

<b>Ex. No. 7</b>	<b>Website Design with JQuery Action</b>
<b>Date of Exercise</b>	<b>15/10/2025</b>

**Objective:**

To design and implement an interactive web application using jQuery to demonstrate real-time style manipulation, animation, and functional web interactions.

**Description:**

This experiment focuses on the use of jQuery, a popular JavaScript library, to enhance web interactivity and styling. It demonstrates how jQuery simplifies HTML document manipulation, event handling, animation, and dynamic content updates. By applying jQuery functions, web applications can respond quickly to user actions and perform visual transitions smoothly. The experiment shows how to manage content dynamically and how to create visually engaging effects using various jQuery methods.

**Syntax:**

```
// Selecting an element  
$(selector).action();
```

```
// Example: Changing text or CSS  
$("#id").text("New Text");  
$("#id").css("color", "blue");
```

```
// Example: Event handling  
$("#button").click(function() {  
    alert("Button Clicked!");  
});
```

```
// Example: Animation  
$("#box").animate({ left: '250px', opacity: '0.5' }, 1000);
```

**Experiment 1: Budget Calculator App using jQuery****HTML (budget.html):**

```
<!DOCTYPE html>
<html>
<head>
  <title>Budget Calculator App</title>
  <link rel="stylesheet" href="budget.css">
  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
  <script src="budget.js" defer></script>
</head>
<body>
  <h1>Budget Calculator</h1>
  <div id="container">
    <label>Income:</label>
    <input type="number" id="income" placeholder="Enter income"><br>

    <label>Expense:</label>
    <input type="number" id="expense" placeholder="Enter expense"><br>

    <button id="add">Add</button>
    <button id="reset">Reset</button>

    <h3>Total Income: <span id="totalIncome">0</span></h3>
    <h3>Total Expense: <span id="totalExpense">0</span></h3>
    <h3>Remaining Balance: <span id="balance">0</span></h3>
  </div>
</body>
</html>
```

**CSS (budget.css):**

```
body {
  font-family: 'Segoe UI', sans-serif;
  background-color: #f4f4f4;
  text-align: center;
  padding-top: 30px;
}
```

```
#container {
  background: #fff;
  width: 400px;
```

```
margin: auto;
padding: 20px;
border-radius: 10px;
box-shadow: 0px 0px 10px rgba(0,0,0,0.2);
}
```

```
input {
  width: 80%;
  padding: 8px;
  margin: 10px;
  border: 1px solid #ccc;
  border-radius: 5px;
}
```

```
button {
  margin: 5px;
  padding: 10px 15px;
  border: none;
  background-color: #007bff;
  color: white;
  border-radius: 5px;
  cursor: pointer;
}
```

```
button:hover {
  background-color: #0056b3;
}
```

```
h3 {
  color: #333;
  margin-top: 10px;
}
```

**JavaScript (budget.js):**

```
$(document).ready(function(){
  let totalIncome = 0, totalExpense = 0;

  $("#add").click(function(){
    let income = parseFloat($("#income").val()) || 0;
    let expense = parseFloat($("#expense").val()) || 0;
```

```
totalIncome += income;
totalExpense += expense;

let balance = totalIncome - totalExpense;

$("#totalIncome").text(totalIncome);
$("#totalExpense").text(totalExpense);
$("#balance").text(balance);

$("#income").val("");
$("#expense").val("");
});

$("#reset").click(function(){
    totalIncome = totalExpense = 0;
    $("#totalIncome, #totalExpense, #balance").text(0);
});
});
```

## Code 2: Demonstration of jQuery Effects and CSS Manipulation

### HTML (effects.html):

```
<!DOCTYPE html>
<html>
<head>
  <title>jQuery Effects and Styles</title>
  <link rel="stylesheet" href="effects.css">
  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
  <script src="effects.js" defer></script>
</head>
<body>
  <h1>jQuery Effects Showcase </h1>
  <div id="box">Welcome to jQuery Effects!</div>

  <div class="buttons">
    <button id="show">Show</button>
    <button id="hide">Hide</button>
    <button id="toggle">Toggle</button>
    <button id="fadeIn">Fade In</button>
    <button id="fadeOut">Fade Out</button>
    <button id="fadeToggle">Fade Toggle</button>
```

```
<button id="slideUp">Slide Up</button>
<button id="slideDown">Slide Down</button>
<button id="slideToggle">Slide Toggle</button>
<button id="animate">Animate</button>
</div>
</body>
</html>
```

**CSS (effects.css):**

```
body {
  text-align: center;
  font-family: 'Poppins', sans-serif;
  background-color: #f0f9ff;
}
```

```
#box {
  width: 250px;
  height: 120px;
  line-height: 120px;
  margin: 30px auto;
  background-color: #ffca28;
  color: #333;
  font-weight: bold;
  border-radius: 15px;
  border: 2px solid #333;
}
```

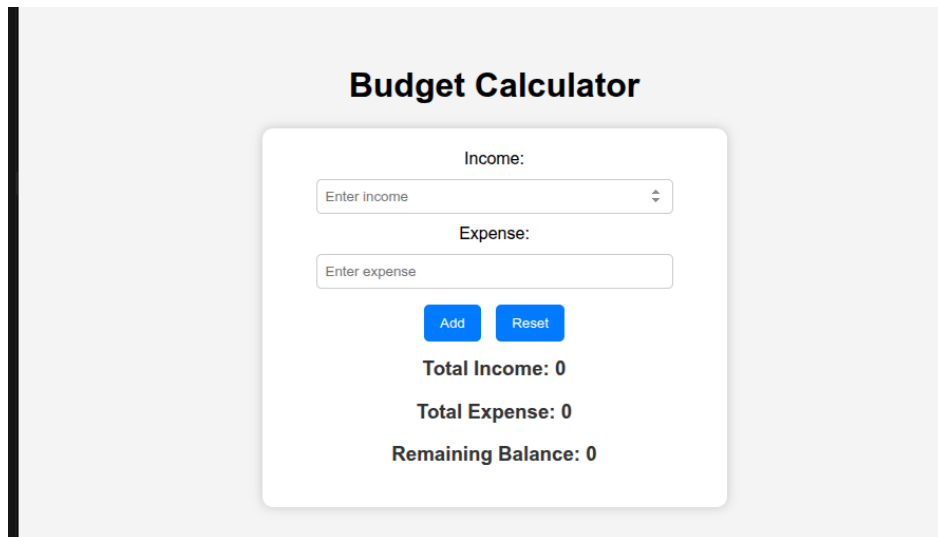
```
button {
  margin: 5px;
  padding: 10px 15px;
  background-color: #0066cc;
  color: white;
  border: none;
  border-radius: 5px;
  cursor: pointer;
}
```

```
button:hover {
  background-color: #004d99;
}
```

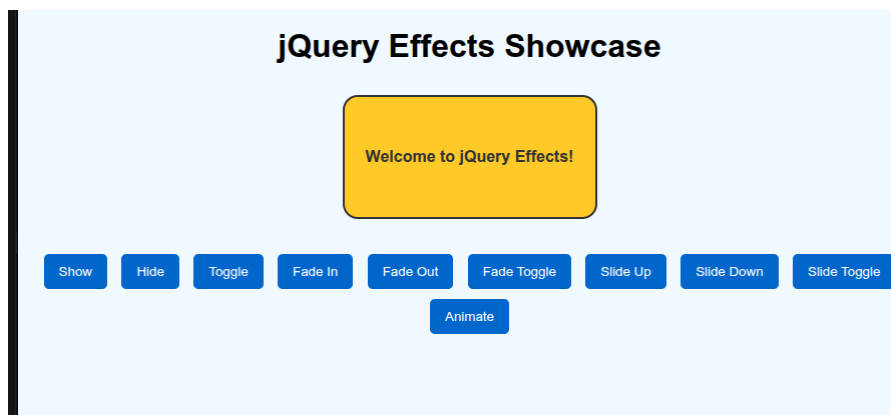
**JavaScript (effects.js):**

```
$(document).ready(function() {  
  $("#show").click() => $("#box").show();  
  $("#hide").click() => $("#box").hide();  
  $("#toggle").click() => $("#box").toggle();  
  $("#fadeIn").click() => $("#box").fadeIn();  
  $("#fadeOut").click() => $("#box").fadeOut();  
  $("#fadeToggle").click() => $("#box").fadeToggle();  
  $("#slideUp").click() => $("#box").slideUp();  
  $("#slideDown").click() => $("#box").slideDown();  
  $("#slideToggle").click() => $("#box").slideToggle();  
  $("#animate").click() => {  
    $("#box").animate({  
      width: "400px",  
      height: "200px",  
      opacity: 0.7,  
      borderRadius: "50px"  
    }, 800).animate({  
      width: "250px",  
      height: "120px",  
      opacity: 1,  
      borderRadius: "15px"  
    }, 800);  
  });  
});
```

## Output



The screenshot displays a web application titled "Budget Calculator". It features a central white card with a light gray border. At the top of the card, the title "Budget Calculator" is centered. Below the title, there are two input fields: "Income:" with a placeholder "Enter income" and a small upward arrow icon, and "Expense:" with a placeholder "Enter expense". Below these fields are two blue buttons labeled "Add" and "Reset". At the bottom of the card, the following text is displayed: "Total Income: 0", "Total Expense: 0", and "Remaining Balance: 0".



## Result

The experiment successfully demonstrated the implementation of jQuery for creating interactive and visually dynamic web applications. The functionalities proved how jQuery simplifies JavaScript coding while enhancing webpage responsiveness and design.