1. (a) Features of Web 2.0 and Technologies Used

Features of Web 2.0:

1. **Dynamic Content**: Content is user-driven and updated frequently.
2. **Rich User Experience**: Enhanced interactivity using AJAX and similar technologies.
3. **Social Media Integration**: Users can connect and share content via platforms like Facebook and Twitter.
4. **User-Generated Content**: Platforms like YouTube, Wikipedia, and blogs allow users to contribute content.
5. **Interoperability**: Integration of applications using APIs and web services.
6. **Scalability**: Designed to handle large user bases effectively.

Technologies Used:

- AJAX (Asynchronous JavaScript and XML)
- RESTful APIs
- JSON for lightweight data exchange
- HTML5 and CSS3 for improved UI
- JavaScript frameworks like React, Angular, and Vue.js

---

1. (b) Well-formed XML Document

A well-formed XML document adheres to the following rules:

1. It has a single root element.
2. Tags are properly nested and case-sensitive.
3. All attributes are quoted.
4. Special characters like `<` and `&` are escaped.

Example:

xml
Copy code

```
<?xml version="1.0" encoding="UTF-8"?>
<student>
    <name>John Doe</name>
    <age>21</age>
    <course>BCA</course>
</student>
```

---

1. (c) WML Code for Input Screen

Code:

wml
Copy code

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>
    <card id="input" title="User Info">
        <p>
            Name: <input name="name" type="text"/>
            <br/>
            Age (in years): <input name="age" type="text"/>
        </p>
    </card>
</wml>
```

---

1. (d) Characteristics of Server-Side Scripting Tools

1. **State Management**: Maintain session and user state.
2. **Database Connectivity**: Provide seamless integration with databases.
3. **Security**: Ensure secure handling of user data and prevent vulnerabilities like SQL injection.

---

1. (e) JSP Scriptlet to Display Numbers 11 to 15

Code:

jsp
Copy code

```
<%
for (int i = 11; i <= 15; i++) {
    out.println(i + "<br>");
}
```

```
%>
```

---

1. (f) JSP Code to Insert Record into Database

Code:

jsp
Copy code

```
<%
String enrolment = request.getParameter("enrolment");
String studentName = request.getParameter("studentname");
int marks = Integer.parseInt(request.getParameter("marks"));

String query = "INSERT INTO masterdata (enrolment, studentname, marks)
VALUES (?, ?, ?)";
PreparedStatement pstmt = connection.prepareStatement(query);
pstmt.setString(1, enrolment);
pstmt.setString(2, studentName);
pstmt.setInt(3, marks);
pstmt.executeUpdate();
%>
```

---

2. (a) Use of `<div>` and `<span>` Tags

- `<div>`: Used to define a block-level container for content.
- `<span>`: Used to define an inline container for text.

Example:

html
Copy code

```
<div style="background-color: lightblue;">
    This is a block-level element.
</div>
<span style="color: red;">This is an inline element.</span>
```

---

2. (b) Class Selector in CSS

Need: To apply the same style to multiple elements.

Example:

html
Copy code
```
<style>
    .highlight { color: blue; }
</style>
<p class="highlight">This text is blue.</p>
```

Difference with ID Selector:

- `class` can be reused; `id` is unique per page.

---

2. (c) Tags for Specific Tasks

Large Text Input: `<textarea>`
html
Copy code
```
<textarea rows="4" cols="50"></textarea>
```

1.

Footer: `<footer>`
html
Copy code
```
<footer>Page Footer</footer>
```

2.

Password Input: `<input type="password">`
html
Copy code
```
<input type="password"/>
```

3.

---

3. (a) JavaScript to Modify Paragraph

Code:

html
Copy code

```
<p id="text">Web Programming is Good</p>
<script>
    document.getElementById("text").innerText = "Learn Web
Programming";
</script>
```

---

3. (b) HTML DOM Node Tree

The HTML DOM is a tree-like structure where each element is a node. Example:

html
Copy code

```
<html>
    <body>
        <div>
            <p>Text</p>
        </div>
    </body>
</html>
```

The root node is `<html>`, with `<body>` and `<div>` as children.

---

3. (c) Purpose and Limitations of DTD

Purpose: Defines the structure and rules for an XML document.

Limitations:

- Limited data type support.
- Cannot specify constraints like unique values.

Difference with XML Schema:

- DTD uses a proprietary syntax; XML Schema uses XML syntax.

---

4. (a) MVC Architecture

Explanation:

- **Model**: Manages data.
- **View**: Displays data to the user.
- **Controller**: Handles user input.

Diagram:

rust
Copy code

```
User <-> Controller <-> Model <-> View
```

---

4. (b) JSP Implicit Objects

1. **request**: Provides data sent by the client.
2. **response**: Sends data back to the client.

Example:

jsp
Copy code

```
<%
String param = request.getParameter("username");
response.getWriter().println("Hello, " + param);
%>
```

---

5. (a) Session vs. Cookies

- **Session**: Stores data on the server.
- **Cookies**: Stores data on the client.

---

5. (b) Purpose of Session Management

Tracks user interactions across multiple requests. Example:

jsp
Copy code
```
<%
session.setAttribute("user", "John");
%>
```

---

5. (c) HTTP Status 500

Indicates a server-side error caused by faulty scripts or misconfigurations.

---

5. (d) Need for JDBC

JDBC is needed for database connectivity in Java. ODBC cannot be used due to platform dependencies and lack of Java optimization.

2022 DEC

1. (a) JavaScript and HTML to Change Paragraph Content

Code:

html
Copy code
```
<!DOCTYPE html>
<html>
<body>
    <p id="IGNOU">BCA Examination</p>
    <button onclick="changeContent()">Click Me</button>
    <script>
        function changeContent() {
            document.getElementById("IGNOU").innerText = "The out of
distance education";
        }
```

```
    </script>
</body>
</html>
```

---

1. (b) Box Model in HTML

The HTML box model describes how the space around elements is calculated, consisting of the following layers (from inside out):

1. **Content**: The inner area where text and images appear.
2. **Padding**: Space between the content and the border.
3. **Border**: The outer edge of the element.
4. **Margin**: Space between the element and neighboring elements.

Diagram:

diff
Copy code

```
+--------------------+
|      Margin        |
+--------------------+
|      Border        |
+--------------------+
|      Padding       |
+--------------------+
|     Content        |
+--------------------+
```

---

1. (c) XML Document

Code:

xml
Copy code

```
<?xml version="1.0" encoding="UTF-8"?>
<students>
    <student>
        <name>
```

```
            <firstname>John</firstname>
            <lastname>Doe</lastname>
        </name>
        <programme>BCA</programme>
    </student>
    <student>
        <name>
            <firstname>Jane</firstname>
            <lastname>Smith</lastname>
        </name>
        <programme>MCA</programme>
    </student>
</students>
```

---

1. (d) Cookies

Definition:
Cookies are small text files stored on the client-side to retain user-specific information such as preferences or session data.

Purpose:

1. To track user sessions.
2. Store user preferences.
3. Personalize web applications.

Code to Create a Cookie:

javascript
Copy code

```
document.cookie = "username=JohnDoe; expires=Fri, 31 Dec 2024 23:59:59 GMT; path=/";
```

---

1. (e) JSP Script to Print Numbers 1 to 5 in Paragraphs

Code:

jsp
Copy code

```
<%
for (int i = 1; i <= 5; i++) {
    out.println("<p>" + i + "</p>");
}
%>
```

---

1. (f) Advantages of JSP over Servlets

1. **Simplified Syntax**: JSP allows embedding HTML directly into Java code.
2. **Separation of Concerns**: JSP separates the presentation layer from the business logic.
3. **Custom Tag Libraries**: Easier to use reusable components.
4. **Auto Compilation**: JSP pages are automatically compiled into Servlets.

---

2. (a) Features and Technologies of Web 2.0

Features:

1. **Interactive Content**: Real-time updates and interactivity.
2. **Social Networking**: Collaboration and sharing among users.
3. **Rich User Interfaces**: Use of AJAX and modern design elements.

Technologies:

1. **AJAX**: Enables asynchronous updates of web pages.
2. **HTML5**: Provides multimedia and enhanced semantic tags.
3. **CSS3**: Enables advanced styling like animations and transitions.

---

2. (b) CSS Usage

Code:

html
Copy code

```
<!DOCTYPE html>
<html>
<head>
    <style>
        body { background-color: red; }
```

```
        p { font-size: 17px; color: blue; }
    </style>
</head>
<body>
    <p>This is a paragraph.</p>
</body>
</html>
```

---

3. (a) DTD in XML and Valid Document

Definition:
A DTD (Document Type Definition) defines the structure and legal elements of an XML document.

Valid XML Document:

xml
Copy code

```
<?xml version="1.0"?>
<!DOCTYPE University [
    <!ELEMENT University (Name, Address)>
    <!ELEMENT Name (#PCDATA)>
    <!ELEMENT Address (Location, Pincode)>
    <!ELEMENT Location (#PCDATA)>
    <!ELEMENT Pincode (#PCDATA)>
]>
<University>
    <Name>IGNOU</Name>
    <Address>
        <Location>Delhi</Location>
        <Pincode>110068</Pincode>
    </Address>
</University>
<University>
    <Name>MIT</Name>
    <Address>
        <Location>Cambridge</Location>
        <Pincode>02139</Pincode>
```

```
    </Address>
</University>
```

---

3. (b) WML Code for Table

Code:

wml
Copy code

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>
    <card id="table" title="Courses">
        <p>
            <table columns="2">
                <tr><td>Course Name</td><td>Teacher</td></tr>
                <tr><td>Web Programming</td><td>ABC</td></tr>
                <tr><td>DBMS</td><td>XYZ</td></tr>
            </table>
        </p>
    </card>
</wml>
```

---

4. (a) Differences

1. **GET vs POST**:
    - GET appends data to the URL; POST sends data in the body.
    - GET is less secure; POST is more secure.
2. **HTTP Server vs Web Container**:
    - HTTP Server handles HTTP requests.
    - Web Container runs Java-based applications like Servlets and JSP.
3. **Action Elements vs JSP Implicit Objects**:
    - Action elements are custom JSP tags like `<jsp:forward>`.
    - Implicit objects are pre-defined objects like `request`, `response`.

---

4. (b) JSP Exceptions and Errors

Exception: Runtime errors in a web application.

Types of Errors:

1. **Syntax Error**: Incorrect Java code. Example: Missing a semicolon.
2. **Runtime Error**: Divide by zero or null pointer.
3. **Logical Error**: Wrong business logic implementation.

---

5. Explanation with Examples

1. **DriverManager Class**:
    - Manages JDBC drivers and establishes database connections.

java
Copy code
```
Connection conn =
DriverManager.getConnection("jdbc:mysql://localhost/db", "user",
"pass");
```

2.
3. **PreparedStatement Method**:
    - Used for parameterized SQL queries.

java
Copy code
```
PreparedStatement pstmt = conn.prepareStatement("INSERT INTO students
VALUES (?, ?)");
pstmt.setString(1, "John");
pstmt.setInt(2, 90);
```

4.
5. **ResultSet Object**:
    - Retrieves query results.

java
Copy code
```
ResultSet rs = pstmt.executeQuery();
while (rs.next()) {
    System.out.println(rs.getString("name"));
}
```

6.
7. **MVC Architecture**:
    - Separates application into **Model**, **View**, and **Controller**.
    - **Example**:
        - Model handles data (e.g., database).
        - View displays HTML pages.
        - Controller processes user input and updates the view.

Answers for the BCS-053 Term-End Examination, June 2023

---

1. (a) XML and DTD

(i) XML Document:

xml
Copy code

```xml
<?xml version="1.0" encoding="UTF-8"?>
<students>
    <student>
        <firstname>Ravi</firstname>
        <lastname>Kumar</lastname>
        <programme>BCA</programme>
    </student>
    <student>
        <firstname>Ananth</firstname>
        <lastname>Malik</lastname>
        <programme>MCA</programme>
    </student>
</students>
```

(ii) DTD to Validate the XML:

xml
Copy code

```xml
<!DOCTYPE students [
    <!ELEMENT students (student+)>
    <!ELEMENT student (firstname, lastname, programme)>
    <!ELEMENT firstname (#PCDATA)>
    <!ELEMENT lastname (#PCDATA)>
```

```
    <!ELEMENT programme (#PCDATA)>
]>
```

---

1. (b) Java Code for Querying Database

Code:

java
Copy code
```
ResultSet rs = stmt.executeQuery("SELECT * FROM Programme WHERE
Duration > 1");
```

Explanation:

- The `stmt` object is used to execute a SQL query.
- The query retrieves all rows from the `Programme` table where the `Duration` is greater than 1 year.

---

1. (c) JavaScript to Change Paragraph Color

Code:

html
Copy code
```
<!DOCTYPE html>
<html>
<body>
    <p id="IGNOU">This is a paragraph.</p>
    <button onclick="changeColor()">Change Color</button>
    <script>
        function changeColor() {
            document.getElementById("IGNOU").style.color = "green";
        }
    </script>
</body>
</html>
```

---

1. (d) GET Method in HTTP

Explanation:

- The **GET** method is used to request data from a server and sends data appended to the URL.
- It is primarily used to retrieve data without modifying server-side resources.

Example:

html
Copy code

```
<form method="GET" action="process.php">
    <input type="text" name="username">
    <input type="submit" value="Submit">
</form>
```

- Data (e.g., `?username=John`) is appended to the URL when the form is submitted.

---

2. (a) MVC Architecture

Explanation:
MVC (Model-View-Controller) is an architectural pattern that separates an application into three interconnected components:

1. **Model**: Manages data and business logic.
2. **View**: Handles the display of data (UI).
3. **Controller**: Manages user input and updates the Model and View.

Diagram:

rust
Copy code

```
User
   |
   v
Controller -> Model <-> Database
   |
   v
```

```
View
```

---

2. (b) JSP Implicit Objects

Explanation:
Implicit objects in JSP are pre-defined objects available to developers to simplify coding.

Request: Represents the HTTP request object and contains request parameters.
jsp
Copy code

```
<%
String name = request.getParameter("username");
out.println("Welcome, " + name);
%>
```

1.

Response: Represents the HTTP response object and is used to redirect or set headers.
jsp
Copy code

```
<%
response.sendRedirect("welcome.jsp");
%>
```

2.

---

3. (a) Session Management in HTTP

Need for Session Management:
HTTP is a stateless protocol. Session management allows the server to retain user-specific information across multiple requests.

Definitions:

- **Session**: A sequence of requests from a single user.
- **State**: Information about a user's interaction with the application.

JSP Code to Create a Session:

jsp
Copy code

```
<%
HttpSession session = request.getSession();
session.setAttribute("username", "JohnDoe");
%>
<p>Session created for user: <%= session.getAttribute("username")
%></p>
```

---

3. (b) Include Directive in JSP

Explanation:
The **include directive** is used to include static resources (like HTML) or JSP fragments at compile time.

Example:

jsp
Copy code

```
<%@ include file="header.jsp" %>
<p>This is the main content.</p>
<%@ include file="footer.jsp" %>
```

---

4. (a) Web 2.0 Technologies

1. **Widgets**: Small, portable web applications that provide interactive features (e.g., weather updates, stock prices).
2. **Blogging**: Allows users to publish content and engage in discussions.
3. **Podcasting**: Distributes audio or video files via RSS feeds.

---

4. (b) HTML Form

Code:

html
Copy code

```
<!DOCTYPE html>
<html>
<body>
```

```
    <form>
        <label for="name">Name:</label><br>
        <input type="text" id="name" name="name"><br>
        <label for="email">Email:</label><br>
        <input type="email" id="email" name="email"><br>
        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

---

5. (a) Advantages of CSS and Linking an External Style Sheet

Advantages of CSS:

1. Separates content from presentation.
2. Reusable styles across multiple pages.
3. Easier maintenance and faster development.

External Style Sheet: A separate file (`styles.css`) containing CSS rules linked to an HTML document.

Example:

html
Copy code

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

---

5. (b) WML Elements

WML Tables:
Used to display tabular data.
wml
Copy code

```
<table columns="2">
    <tr><td>Course</td><td>Instructor</td></tr>
    <tr><td>BCA</td><td>John</td></tr>
</table>
```

1.

Anchor Element:
Provides navigation links in WML.
wml
Copy code
```
<anchor title="Next">Go to next page</anchor>
```

2.

Answers for BCS-053 Term-End Examination, December 2023

---

1. (a) What is WAP and How it Works?

WAP (Wireless Application Protocol):

- WAP is a protocol for accessing information over a mobile wireless network.
- It allows mobile devices to access web content formatted for smaller screens.

WAP Model:

1. **Mobile Device**: Sends a request using WAP.
2. **WAP Gateway**: Converts WAP requests into HTTP requests.
3. **Web Server**: Processes the HTTP request and sends the response back.
4. **WAP Gateway**: Converts HTTP response into WAP format.
5. **Mobile Device**: Displays the response.

Diagram:

rust
Copy code
```
Mobile Device -> WAP Gateway -> Web Server
```

---

1. (b) Uses of HTML Elements in Form Design

`<form>`: Defines a form for user input.
html
Copy code
```
<form action="/submit" method="POST"></form>
```

1.

<input>: Specifies an input field.
html
Copy code
```
<input type="text" name="username">
```

2.

<label>: Provides a label for an input field.
html
Copy code
```
<label for="username">Username:</label>
```

3.

<textarea>: Creates a multi-line text input field.
html
Copy code
```
<textarea name="comments"></textarea>
```

4.

<button>: Defines a clickable button.
html
Copy code
```
<button type="submit">Submit</button>
```

5.

---

1. (c) Document Object Model (DOM)

Definition: The DOM is a programming interface for HTML and XML documents, representing the structure of a document as a tree of nodes.

HTML DOM Node Tree:

lua
Copy code
```
Document
  |
  +-- <html>
       |
       +-- <head>
```

```
|    |
|    +-- <title>
|
+-- <body>
     |
     +-- <h1>
     +-- <p>
```

---

1. (d) AJAX vs. JavaScript

AJAX (Asynchronous JavaScript and XML):

- A technique to fetch data from a server without refreshing the page.

Differences:

Feature	JavaScript	AJAX
Definition	A scripting language for dynamic web content.	A technique using JavaScript for asynchronous server communication.
Reloading	Requires page reload.	No page reload.
Data Format	Works with static data.	Can handle dynamic data (JSON, XML).

---

1. (e) CSS and Embedded Style Sheets

Definition: CSS (Cascading Style Sheets) is used to control the style of HTML elements.

Embedded Style Sheets Example:

html
Copy code

```
<!DOCTYPE html>
<html>
<head>
    <style>
        body {
            background-color: lightblue;
```

```
        }
        h1 {
            color: navy;
        }
    </style>
</head>
<body>
    <h1>Hello, World!</h1>
</body>
</html>
```

---

2. (a) What is DTD and Its Role in XML

Definition: DTD (Document Type Definition) defines the structure and rules of an XML document.

Role:

- Ensures data consistency.
- Validates the structure of XML documents.

---

2. (b) Server-Side Scripting vs. Client-Side Scripting

Server-Side Scripting:

- Code executed on the server (e.g., JSP, PHP).
- Secure and used for database interactions.

Client-Side Scripting:

- Code executed on the browser (e.g., JavaScript).
- Handles UI interactions.

---

3. (a) Cookies in JSP

Definition: Cookies store small pieces of data on the client side.

JSP Code to Create and Set Cookie Lifetime:

jsp
Copy code

```jsp
<%
Cookie userCookie = new Cookie("username", "JohnDoe");
userCookie.setMaxAge(60 * 60 * 24); // 1 day
response.addCookie(userCookie);
%>
```

---

3. (b) WML Program for Username and Password

wml
Copy code

```wml
<card id="login" title="Login">
    <p>Username: <input name="username" /></p>
    <p>Password: <input type="password" name="password" /></p>
</card>
```

---

3. (c) GET vs. POST in HTTP

Method	Description
GET	Sends data via URL; less secure.
POST	Sends data via request body; secure.

---

4. (a) Differences

1. **Include Directive vs. Include Action**
    - **Include Directive**: Includes content at compile time.
    - **Include Action**: Includes content at request time.
2. **Model vs. View in MVC**
    - **Model**: Handles business logic and data.
    - **View**: Manages UI and presentation.

---

4. (b) WML Elements

<optgroup>: Groups related options in a menu.
wml
Copy code
```
<select>
    <optgroup title="Fruits">
        <option value="apple">Apple</option>
    </optgroup>
</select>
```

1.

<setvar>: Sets a variable's value.
wml
Copy code
```
<setvar name="username" value="JohnDoe" />
```

2.

---

4. (c) HTML Box Model

Diagram:

css
Copy code
```
Content -> Padding -> Border -> Margin
```

Explanation:

- **Content**: Actual text or images.
- **Padding**: Space between content and border.
- **Border**: Edge of the element.
- **Margin**: Space between elements.

---

5. (a) JSP Program to Fetch Student Records

jsp
Copy code
```
<%
```

```
String query = "SELECT name, programme, course_code, grade FROM
students";
ResultSet rs = stmt.executeQuery(query);
while (rs.next()) {
    out.println("<p>" + rs.getString("name") + ", " +
rs.getString("programme") + ", " +
                rs.getString("course_code") + ", " +
rs.getString("grade") + "</p>");
}
%>
```

---

5. (b) `getElementById()` in JavaScript

Definition: Fetches an HTML element by its ID.

Example:

html
Copy code
```
<p id="demo">Hello!</p>
<script>
    document.getElementById("demo").innerHTML = "Welcome!";
</script>
```

Answers for BCS-053 Term-End Examination, June 2024

---

1. (a) JSP Program Using `<jsp:forward>` and `<jsp:param>`

Code Example:

jsp
Copy code
```
<jsp:forward page="target.jsp">
    <jsp:param name="username" value="JohnDoe" />
    <jsp:param name="userId" value="12345" />
</jsp:forward>
```

Explanation:

- `<jsp:forward>`: Redirects the request to another JSP or servlet.
- `<jsp:param>`: Passes parameters to the forwarded page.

---

1. (b) Safe Methods in HTTP

Definition: A safe method does not alter the state of the server. Examples: `GET`, `HEAD`.

Is `GET` Safe? Yes, because it only retrieves data without modifying resources on the server. It is idempotent, meaning multiple requests yield the same result.

---

1. (c) Error Handling in JSP

Explanation: Errors in JSP can be handled using:

- **Error Pages**: Configure error handling in `web.xml`.
- **Try-Catch Blocks**: Handle exceptions in scripts.

Program Fragment:

jsp
Copy code

```
<%@ page errorPage="error.jsp" %>
<%
try {
    int result = 10 / 0;
} catch (ArithmeticException e) {
    out.println("An error occurred: " + e.getMessage());
}
%>
```

---

1. (d) Why XML and Its Advantages Over HTML

Why XML? XML (Extensible Markup Language) is used for data representation and storage in a structured format.

Advantages Over HTML:

1. **Self-descriptive**: Data is stored with tags describing the content.
2. **Separation of Data and Presentation**: XML focuses on data, while HTML is presentation-oriented.
3. **Extensibility**: Custom tags can be created.
4. **Platform Independent**: XML is widely supported across platforms.

---

1. (e) Web Container and Examples

Definition: A web container is a runtime environment that manages the lifecycle, security, and interactions of servlets and JSPs.

Examples:

1. Apache Tomcat
2. Jetty
3. GlassFish

---

2. (a) JSP Implicit Objects

Definition: Implicit objects are pre-defined objects in JSP, accessible without explicitly declaring them.

Examples:

`request`: Provides information about the HTTP request.
jsp
Copy code
```
out.println("Client IP: " + request.getRemoteAddr());
```

1.

`session`: Tracks user sessions across multiple requests.
jsp
Copy code
```
session.setAttribute("username", "JohnDoe");
```

2.

---

2. (b) Differences

1. **Static vs. Dynamic Web Pages**
    - **Static Pages**: Content remains fixed; written using HTML.
    - **Dynamic Pages**: Content is generated dynamically using server-side scripting (e.g., JSP, PHP).
2. **2-Tier vs. 3-Tier Architecture**
    - **2-Tier**: Direct communication between client and database.
    - **3-Tier**: Includes a middle layer (server) between client and database for added logic.

---

3. (a) WML Navigation Elements

Explanation: WML uses elements like `<card>` and `<a>` for navigation.

Example Program:

wml
Copy code

```
<card id="main" title="Main Menu">
    <p>Welcome!</p>
    <a href="#next">Go to Next Page</a>
</card>
<card id="next" title="Next Page">
    <p>This is the next page.</p>
</card>
```

---

3. (b) HTML Program Using `<details>` and `<summary>`

HTML Code:

html
Copy code

```
<details>
    <summary>Click to see more</summary>
    <p>This is additional content revealed upon clicking.</p>
</details>
```

Explanation:

- `<details>`: Creates a collapsible section.
- `<summary>`: Defines the heading that users click to expand or collapse.

---

4. Explanations

1. **WAP Protocol Stack**
    - Layers: Application, Session, Transaction, Security, Transport.
    - Enables communication between mobile devices and web servers.
2. **Deployment Descriptor**
    - XML file (`web.xml`) that configures a web application.

xml
Copy code

```
<servlet>
    <servlet-name>HelloServlet</servlet-name>
    <servlet-class>HelloWorld</servlet-class>
</servlet>
```

3.
4. **XML XSLT**
    - Transforms XML documents into different formats (e.g., HTML, XML).

xml
Copy code

```
<xsl:template match="/">
    <html><body><h1><xsl:value-of select="title"/></h1></body></html>
</xsl:template>
```

5.
6. **JSP Scriptlets**
    - Embeds Java code in JSP pages.

jsp
Copy code

```
<% out.println("Hello, World!"); %>
```

7.

---

5. (a) Features of Web Technologies

1. **Mashups**: Combines multiple APIs to create a new service.
2. **Rich Internet Applications (RIA)**: Provides desktop-like experience in browsers (e.g., Google Maps).
3. **Web Services**: Enables interoperability between applications over the web using SOAP/REST.

---

5. (b) JSP Program to Print Odd Numbers and Sum

Code:

jsp
Copy code

```
<%
int sum = 0;
for (int i = 1; i <= 100; i += 2) {
    out.println(i + "<br>");
    sum += i;
}
out.println("Sum: " + sum);
%>
```