**Experiment Documentation: PHP Session Management Example**

**Experiment Title: PHP Session Management**

**1. Problem Statement**

**The goal of this experiment is to demonstrate session management in PHP. The session stores user information (username and email), and the user can submit the data via a form or destroy the session data to log out.**

**2. Steps and Explanation**

1. **Session Management in PHP:**
   * **session\_start() is called at the beginning of the script to initiate or resume a session.**
   * **Sessions allow PHP to store user-specific information across multiple page requests.**
2. **Form Handling:**
   * **When the form is submitted via POST, the script captures the username and email and stores them in the session ($\_SESSION).**
   * **htmlspecialchars() is used to sanitize the input, preventing Cross-Site Scripting (XSS) attacks.**
   * **If the action destroy is triggered via a GET request, the session is destroyed using session\_unset() and session\_destroy().**
3. **HTML Form and UI:**
   * **The <form> allows the user to input their username and email.**
   * **The form is submitted via the POST method.**
   * **If session data (username and email) exists, the form is hidden, and the stored session data is displayed.**
   * **A button allows the user to destroy the session, effectively "logging out" by erasing the session data.**
4. **CSS Styling:**
   * **The .form-container provides a centered form with padding and a light background.**
   * **Input fields (text and email) are styled for consistent width and padding.**
   * **The submit button has a hover effect, and a special .button class is used for the "Destroy Session" link, styled like a button.**

**3. Expected Output**

1. **Before Submission:**
   * **A form will appear with two input fields:**
     + **Username (text input).**
     + **Email (email input).**
   * **A Submit button will submit the form data.**
2. **After Submission:**
   * **The form will disappear, and the session-stored username and email will be displayed:**

**less**

**Copy code**

**Session Data**

**Username: [Submitted Username]**

**Email: [Submitted Email]**

* + **A "Destroy Session" button will appear, allowing the user to clear the session data.**

1. **After Destroying Session:**
   * **The session data will be removed, and the user will be presented with the input form again.**
2. **Sample Input and Output:**

* **Sample Input:**
  + **Username: JohnDoe**
  + **Email: john.doe@example.com**
* **Sample Output:**

**makefile**

**Copy code**

**Session Data**

**Username: JohnDoe**

**Email: john.doe@example.com**

**[Destroy Session Button]**

1. **After Clicking "Destroy Session":**
   * **The session will be destroyed, and the user will see the form again:**

**csharp**

**Copy code**

**[Form to Input Username and Email]**

**This example demonstrates basic session management i**

