**Online Food Ordering System**

**Scenario:**  
You are required to build a **web-based food ordering system** that allows users to browse a menu, place an order, and track order status until it is delivered.

**Requirements:**

1. **Menu & Order Form (HTML + CSS)**
   * Create a menu with at least 3 food items (e.g., Pizza, Burger, Fries) each having a **name, price, and “Add to Cart” button**.
   * Show the cart section with:
     + List of ordered items
     + Total price
     + “Place Order” button
   * Style the page with CSS for a clean layout.
2. **Data Handling (Functions, Objects, Arrays)**
   * Represent menu items as an **array of objects**, each containing:
     + Item ID
     + Name
     + Price
   * Maintain the cart as an array of objects (item + quantity).
   * Use **functions** to calculate the total price dynamically.
3. **Date Object**
   * Show the **order placed time** when the user clicks “Place Order”.
4. **DOM Manipulation & Event Handling**
   * Add items to the cart using button click events.
   * Display cart contents dynamically inside a <ul> or <table>.
   * Update total price in real time.
   * On “Place Order”, clear the cart and show order confirmation in DOM.
5. **BOM**
   * Use an alert (window.alert) to confirm when order is placed.
   * Display a confirmation prompt before refreshing/closing page (beforeunload).
6. **Asynchronous Operations**
   * Simulate order processing using setInterval.
   * Order status should change every 3 seconds in sequence:
     + Order Placed → Preparing → Out for Delivery → Delivered
7. **Callbacks**
   * Implement a callback to update the cart total whenever an item is added.
8. **Promises**
   * When the user clicks “Place Order”:
     + Create a promise that **resolves** if the cart has items.
     + **Rejects** if the cart is empty (show error message in DOM).
9. **Async/Await**
   * Simulate waiting for server confirmation using async/await.
   * Example: After placing the order, wait 2 seconds before confirming “Order Accepted”.

**Expected Outcome:**

* User adds food items to the cart → cart updates dynamically with total.
* On “Place Order”:
  + If cart is empty → show error.
  + If items are present → show confirmation with order time.
  + Status updates asynchronously until “Delivered”.
* User can cancel the order anytime.