**Quick User Guide: Silent Auction Web Page**

This guide will walk you through using the **Silent Auction Web Page** to view and interact with auction items.

**Accessing the Application**

1. **Start the Application:**
   * Open the project in **Eclipse** or your preferred IDE.
   * Locate the ShoppingApplication.java file.
   * Right-click the file and select **Run as → Java Application**.
   * Open your web browser and go to [http://localhost:8080](http://localhost:8080/), this will display the home page.
2. **Access Product Information:**
   * Navigate to the following URLs for specific views:
     + <http://localhost:8080/products>: Displays all products in JSON format.
     + <http://localhost:8080/products/html>: Displays products in an easy-to-read HTML table.
     + [http://localhost:8080/products/{id}](http://localhost:8080/products/%7Bid%7D): Displays detailed information for a specific product (replace {id} with the product's ID).

**Navigating the Auction Page**

1. **Homepage Overview:**
   * Visit [http://localhost:8080](http://localhost:8080/) to view the **Silent Auction Items** section.
2. **Auction Items:**
   * The **Silent Auctions** section displays auction items dynamically loaded from the backend.
   * Each item includes:
     + **ID:** A unique number to identify each item.
     + **Name**: A short title describing the auction item.
     + **Starting Bid**: The minimum price for the item (e.g., $50.00).
3. **Dynamic Updates:**
   * Auction items load automatically based on the data from the backend API. A loading message is displayed until the items are fully loaded.

**Troubleshooting**

1. **No Items Displayed:**
   * Verify that the backend is running and that the /products API is correctly returning data.
   * Check the console for errors and ensure the database has data.
2. **Page Not Loading:**
   * Ensure that the project is running on the correct port (default: 8080).
   * Restart the application if needed.
3. **JSON Endpoint Issues:**
   * Test the API by visiting <http://localhost:8080/products> to confirm it’s returning a valid JSON response.

**Additional Notes**

* The page is currently focused on displaying auction items using **CRUD operations,** and **Angular JS.**
* Bidding and login functionality are not implemented in this version but can be added in future iterations.

Thank you for using the Silent Auction Web Page!