

**Output:**

### Areas of Different Shapes

Radius:

Area  
**50.26548245743669**

Calculate Reset

Length:  
  
Breadth:

Area  
**322**

Calculate Reset

Side A:  
  
Side B:  
  
Side C:

Area  
**321.99378875996973**

Calculate Reset

**Code:****1. HTML:**

```
<script type="text/javascript" src="areas.js"></script>
<link rel="stylesheet" href="style.css">

<body>

<h1> Areas of Different Shapes </h1>

<div class="shape-box">

  <div class="form-section">

    Radius: <input id="radiusOfCircle" placeholder=0 type="number"><br/>

    <div class="button-row">

      <button onclick="circle_area()"> Calculate </button>

      <button id="ResetCircle" onclick="reset()"> Reset </button>

    </div>

  </div>

</div>
```

## Experiment 1

Roll no: 42405

Batch: Q6

```
<!-- <canvas id="myCanvas" width="300" height="300" style="border:4px
solid #000000; border-radius: 10px;"></canvas> -->

    <div class="result-section">

        <div class="area-label">Area</div>

        <div class="area-box" id="AreaCircle">0</div>

    </div>
</div>
<br/>

<div class="shape-box">

    <div class="form-section">

        Length: <input id="lengthOfRect" placeholder=0 type="number"><br/>
        Breadth: <input id="breadthOfRect" placeholder=0 type="number"><br/>

        <div class="button-row">

            <button onclick="rect_area()"> Calculate </button>

            <button id="ResetRect" onclick="reset()"> Reset </button>

        </div>

    </div>

    <!-- <canvas id="myCanvas" width="300" height="300" style="border:4px
solid #000000; border-radius: 10px;"></canvas> -->

    <div class="result-section">

        <div class="area-label">Area</div>

        <div class="area-box" id="AreaRect">0</div>

    </div>
</div>
<br/>

<div class="shape-box">

    <div class="form-section">

        Side A: <input id="sideA" placeholder=0 type="number"><br/>
        Side B: <input id="sideB" placeholder=0 type="number"><br/>
```

## Experiment 1

Roll no: 42405

Batch: Q6

```
Side C: <input id="sideC" placeholder=0 type="number"><br/>

<div class="button-row">

    <button onclick="triangle_area()"> Calculate </button>

    <button id="ResetTriangle" onclick="reset()"> Reset </button>

</div>

</div>

<!-- <canvas id="myCanvas" width="300" height="300" style="border:4px
solid #000000; border-radius: 10px;"></canvas> -->

<div class="result-section">

    <div class="area-label">Area</div>

    <div class="area-box" id="AreaTriangle">0</div>

</div>

</div>

<br/>

</body>

</html>
```

### 2. CSS:

```
body {

    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

    background: linear-gradient(to right, #f3f4f6, #e0e7ff);

    margin: 0;

    padding: 20px;

    color: #333;

}

h1 {

    text-align: center;

    color: #1f2937;

    margin-bottom: 40px;
```

```
    font-size: 2.5rem;
    animation: fadeIn 1s ease;
}
```

```
.shape-box {
    background-color: #ffffff;
    max-width: 900px;
    margin: 20px auto;
    padding: 30px;
    border-radius: 20px;
    box-shadow: 0 6px 20px rgba(0, 0, 0, 0.1);
    transition: transform 0.3s ease;
    animation: slideUp 0.7s ease;
    display: flex;
    justify-content: space-between;
    align-items: center;
    gap: 30px;
}
```

```
.form-section {
    flex: 2;
    display: flex;
    flex-direction: column;
    gap: 10px;
}
```

```
.result-section {
    flex: 1;
    text-align: center;
    display: flex;
```

```
    flex-direction: column;
    align-items: center;
    gap: 10px;
}
```

```
.area-label {
    font-size: 1.2rem;
    color: #475569;
    font-weight: 600;
}
```

```
.area-box {
    background-color: #f1f5f9;
    padding: 20px 30px;
    border-radius: 12px;
    box-shadow: inset 0 2px 5px rgba(0, 0, 0, 0.05);
    font-size: 2rem;
    color: #1e40af;
    font-weight: bold;
    min-width: 120px;
    transition: all 0.3s ease;
}
```

```
input[type="number"] {
    padding: 10px;
    border: 2px solid #cbd5e1;
    border-radius: 10px;
    font-size: 1rem;
    width: 200px;
```

## Experiment 1

Roll no: 42405

Batch: Q6

```
    transition: border 0.3s;
}
```

```
input[type="number"]:focus {
    border-color: #6366f1;
    outline: none;
}
```

```
button {
    background-color: #6366f1;
    color: white;
    padding: 10px 20px;
    border: none;
    border-radius: 10px;
    cursor: pointer;
    font-size: 1rem;
    transition: background-color 0.3s, transform 0.2s;
}
```

```
button:hover {
    background-color: #4f46e5;
    transform: scale(1.05);
}
```

```
/* Animations */
```

```
@keyframes fadeIn {
    from { opacity: 0; transform: translateY(-20px); }
    to { opacity: 1; transform: translateY(0); }
}
```

```
@keyframes slideUp {  
    from { transform: translateY(30px); opacity: 0; }  
    to { transform: translateY(0); opacity: 1; }  
}
```

### 3. Javascript:

```
function circle_area() {  
    var radius = document.getElementById("radiusOfCircle").value;  
    if (radius < 0)  
        // Alert  
        return null;  
  
    let area = Math.PI * radius * radius;  
    console.log(area);  
    document.getElementById("AreaCircle").innerHTML = area;  
}
```

```
function rect_area() {  
    var length = document.getElementById("lengthOfRect").value;  
    var breadth = document.getElementById("breadthOfRect").value;  
  
    if (length < 0 || breadth < 0)  
        // Alert  
        return null;  
  
    let area = (length * breadth);  
    // Debug  
    console.log(area);  
  
    document.getElementById("AreaRect").innerHTML = area;
```

```
}
```

```
function triangle_area() {  
    var a = +document.getElementById("sideA").value;  
    var b = +document.getElementById("sideB").value;  
    var c = +document.getElementById("sideC").value;  
  
    // Semi Perimeter  
    console.log(a + " " + b + " " + c);  
    var s = (a + b + c) / 2;  
    var val = s * (s - a) * (s - b) * (s - c);  
    console.log(val);  
  
    if (val < 0) {  
        alert("Not a triangle !!");  
        return;  
    }  
    var area = Math.sqrt(val);  
    console.log(area);  
  
    document.getElementById("AreaTriangle").innerHTML = area;  
}
```

```
function reset() {  
    var id = "";  
    switch (event.target.id) {  
    case "ResetCircle":  
        id = "AreaCircle"  
        break;  
    case "ResetRect":
```



```
        id = "AreaRect"

        break;

    case "ResetTriangle":

        id = "AreaTriangle"

        break;

    default:

        alert("Nope");

    }

    document.getElementById(id).innerHTML = "-";

}
```