>1. Write a DTD that will define the following structure for documents of type studentlist</p> <ul style="list-style-type: none"> <li>A studentlist element consists of one or more student entries.</li> <li>A student entry consists of the student's enrollmentno, name, the study_program in which the student is enrolled and enrolment_year.</li> <li>A student has an enrollmentno by which they can be uniquely identified.</li> <li>A student's name consists of three elements, the firstname, the middlename which is optional, and the lastname.</li> <li>study_program contain values like are "BSc", "MSc", "MSW", and "MBA" which is required.</li> <li>enrolment_year is four-digit numbers like 2017 which is required.</li> </ul> <p>Write XML file of a studentlist document and validate with your DTD.</p>				
13.	<p>1. Write a Schema that will define the following structure for documents of type studentlist</p> <ul style="list-style-type: none"> <li>A studentlist element consists of one or more student entries.</li> <li>A student entry consists of the student's enrollmentno, name, the study_program in which the student is enrolled and enrolment_year.</li> <li>A student has an enrollmentno by which they can be uniquely identified.</li> </ul>				

	<ul style="list-style-type: none"> <li>○ A student's name consists of three elements, the firstname, the middlename which is optional, and the lastname.</li> <li>○ study_program contain values like are "BSc", "MSc", "MSW", and "MBA" which is required.</li> <li>○ enrolment_year is four-digit numbers like 2017 which is required.</li> </ul> <p>2. Write XML file of a studentlist document and validate with your DTD.</p>				
14.	<p>1. Write an XML Schema that will define the following structure for documents of type phonebook</p> <ul style="list-style-type: none"> <li>○ • A phonebook is a root element and that consists of one or more department elements.</li> <li>○ • A department element consists of the deptname, deptphone, deptemail, and one or more employee elements.</li> <li>○ • An employee element consists of empid, empname, empphone, and empemail.</li> <li>○ • An empid by which they can be uniquely identified.</li> </ul> <p>2. Write XML file of a phonebook document and validate with your XML Schema.</p>				
15.	<p>Design and Develop MVC based User Management web application that manages a collection of users with the basic features: list, insert, update, delete (or CRUD operations - Create, Update, Read and Delete).</p> <p><b>Development Steps:</b> 1. Create an Eclipse/NetBeans Dynamic Web Project 2. Add Dependencies 3. Project Structure 4. MySQL Database Setup 5. Create JavaBean - User.java 6. Create a UserDao.java 7. Create a UserServlet.java 8. Creating User Listing JSP Page - user-list.jsp 9. Create a User Form JSP Page - user-form.jsp 10. Creating Error JSP page 11. Deploying and Testing the Application</p>				
16.	<p>Design and develop JSP based applications using JSP using JSTL tags.</p>				

Grade	A	B	C	D	E	F	S
Marks	18 - 20	14-17	11-13	8-10	5-7	1-4	Absent