Roll no: 42405 Batch: Q6

Output:

Fig. Basic Website

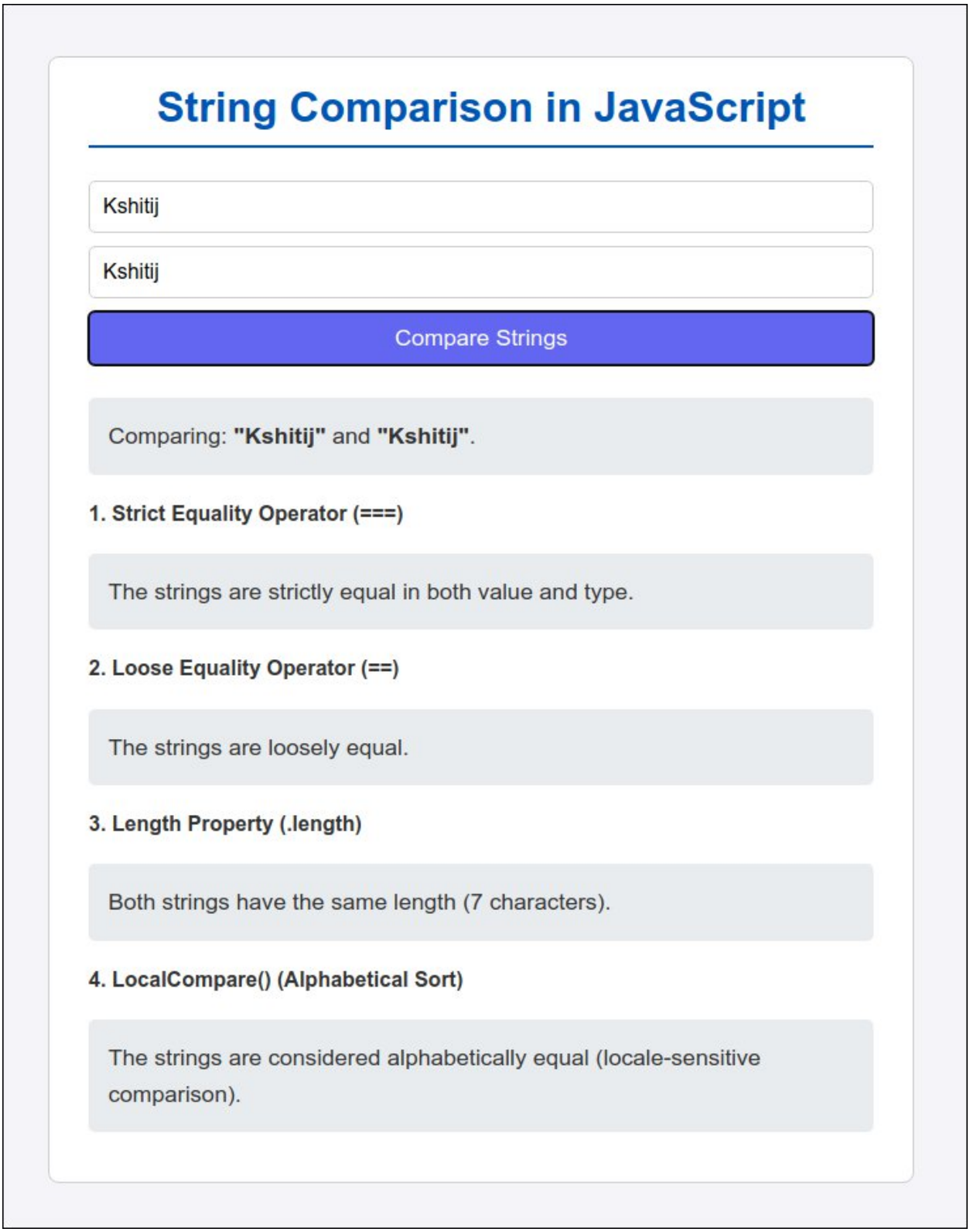

Fig. Equal Strings

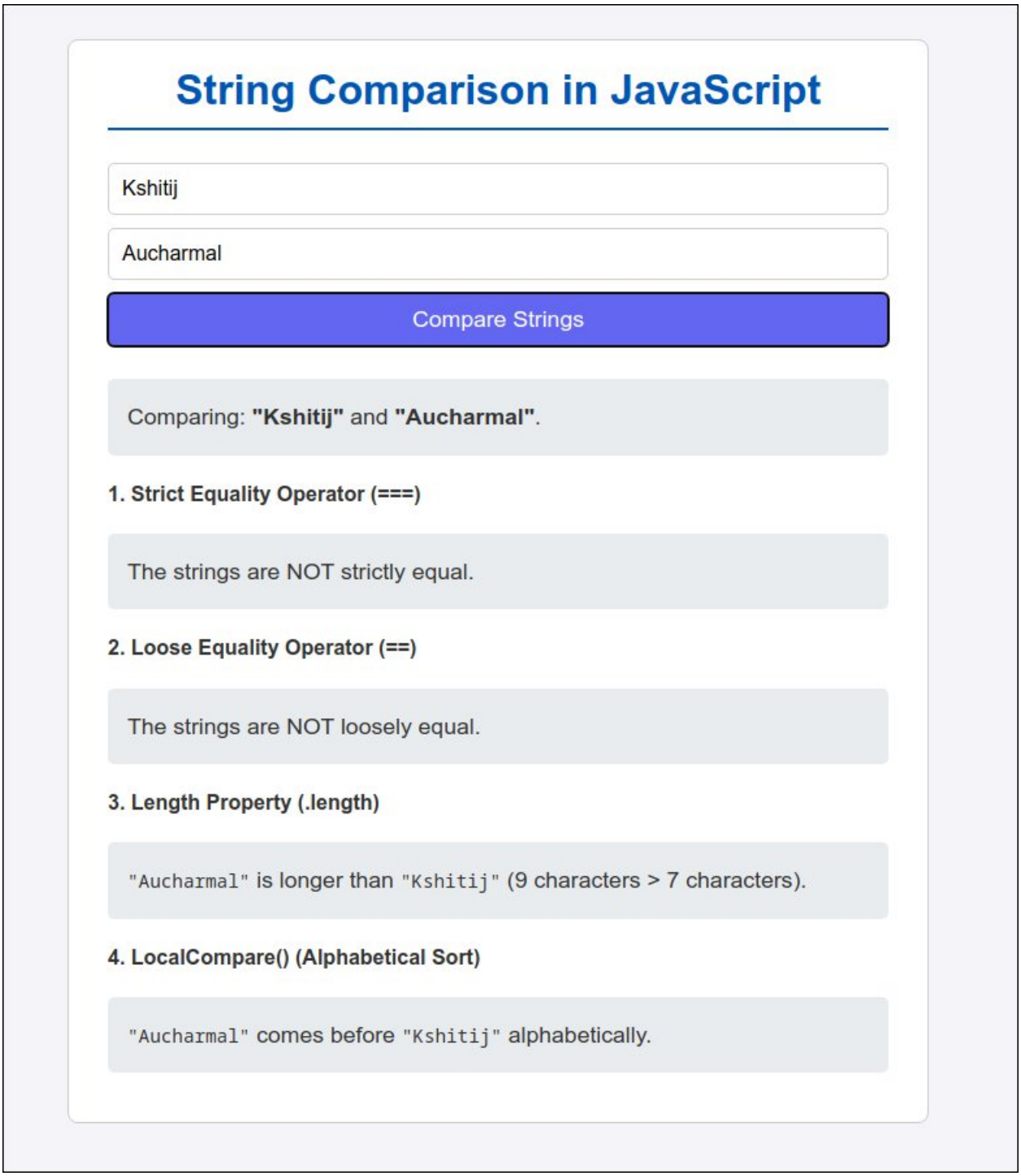

Fig. Not Equal

Experiment 4

Roll no: 42405 Batch: Q6

Code:

1. **HTML:**

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>String Comparison in JavaScript</title>
    <link rel="stylesheet" href="style.css" />
  </head>
  <body>
    <div class="container">
      <h1>String Comparison in JavaScript</h1>
      <div class="input-area">
        <input
          type="text"
          id="stringInput1"
          placeholder="Enter the first string..."
        />
        <input
          type="text"
          id="stringInput2"
          placeholder="Enter the second string..."
        />
        <button id="compareBtn" onclick="compareStrings()">
          Compare Strings
        </button>
      </div>
```

```
      <div id="results-area" class="hidden">
        <p id="comparisonIntro"></p>
        <h4>1. Strict Equality Operator (===)</h4>
        <p id="strictEqualityResult"></p>
        <h4>2. Loose Equality Operator (==)</h4>
        <p id="looseEqualityResult"></p>
        <h4>3. Length Property (.length)</h4>
        <p id="lengthResult"></p>
        <h4>4. LocalCompare() (Alphabetical Sort)</h4>
        <p id="localeResult"></p>
      </div>
    </div>

    <script>
      function compareStrings() {
        const string1 = document.getElementById("stringInput1").value;
        const string2 = document.getElementById("stringInput2").value;
        const resultsArea = document.getElementById("results-area"); //
Dev.logic: Guard clause to prevent comparison on empty input

        if (string1.trim() === "" || string2.trim() === "") {
          resultsArea.classList.add("hidden");
          return;
        }

        resultsArea.classList.remove("hidden");
        document.getElementById("comparisonIntro").innerHTML =
          `Comparing: <strong>"${string1}"</strong> and
<strong>"${string2}"</strong>.`; // 1. Strict Equality (===)
```

```
        let strictEqualityOutput;
        if (string1 === string2) {
          strictEqualityOutput = `The strings are strictly equal in both value
and type.`;
        } else {
          strictEqualityOutput = `The strings are NOT strictly equal.`;
        }
        document.getElementById("strictEqualityResult").innerHTML =
          strictEqualityOutput;

        // 2. Loose Equality (==)
        let looseEqualityOutput;
        if (string1 == string2) {
          looseEqualityOutput = `The strings are loosely equal.`;
        } else {
          looseEqualityOutput = `The strings are NOT loosely equal.`;
        }
        document.getElementById("looseEqualityResult").innerHTML =
          looseEqualityOutput; // 3. Length Compare

        let lengthOutput;
        const len1 = string1.length;
        const len2 = string2.length;
        if (len1 > len2) {
          lengthOutput = `<code>"${string1}"</code> is longer than
<code>"${string2}"</code> (${len1} characters > ${len2} characters).`;
        } else if (len1 < len2) {
```

```
          lengthOutput = `<code>"${string2}"</code> is longer than
<code>"${string1}"</code> (${len2} characters > ${len1} characters).`;

        } else {

          lengthOutput = `Both strings have the same length (${len1}
characters).`;

        }

        document.getElementById("lengthResult").innerHTML = lengthOutput; // 4.
localCompare



        let localeOutput; // Dev.logic: Returns negative, positive, or zero
based on alphabetical order.

        const comparisonResult = string1.localeCompare(string2);

        if (comparisonResult < 0) {

          localeOutput = `<code>"${string1}"</code> comes before
<code>"${string2}"</code> alphabetically.`;

        } else if (comparisonResult > 0) {

          localeOutput = `<code>"${string2}"</code> comes before
<code>"${string1}"</code> alphabetically.`;

        } else {

          localeOutput = `The strings are considered alphabetically equal
(locale-sensitive comparison).`;

        }

        document.getElementById("localeResult").innerHTML = localeOutput;

      }

    </script>

    w

  </body>

</html>
```

2. **Javascript:**

```
function compareStrings() {

  const string1 = document.getElementById("stringInput1").value;
```

```
  const string2 = document.getElementById("stringInput2").value;

  const resultsArea = document.getElementById("results-area");


  if (string1.trim() === "" || string2.trim() === "") {

    resultsArea.classList.add("hidden");

    return;

  }

  resultsArea.classList.remove("hidden");

  let equalityOutput;

  if (string1 === string2) {

    equalityOutput = `The strings are strictly equal.`;

  } else {

    equalityOutput = `The strings are NOT strictly equal.`;

  }

  document.getElementById("equalityResult").innerHTML = equalityOutput;


  let lengthOutput;

  const len1 = string1.length;

  const len2 = string2.length;

  if (len1 > len2) {

    lengthOutput = `<code>string1</code> is longer than <code>string2</code>
(${len1} > ${len2}).`;

  } else if (len1 < len2) {

    lengthOutput = `<code>string2</code> is longer than <code>string1</code>
(${len2} > ${len1}).`;

  } else {

    lengthOutput = `Both strings have the same length (${len1}).`;

  }

  document.getElementById("lengthResult").innerHTML = lengthOutput;
```

```
  let localeOutput;

  const comparisonResult = string1.localeCompare(string2);

  if (comparisonResult < 0) {

    localeOutput = `<code>"${string1}"</code> comes before
<code>"${string2}"</code> alphabetically.`;

  } else if (comparisonResult > 0) {

    localeOutput = `<code>"${string2}"</code> comes before
<code>"${string1}"</code> alphabetically.`;

  } else {

    localeOutput = `The strings are alphabetically equal.`;

  }

  document.getElementById("localeResult").innerHTML = localeOutput;

}
```