**Use Case: Responsive Design for a Table with Too Many Columns**

A UI that includes a data table. But on mobile devices, too many columns make it unreadable, causing horizontal scrolling or squished data.

**Fixing it with Responsive Design Patterns**

**1. Horizontal Scrollable Table (Baseline)**

* Wrap the table in a container with overflow-x: auto.
* **Pros**: Preserves all data.
* **Cons**: Requires horizontal scrolling on small screens.

<div class="table-container">

<table>

<thead>...</thead>

<tbody>...</tbody>

</table>

</div>

.table-container {

overflow-x: auto;

max-width: 100%;

}

**2. Card/Stacked Layout (Mobile-First Redesign)**

* On smaller screens, transform each row into a **stacked card** using CSS or JavaScript.
* Each "card" shows a record, with key-value pairs instead of table rows.

<div class="data-card">

<p><strong>Name:</strong> Jane Doe</p>

<p><strong>Email:</strong> jane@example.com</p>

<p><strong>Status:</strong> Active</p>

...

</div>

* **Pros**: Mobile-friendly, readable.
* **Cons**: Takes more vertical space.

**3. Column Priority & Hiding**

* Use display: none via media queries to hide non-essential columns on smaller screens.
* Provide a toggle ("Show more details") to expand the rest.

@media (max-width: 600px) {

.col-id, .col-joined, .col-status {

display: none;

}

}

**Reusable Component Design Tips**

Turn this into a reusable table pattern:

* Allow **column config**: what to hide, show, or prioritize.
* Allow **mobile card fallback**: pass in templates or slots for stacked views.
* Keep **ARIA roles** for screen reader accessibility.

**Responsive Data Table Redesign**

**Objective:**

Redesign a wide data table using two of the following responsive patterns:

1. Stacked card layout for mobile
2. Column priority hiding with "More details" toggle