| SMTP | HTTP |
|---|---|
| Simple Mail Transfer Protocol | Hypertext Transfer Protocol |
| It is used for mail services | It is used for file transfer |
| Push protocol | Pull protocol |
| Smaller message size limit | Larger message size limit |
| Requires authentication | Doesn't require authentication |
| It uses port 25 | It uses port 80 |
| Doesn't support session management or cookies | Supports session management and cookies |

**Discuss the benefits of JavaScript in Dynamic HTML.**

1. **Engaging User Interaction:** Real-time feedback, animations, and interactive forms

2. **Scalable Performance:** Reduces server load by handling tasks locally

3. **Simplified Development:** Beginner-friendly language with extensive resources

4. **Dynamic Content Updates:** Refresh content without reloading entire pages

5. **Responsive User Interface:** Capture user actions and respond dynamically

6. **Flexible Page Structure:** Real-time manipulation of document structure

7. **Secure and Reliable Forms:** Validate user input before submission to the server

**Design a website for Online Air ticket reservation. The website should contain the report Generation if user wants to check his itinerary on yearly basis. Website should also contain the important features on any airline management system.**

**index.html**

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Airline Reservation System</title>
    <link rel="stylesheet" type="text/css" href="styles.css">
</head>
<body>

<header>
    <h1>Airline Reservation System</h1>
</header>

<nav>
    <a href="#home">Home</a>
    <a href="#search">Search & Book</a>
```

```html
    <a href="#dashboard">Dashboard</a>
    <a href="#reports">Reports</a>
    <a href="#admin">Admin</a>
</nav>

<section>
    <h2>Welcome to our Airline Reservation System</h2>
    <p>Get premium luxury</p>
</section>

<footer>
    &copy; 2023 Airline Reservation System. All rights reserved.
</footer>

</body>
</html>
```

## style.css

```css
body {
    font-family: Arial, sans-serif;
    margin: 0;
    padding: 0;
}

header {
    background-color: #3498db;
    color: #fff;
    padding: 10px;
    text-align: center;
}

nav {
    background-color: #2c3e50;
    color: #fff;
    padding: 10px;
}

nav a {
    color: #fff;
    text-decoration: none;
    padding: 10px;
    margin: 0 10px;
}

section {
    padding: 20px;
}

footer {
    background-color: #34495e;
    color: #fff;
    text-align: center;
    padding: 10px;
```

```css
    position: fixed;
    width: 100%;
    bottom: 0;
}
```

1. **Reusability:** Components can be easily reused across different parts of a website.

2. **Ease of Integration:** JavaBeans integrate seamlessly into various frameworks.

3. **Component-Based Development:** Promotes modular development by combining and reusing components.

4. **Platform Independence:** JavaBeans are platform-independent

5. **Event Handling:** Supports event handling, enhancing responsiveness.

6. **Serialization:** Supports easy saving and restoration of component states.

7. **Encapsulation:** Promotes encapsulation for secure and maintainable web applications.

Benefits of javax.sql package for database:

1. It offers a more flexible and standardized mechanism compared to directly using DriverManager in JDBC.
2. It allows applications to efficiently manage connections in a pooled environment.
3. It allows coordination of transactions across multiple databases
4. It allows applications to receive notifications about connection events
5. It simplifies working with data in a tabular form.

JDBC is needed for complete utilization of javax.sql package

1. **Import JDBC Library:** Include the JDBC library in your JSP page using the `page` directive.

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

2. **Establish Database Connection:** Create a database connection using `DriverManager.getConnection` with the appropriate database URL, username, and password.

```jsp
<%
    String url = "jdbc:mysql://localhost:3306/your_database";
    String username = "your_username";
    String password = "your_password";

    Connection connection = DriverManager.getConnection(url, username,
password);
```

```
%>
```

3. **Execute SQL Queries:** Create and execute SQL queries using `Statement` or `PreparedStatement` objects.

```
<%
    Statement statement = connection.createStatement();
    ResultSet resultSet = statement.executeQuery("SELECT * FROM
your_table");
%>
```

4. **Process Results:** Iterate through the `ResultSet` to retrieve and display data.

```
<%
    while(resultSet.next()) {
        String columnName = resultSet.getString("column_name");
    }
%>
```

5. **Close Database Connection:** Close the database connection to release resources.

```
<%
    resultSet.close();
    statement.close();
    connection.close();
%>
```

Write the steps to read servlet parameters through a webpage

1. **Access Servlet Parameters:** Use the `request` object to access parameters sent by a servlet.

```
<%
    String parameterValue = request.getParameter("parameter_name");
%>
```

2. **Display Parameter Value:** Display the parameter value within your JSP page.

```
<p>Parameter Value: <%= parameterValue %></p>
```

Write short note on XML processor

An XML processor is a software program capable of reading, parsing, manipulating, and validating XML documents.

Key functions:

- **Parsing:** Converts XML document into a structured representation
- **Validation:** Checks the XML document for syntax and structural errors

- **Manipulation:** Allows manipulation of content within the XML document

- **Transformation:** Can transform XML documents into other formats

- **Querying:** Enables querying XML documents to extract specific data

Benefits of using XML processors:

- **Data handling:** Provide a structured and efficient way to work with XML documents

- **Data integration:** Facilitate data integration between different applications

- **Platform Independence:** Usable across different platforms