

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**Programme : B.Tech CSE**

**Year / Semester : III/V Section: CSE A & B**

**Course Name : Full Stack Web Development Course Code : BTCL3501**

**Faculty Name(s) and Designation : Dr. Stephen Dass A , Assistant Professor**

**Worksheet-5**

**Date: 02-09-2025 & 03-09-2025**

Full Stack Web Development Practice Worksheet  
Javascript basic and Regular Expression

---

# Javascript

## 1. Student Registration System

Scenario: A university website allows students to register with their details (Name, Roll No, Department). Create an HTML form and use a **JavaScript Constructor** Student(name, rollNo, dept) to create student objects. Display all registered students in a table format below the form.

*Hint:*

- Use a constructor to store student info.
- Push students into an array.
- Dynamically update the HTML table.

## 2. Online Bookstore

Scenario: Create a bookstore webpage with a list of books. Use a constructor Book(title, author, price) to represent each book. Display them in cards (using CSS). Add a button **"Add to Cart"** that shows selected books in a cart section.

*Hint:*

- Use CSS Grid/Flexbox for book cards.
- Cart updates dynamically using DOM + constructor-created objects.

## 3. Movie Ticket Booking

Scenario: Create a movie booking page. Use a constructor Movie(title, time, price). Display available movies with show timings. On clicking **"Book Now"**, add tickets to a cart section and calculate the total price.

*Hint:*

- Store movies as objects created using constructor.

- Use event listeners for booking.
- Display selected tickets in a styled cart.

#### 4. Employee Payroll System

Scenario: Create a payroll system page. Use constructor Employee(name, designation, salary) and add a method getAnnualSalary(). Create employees from an HTML form and display their annual salary in a styled table.

*Hint:*

- Form takes employee details.
- Use object methods inside constructor.
- Show calculated annual salary in output table.

#### 5. Online Shopping Product Catalog

Scenario: Create a product catalog webpage using constructor Product(name, category, price). Display products in a grid with CSS. On clicking "Buy", show product details in an alert box and also maintain a **shopping summary list** at the bottom.

*Hint:*

- Create multiple product objects.
- Display dynamically with JavaScript DOM.
- Use CSS for product cards (shadow, hover effect).

## Regular Expressions in JavaScript

#### 6. Email Signup Validation

Scenario: An online signup form requires users to enter a valid email ID.

Write a JavaScript program using regex to ensure:

- Username part can have letters, numbers, dots, or underscores.
- Must contain @ symbol.
- Domain name must be valid (e.g., gmail.com, yahoo.in).

#### 7. Password Strength Checker

Scenario: In a banking website, the password should meet security standards.

Use regex to check if the password:

- Has at least 8 characters.
- Contains one uppercase, one lowercase, one digit, and one special character.
- Show error messages if conditions fail.

#### 8. Username Validation

Scenario: A gaming website requires usernames.

Rules (use regex to validate):

- Only alphabets, digits, and underscores are allowed.
- Must start with a letter.
- Length: 5 to 15 characters.

## 9. Extracting Hashtags from Posts

Scenario: In a social media app, users type hashtags in posts (e.g., *"Love #coding #JavaScript #100DaysOfCode"*).

Use regex to extract all hashtags and display them as clickable links.

## 10. Date Validation

Scenario: A railway booking website requires users to enter travel date in **DD/MM/YYYY** format.

Write regex to ensure:

- Day is 01–31.
- Month is 01–12.
- Year is 1900–2099.