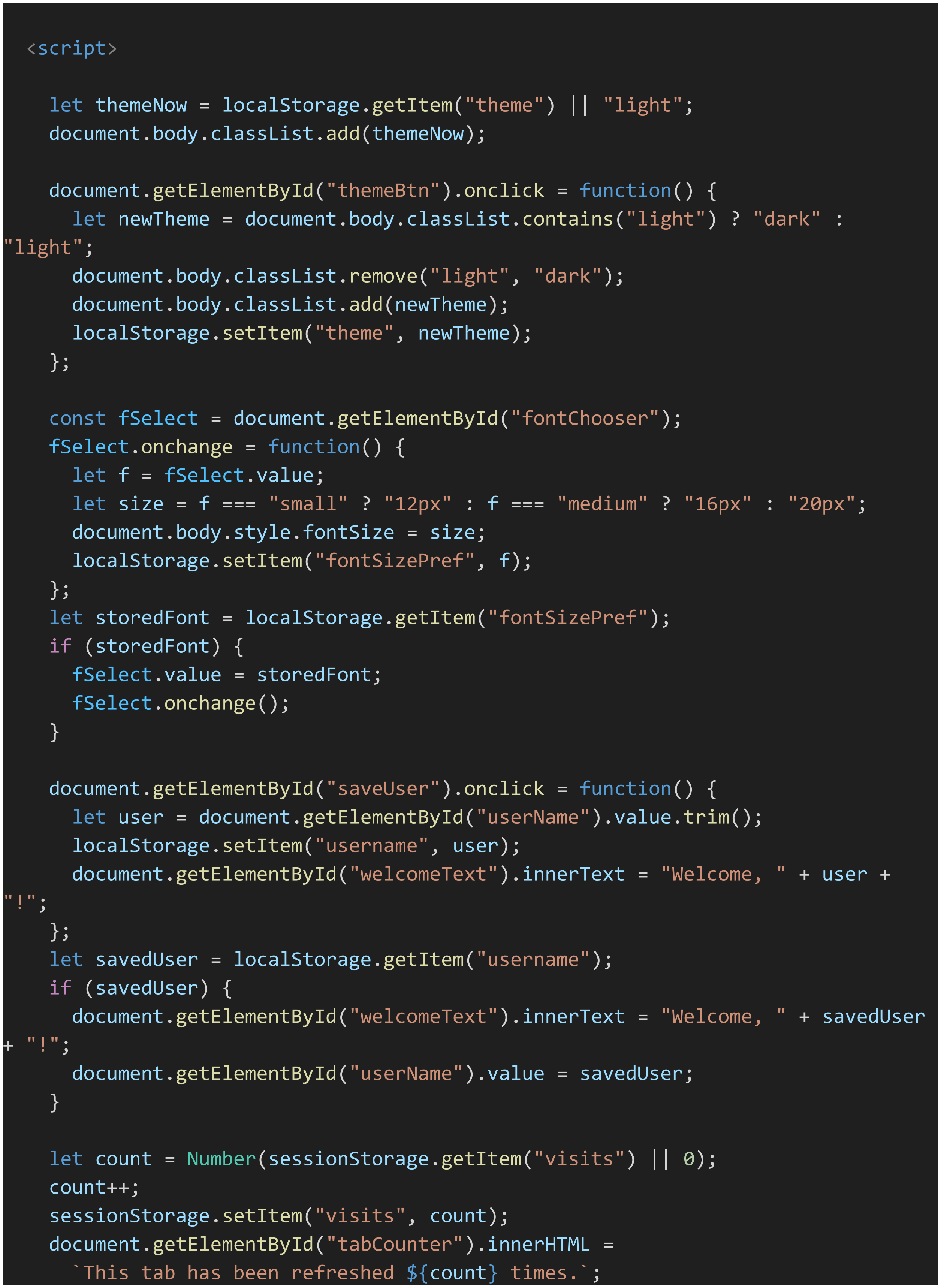
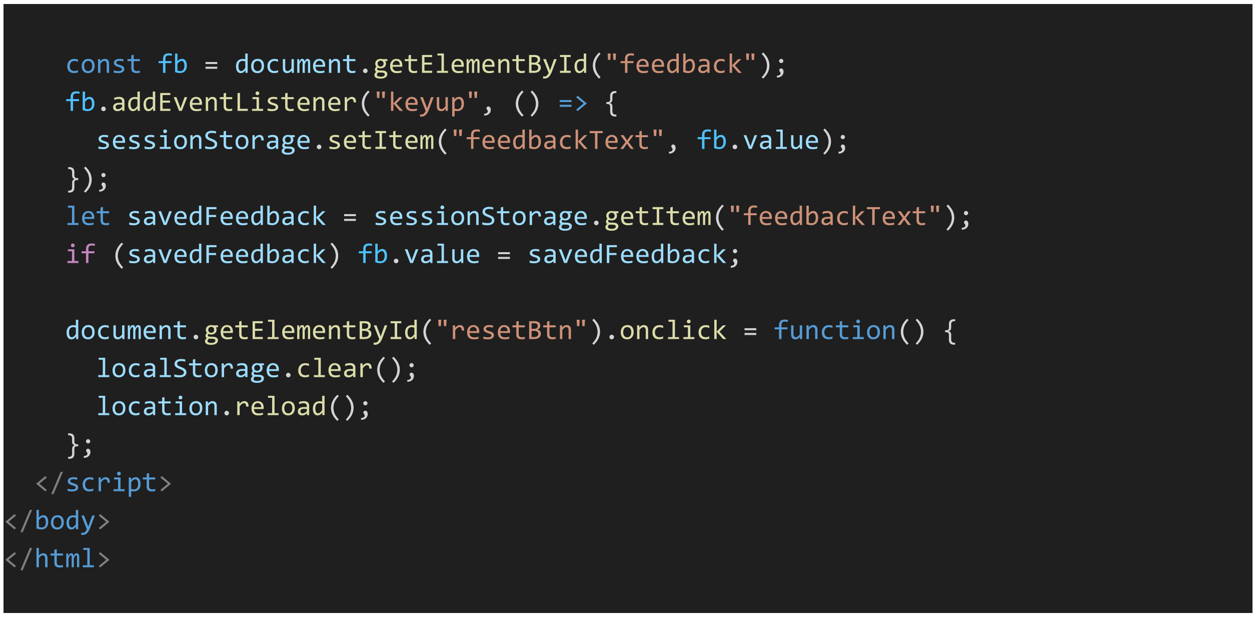
CODE:

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Personal Dashboard</title>
  <style>
    body {
      font-family: Verdana, sans-serif;
      transition: all 0.4s ease;
      text-align: center;
      padding: 20px;
    }
    .dark { background-color: #121212; color: white; }
    .light { background-color: #ffffff; color: black; }
    #controls button, select {
      margin: 6px;
      padding: 8px 14px;
      border: none;
      border-radius: 5px;
      cursor: pointer;
    }
  </style>
</head>
<body>
  <h1 id="welcomeText">Welcome!</h1>

  <div id="controls">
    <input type="text" id="userName" placeholder="Enter your name">
    <button id="saveUser">Save Name</button>
    <button id="themeBtn">Switch Theme</button>
    <select id="fontChooser">
      <option value="small">Small</option>
      <option value="medium">Medium</option>
      <option value="large">Large</option>
    </select>
    <button id="resetBtn">Reset Preferences</button>
  </div>

  <p id="tabCounter"></p>

  <h3>Feedback</h3>
  <textarea id="feedback" rows="5" cols="40" placeholder="Your
feedback..."></textarea>
```

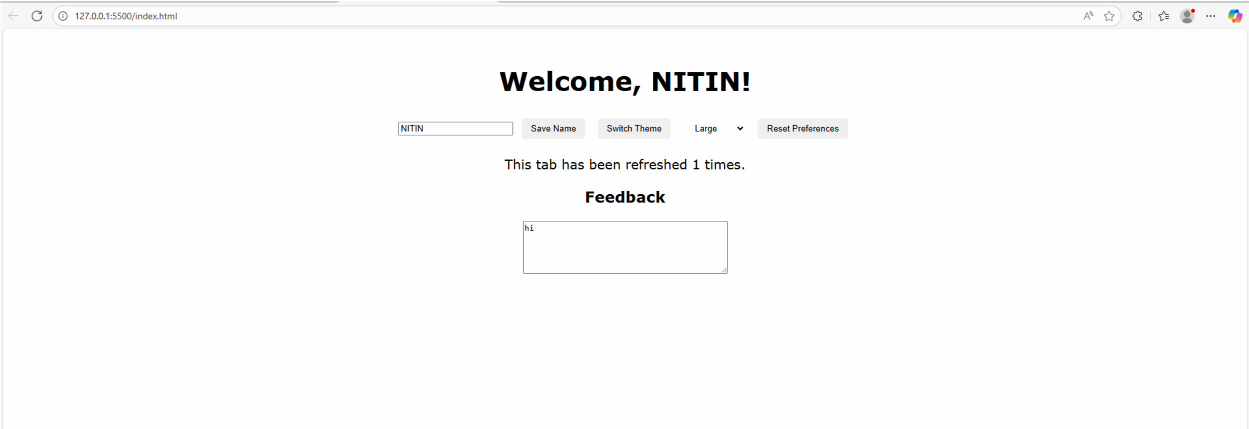


```
  <script>

    let themeNow = localStorage.getItem("theme") || "light";
    document.body.classList.add(themeNow);

    document.getElementById("themeBtn").onclick = function() {
      let newTheme = document.body.classList.contains("light") ? "dark" :
"light";
      document.body.classList.remove("light", "dark");
      document.body.classList.add(newTheme);
      localStorage.setItem("theme", newTheme);
    };

    const fSelect = document.getElementById("fontChooser");
    fSelect.onchange = function() {
      let f = fSelect.value;
      let size = f === "small" ? "12px" : f === "medium" ? "16px" : "20px";
      document.body.style.fontSize = size;
      localStorage.setItem("fontSizePref", f);
    };
    let storedFont = localStorage.getItem("fontSizePref");
    if (storedFont) {
      fSelect.value = storedFont;
      fSelect.onchange();
    }

    document.getElementById("saveUser").onclick = function() {
      let user = document.getElementById("userName").value.trim();
      localStorage.setItem("username", user);
      document.getElementById("welcomeText").innerText = "Welcome, " + user +
"!";
    };
    let savedUser = localStorage.getItem("username");
    if (savedUser) {
      document.getElementById("welcomeText").innerText = "Welcome, " + savedUser
+ "!";
      document.getElementById("userName").value = savedUser;
    }

    let count = Number(sessionStorage.getItem("visits") || 0);
    count++;
    sessionStorage.setItem("visits", count);
    document.getElementById("tabCounter").innerHTML =
      `This tab has been refreshed ${count} times.`;
```



```
    const fb = document.getElementById("feedback");
    fb.addEventListener("keyup", () => {
      sessionStorage.setItem("feedbackText", fb.value);
    });
    let savedFeedback = sessionStorage.getItem("feedbackText");
    if (savedFeedback) fb.value = savedFeedback;

    document.getElementById("resetBtn").onclick = function() {
      localStorage.clear();
      location.reload();
    };
  </script>
</body>
</html>
```

Outputs:



Welcome, NITIN!

NITIN

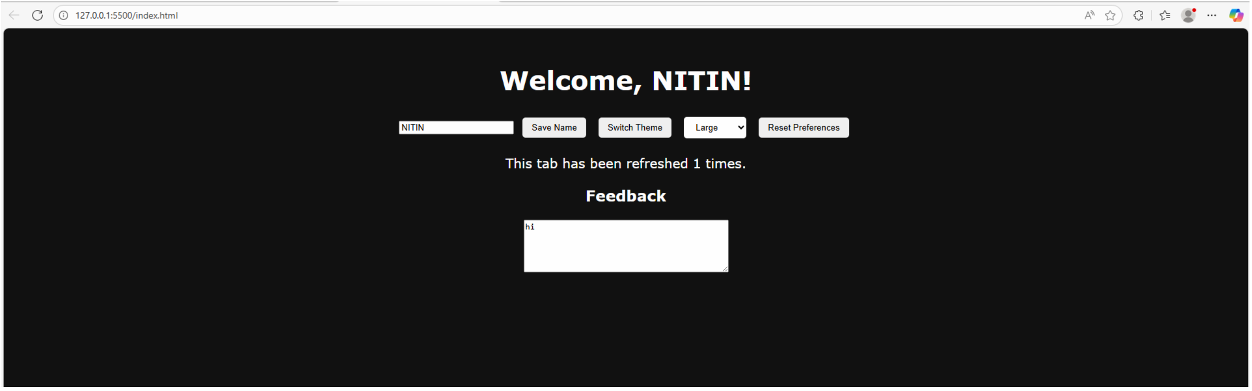
Large

This tab has been refreshed 1 times.

Feedback

hi

Theme Changed:



Welcome, NITIN!

NITIN

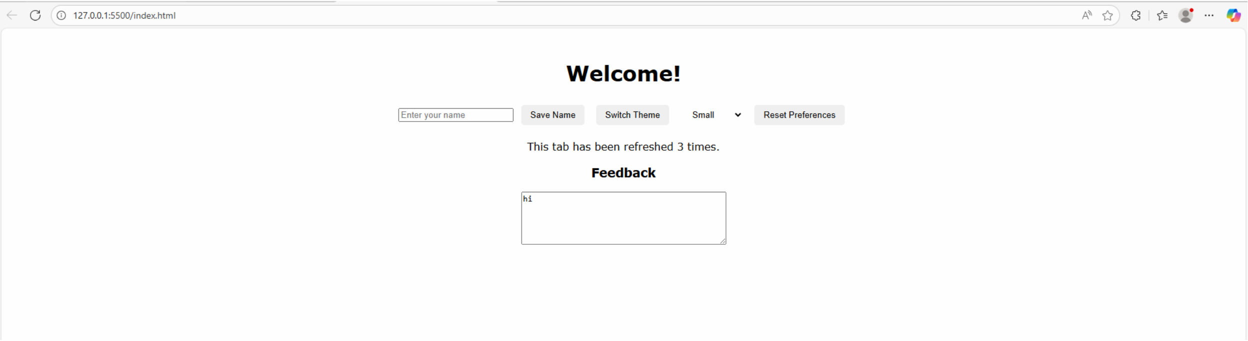
Large

This tab has been refreshed 1 times.

Feedback

hi

RESET:



Welcome!

This tab has been refreshed 3 times.

Feedback

hi

Issues faced:

Task 1: Theme

Problem: Two different keys were used for theme ("ThemeMode" and "theme"), and the default was set to dark instead of light. Also, `.classList = t` was invalid syntax for toggling classes.

Fix: Use one consistent key "theme", default to "light", and use `classList.add/remove` to properly toggle and persist the theme.

Task 2: Font Size

Problem: Wrong key name capitalization ("fontsizePref" vs "fontSizePref") and semantic size names ("small", "medium", "large") were not valid CSS values.

Fix: Used correct key name "fontSizePref" and mapped sizes to px valuesso font size changes properly and stays after reload.

Task 3: Username Greeting

Problem: localStorage.name was used instead of localStorage.setItem("username") and saved name was not shown automatically after reload.

Fix: Used setItem/getItem with key "username" and added logicto display saved username on page load.

Task 4: Session Counter

Problem: count++ was used incorrectly before saving and value was stored as string.

Fix:Converted value to number, then incremented and saved it properlyso counter increases each time page reloads and resets when tab closes.

Task 5: Feedback Restoration

Problem: Used innerHTML instead of value for textarea and no restore logic was added.

Fix: Used textarea.value to save feedback and added code to load saved feedback when page reloads (within same tab).