<h1 style="text-align:center">Practical No 5</h1>

Aim: Write a Java program to implement mail validation using

Java Script

Resources Required:

- · Java Development Kit (JDK)

- · Text Editor (e.g., Notepad++) or IDE (e.g., Eclipse, IntelliJ IDEA, NetBeans)

- · Command-line terminal or Java compiler

JavaScript is a high-level, interpreted, object-oriented programming language primarily used for adding interactivity and dynamic behavior to websites. It was developed by Brendan Eich in 1995 at Netscape and has since become one of the core technologies of the web, alongside HTML and CSS.

Features of JavaScript

- **Client-Side Execution**: Runs directly in the user's browser without needing server interaction.

- **Interpreted Language**: Does not require compilation; executed line by line.

- **Lightweight & Fast**: Designed for quick execution inside web pages.

- **Object-Oriented**: Uses objects and prototypes (not classical classes like Java, though ES6 introduced class syntax).

- **Event-Driven**: Reacts to user actions like clicks, inputs, mouse movements, etc.

- **Platform Independent**: Runs on any operating system and browser.

- **Case-Sensitive**: Identifiers (variable names, functions) are case-sensitive.

Table: Explanation of Syntax Used in Program

Syntax / Keyword	Meaning / Use
`<!DOCTYPE html>`	Declares the document type as HTML5.
`<html> ... </html>`	Root element of the HTML document.
`<head> ... </head>`	Contains page metadata, title, and script code.
`<body> ... </body>`	Contains visible content (button for email input).
`<script> ... </script>`	Section where JavaScript code is written.
`function validateEmail()`	Declares a user-defined function named validateEmail.
`{ ... }` (curly braces)	Groups multiple statements inside a function or control block.
`let email = prompt()`	Declares a variable email and assigns input value entered by the user.
`const emailPattern = ...`	Declares a constant variable to store the regex (cannot be changed).

Syntax / Keyword	Meaning / Use
/ˆ[ˆ\s@]+@[ˆ\s@]+\.[ˆ\s@]+$/	Regular Expression (Regex) pattern for validating email format.
while (true)	Infinite loop that keeps executing until break is used.
if (condition)	Checks if a condition is true; executes block if true.
else { ... }	Executes alternative block if the if condition is false.
emailPattern.test(email)	Tests whether the input matches the regex pattern (returns true/false).
alert("message")	Displays a popup alert box with a message.
break	Exits the loop immediately.
<button onclick="...">	Creates a clickable button and calls a function when clicked.

Source Code:

```
<!DOCTYPE html>
<html>
<head>
 <title>Email Validation</title>
 <script>
  function validateEmail() {
   let email = prompt("Please enter your email address:");

   // Regex for simple email validation
   const emailPattern = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

   // Keep asking until a valid email is entered
   while (true) {
    if (emailPattern.test(email)) {
     alert("Email is valid: " + email);
     break;
    } else {
     alert(" Invalid email! Please enter a valid format (example: user@example.com).");
     email = prompt("Re-enter your email address:");
    }
   }
  }
 </script>
```

```
</head>
<body>
  <h2>Email Validation Example</h2>
  <button onclick="validateEmail()">Enter Email</button>
</body>
</html>
```

Output:

Conclusion:
This Java program successfully demonstrates the implementation of Email Validation using Java Scipt.