

Web Development Exercise: Order History with Status Badges

Objective

Build a small **HTML + CSS + JavaScript** web page that loads **order history from a JSON file** and dynamically renders it using **DOM manipulation**.

This exercise tests:

- JSON parsing
 - DOM creation & updates
 - Conditional rendering
 - Event handling (event delegation)
 - Basic state management
-

Given

You are provided with a file named `orders.json` that contains an array of order objects.

Task

Create an **Order History** page with:

1. A list (or table/cards) of orders
 2. A colored **status badge** for each order
 3. A **details panel** that shows full order details when an order is clicked
-

UI Requirements

1. Order List

Each order entry must display:

- Order ID
- Order Date
- Customer Name
- Total Amount (calculated)
- Status Badge

Each order row/card must:

- Be clickable
 - Store the order id using a `data-order-id` attribute
-

2. Order Details Panel

When an order is clicked, show:

- Order ID
- Date
- Customer name and city
- Status badge
- List of items:
 - Product name
 - Quantity
 - Price
 - Line total (**qty × price**)
- Grand total (computed from items)

If the order has **no items**, display:

No items in this order

Status Badge Rules

Map order status to CSS classes:

Status	CSS Class
DELIVERED	badge badge--delivered
SHIPPED	badge badge--shipped
PENDING	badge badge--pending
CANCELLED	badge badge--cancelled
Unknown	badge badge--unknown

If the status value is not recognized, show an **UNKNOWN** badge.

Data Rules

- **Do not trust any **total** field in JSON**
- Always calculate totals from line items
- Skip invalid items (missing or non-numeric qty/price)
- Show a small warning note if invalid items are skipped

Interaction Requirements

- Use **event delegation** (single click handler on the order list container)
- Highlight the selected order visually using a **.selected** CSS class
- Clicking a different order updates the details panel

Constraints

- ✗ No `alert()` usage
 - ✗ No page reloads
 - ☑ Must use `fetch()` to load JSON
 - ☑ Must handle:
 - Empty JSON array
 - Missing optional fields
 - Invalid item data
 - ★ Bonus: No use of `innerHTML`; use `createElement()` and `textContent`
-

Sample `orders.json`

```
[
  {
    "orderId": "ORD-1007",
    "date": "2026-02-01",
    "customer": { "name": "Ananya Rao", "city": "Hyderabad" },
    "status": "DELIVERED",
    "items": [
      { "name": "Cotton Kurta", "qty": 2, "price": 899 },
      { "name": "Dupatta", "qty": 1, "price": 499 }
    ]
  },
  {
    "orderId": "ORD-1008",
    "date": "2026-02-03",
    "customer": { "name": "Rahul Mehta", "city": "Mumbai" },
    "status": "CANCELLED",
    "items": [
      { "name": "Sneakers", "qty": 1, "price": 2499 }
    ]
  }
]
```