SMTP	POP3
Simple Mail Transfer Protocol	Post Office Protocol
It is used for sending messages	It is used for accessing messages
It is Message Transfer Agent	It is Message Access Agent
It is a stateful protocol	It is a stateless protocol
It doesn't store emails	It allows users to download emails from the server

DOM	SAX
Document Object Model	Simple API for XML
It is faster	It is slower
It has a tree structure	It is event-based
It can insert or delete nodes	It is read-only
Backward and forward search is possible	Backward and forward search is not possible
Suitable for large XML document	Suitable for memory
It loads whole XML documents in memory	It loads small part of the XML file in memory

Stateful Session	Stateless Session
Maintains state across client requests	Does not maintain client state
Session data stored on server	No session data stored
Identified by session ID	Not identified
Less scalable	More scalable
Requires more server resources	Requires less server resources
More susceptible to server failures	Less susceptible to server failures
Complex implementation	Simple implementation

ODBC	JDBC
Open Database Connectivity	Java database connectivity
Introduced by Microsoft	Introduced by SUN Micro Systems
We can use ODBC for any language	We can use JDBC only for Java language
ODBC is procedural	JDBC is object-oriented
We can choose ODBC only on Windows platform	We can use JDBC on any platform
It is not recommended to use ODBC for Java applications	It is highly recommended to use JDBC for Java applications

Explain and provide the syntax of various tags used in HTML

Tag	Description	Syntax
<html>	Root element of the HTML document	<html> ... </html>
<head>	Contains information about the document	<head> ... </head>
<title>	Defines the title of the webpage	<title>My Website Title</title>
<meta>	Provides meta information about the document	<meta name="description" >
<body>	Contains the content that will be displayed in the browser window	<body> ... </body>
<h1> ... <h6>	Define headings of different sizes	<h1>Heading 1</h1> ... <h6>Heading 6</h6>
<p>	Defines a paragraph of text	<p>This is a paragraph.</p>
<b>	Bolds text	<b>Bold text</b>
<i>	Italicizes text	<i>Italicized text</i>
<ul>	Creates an unordered list	<ul></ul>
<ol>	Creates an ordered list	<ol></ol>
<li>	Defines a list item	<li>Item 1</li>
<a>	Defines a hyperlink	<a href="https://example.com">Click</a>
<img>	Inserts an image	<img src="image.jpg">
<table>	Creates a table	<table></table>
<tr>	Defines a table row	<tr></tr>
<th>	Defines a table heading	<th></th>
<td>	Defines a table cell	<td>Cell 1</td>
<form>	Defines a form for user input	<form></form>
<br>	Inserts a line break	<br>

How is JavaScript useful in Dynamic HTML?

1. **Dynamic Content:** JavaScript allows the modification of HTML elements in real-time.

2. **DOM Manipulation:** JavaScript provides access to the Document Object Model (DOM)

3. **User Interactions:** JavaScript enables the handling of various user events, such as clicks, key presses, etc.

4. **AJAX** (Asynchronous JavaScript and XML): JavaScript facilitates asynchronous requests to the server

5. **CSS Effects:** JavaScript can be used to dynamically apply and manage CSS styles

6. **Form Validation:** JavaScript enables the validation of user inputs in forms

Discuss about the following with examples:

(a) CSS

CSS is a style sheet language used for describing the presentation of a document written in HTML or XML. CSS defines how elements should be displayed, including layout, colours, fonts, etc. Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>CSS Example</title>
  <style>
    /* CSS styles for the paragraph */
    p {
      color: blue;
      font-size: 18px;
      text-align: center;
    }
  </style>
</head>
<body>
  <p>This is a styled paragraph.</p>
</body>
</html>
```

(b) IMAP

The IMAP (Internet Message Access Protocol) is a protocol for receiving emails from a server. It allows access to emails from multiple locations simultaneously. It doesn't download the entire email until the recipient opens it.

Advantages of IMAP

- Not affected by local machine failures since the emails are stored in the server.
- Can be accessed through multiple devices at the same time.
- It supports keyboard-based email searching.
- Local storage use is minimal.

Example:

```
IMAP Command: LOGIN user@example.com password
Server Response: OK LOGIN completed
```

Design a simple JSP page to display current date and time from system

```
<%@ page import="java.util.Date" %>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Current Date and Time</title>
</head>
<body>
    <h1>Current Date and Time</h1>
    <p>The current date and time is: <%= new Date() %></p>
</body>
</html>
```

Write steps to install TOMCAT server in your machine

1. Visit https://tomcat.apache.org/
2. On the left hand side, under download section, click in version the desired version
3. Click on 64 bit Windows installer. Your download will start
4. A set up window will appear
5. Accept the license agreement. Click next.
6. Finish the installation

What are XML file? Show the use of XML in JSP pages

XML stands for Extensible Markup Language. It is a text-based format for storing and exchanging structured data.

Uses of XML in JSP:

1. Reading and parsing XML data
2. Displaying XML data
3. Transforming XML data
4. Validating XML data

Explain the important features of a servlet

1. **Platform-Independence:** They can run on any platform that supports the JVM

2. **Java EE Integration:** They integrate with other Java EE technologies, such as JSP, JDBC, EJB, etc.

3. **Multithreading Support:** Each servlet instance can handle multiple requests concurrently using separate threads.

4. **HTTP Protocol Support:** They can handle HTTP requests and generate HTTP responses

5. **Request and Response Handling:** They process client requests and generate dynamic responses.

6. **Reusable Components:** Servlets can be reused across multiple web applications

7. **Community Support:** There is extensive community support for developers working with servlets.

Discuss the various security issues and their solutions in web programming in detail

- **SQL Injection:** Malicious code is injected into database queries

  Solution: Avoid dynamic SQL queries.

- **Cross-Site Scripting (XSS):** Malicious scripts are injected into web pages

  Solution: Encode all user input before displaying it on the web page

- **Weak passwords:** Users often choose easy or reused passwords

  Solution: Use strong passwords and implement multi-factor authentication

- **Session hijacking:** Attackers steal user's session cookie to impersonate the user and access sensitive information.

  Solution: Use secure cookies with the `HTTPS` flag

- **Unprotected databases:** Attackers may gain unauthorized access to databases containing sensitive data.

  Solution: Use strong database passwords and encrypt database files

- **Outdated software and libraries:** Outdated software and libraries often contain known vulnerabilities that attackers can exploit.

  Solution: Regularly update all software and libraries, and avoid using outdated software.

Explain the steps and important functions to perform Database Programming using JDBC. Is there any other method to provide the database connectivity? If yes, kindly elaborate the method in detail.

1. **Load the JDBC Driver:** Load the JDBC driver specific to the target database using the Class.forName() method.
2. **Establish a Connection:** Create a connection to the database using the DriverManager.getConnection() method. This method requires the database URL, username, and password.
3. **Create Statements:** Use the Connection object to create statements for executing SQL queries.

4. **Execute Queries:** Execute the SQL query using
   - executeQuery() for SELECT statements
   - executeUpdate() for INSERT, UPDATE, and DELETE statements
   - execute() for stored procedures.

5. **Process Results:** For SELECT queries, use the ResultSet object to access and process the retrieved data.

6. **Close Resources:** Close all resources in reverse order of creation using close() method

ODBC (Open Database Connectivity) is a general-purpose API that allows applications to interact with various relational databases.

1. **Connect to the Database:** Connect to the database using SQLConnect() method

2. **Allocate Memory:** Allocate memory for statements and buffers using methods like SQLAllocStmt() and SQLAllocEnv()

3. **Build and Execute SQL Statements:** Build your SQL query and execute the prepared statement using the SQLExecDirect() function

4. **Process Results:** Process the result using methods like SQLFetch() and SQLRowCount()

5. **Close Resources:** Close all resources using SQLDisconnect method

Write a program, where illustration of JAVA Beans takes place. Explain the various features of Java beans in Web programming.

```
import java.io.Serializable;

public class PersonBean implements Serializable {

    private String name;
    private int age;

    // Default constructor
    public PersonBean() {
        name = "John Doe";
        age = 25;
    }

    // Getter and Setter methods for 'name'
    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
    // Getter and Setter methods for 'age'
    public int getAge() {
        return age;
    }

    public void setAge(int age) {
        this.age = age;
    }
}
```

Features of JavaBeans:

1. **Serializable:** They implement the Serializable interface, making them suitable for storage or transmission.

2. **No-argument Constructor:** They provide a no-argument constructor

3. **Getter and Setter Methods:** They provide getter and setter methods

4. **Reusable:** They are reusable components, promoting modular and maintainable code

5. **Events and Listeners:** They support the concept of events and listeners

6. **Customization:** They allow developers to customize bean properties

7. **Naming Conventions:** They follow naming conventions for methods and properties

javax.Servelet HTTP Package

It provides a set of classes and interfaces for implementing servlets that handle HTTP requests and responses. Some important functions are:

- doGet(): Handles HTTP GET request
- doPost(): Handles HTTP POST request
- getMethod(): Get the HTTP method used
- getParameter(): Get the parameter value from the request body
- setContentType(): Set the content type of the response.
- getWriter(): Get a writer to write text data to the response body.
- getAttribute(): Get the value of an attribute from the session
- setAttribute(): Store an attribute in the session

BDK

Bean Developer Kit (BDK) is a software toolkit developed by Oracle to facilitate the development of JavaBeans components. Key components include:

- **BeanBox:** A visual application for testing and interacting with JavaBeans
- **Example Code:** Provides a collection of source code demonstrating various functionalities of JavaBeans
- **Design Patterns:** Offers implementations of common design patterns used in JavaBeans
- **Validation Tools:** Tools to ensure JavaBeans properties meet specific requirements
- **Debugging Tools:** Tools to identify and fix problems in JavaBeans code
- **Documentation:** Documentation for BeanBox, example code, and other features