

Project Report: Regex Matcher Application

1. Introduction

Regular Expressions (Regex) are used to search and match patterns in text. They are widely used in data processing, validation, and text analysis. Tools like regex101 help users test and understand regular expressions.

In this project, a simple web application is created to replicate the **core functionality** of regex101, where a user can enter a test string and a regular expression to find matching patterns.

2. Objective

The main objectives of this project are:

- To take a test string as input from the user
 - To take a regular expression as input
 - To display all matched strings when the user clicks the submit button
 - To handle invalid regular expressions properly
-

3. Technology Stack Used

- **HTML** - Used for structuring the web page
 - **CSS** - Used for styling and improving the appearance
 - **JavaScript** - Used to process the regex logic and display results
-

4. Description of the Application

The application consists of:

- A textarea where the user enters the test string
- An input field where the user enters the regular expression
- A submit button to execute the regex
- An output section where matched strings or error messages are displayed

When the submit button is clicked, JavaScript uses the built-in `RegExp` object to find all matches in the test string.

5. Working of the Application

1. The user enters a test string and a regex pattern
 2. On clicking the submit button:
 - The inputs are validated
 - The regex is applied using the global (`g`) flag
 - All matched strings are extracted
 3. If matches are found, they are displayed as a list
 4. If no match is found, a proper message is shown
 5. If the regex is invalid, an error message is displayed
-

6. Sample Use Cases

- Extracting email addresses from text
- Finding phone numbers

- Matching dates or numeric values
 - Testing custom regex patterns
-

7. Screenshots

Screenshots of the following are included:

The image displays two screenshots of a web-based 'Regex Matcher' application. Both screenshots show a light blue header bar with the title 'Regex Matcher' in bold black font. Below the header is a white input form.

Screenshot 1 (Top): The 'Test String' field contains the text 'The events are scheduled on 01/01/2020 and 15/08/1947.' The 'Regular Expression' field contains the pattern '\b\d{2}/\d{2}/\d{4}\b'. The 'Matched Strings:' section lists two matches: '01/01/2020' and '15/08/1947'.

Screenshot 2 (Bottom): The 'Test String' field contains the text 'Contact us at support@gmail.com or admin@company.in for more info.' The 'Regular Expression' field contains the pattern '[a-zA-Z0-9._%+-]+@[a-zA-Z.-]+\.[a-zA-Z]{2,}'. The 'Matched Strings:' section lists two matches: 'support@gmail.com' and 'admin@company.in'.

- User input of test string and regex
- Display of matched results

- Error handling for invalid regex
-

8. Limitations

- Advanced features like regex explanation, highlighting, and multiple regex engines are not included
 - The project focuses only on the core matching functionality as per the task requirement
-

9. Conclusion

This project successfully demonstrates the basic working of a regex testing tool. By keeping the application simple and focused, the core concept of regular expressions is clearly implemented. The project helped in understanding regex usage, JavaScript string processing, and basic frontend development.

10. Future Scope

- Adding match count
- Highlighting matched text
- Supporting regex flags (i, m, etc.)