### 1.

**Fake Dataset:**

let products = [  
 { id: 201, name: "Smartphone", price: 699, category: "Electronics", stock: 50, rating: 4.6 },  
 { id: 202, name: "Blender", price: 49, category: "Appliances", stock: 20, rating: 4.3 },  
 { id: 203, name: "Yoga Mat", price: 25, category: "Fitness", stock: 100, rating: 4.9 },  
 { id: 204, name: "Office Desk", price: 299, category: "Furniture", stock: 10, rating: 4.2 },  
 { id: 205, name: "Bluetooth Speaker", price: 79, category: "Electronics", stock: 60, rating: 4.7 }  
];

**Question:**  
Display the products dynamically as cards on a webpage, with each card showing the product’s name, price, category, rating, and a “Buy Now” button. Ensure the layout is responsive and visually appealing.

### 2.

**Fake Dataset:**

let products = [  
 { id: 701, name: "Action Camera", price: 299, category: "Electronics", stock: 20, rating: 4.8 },  
 { id: 702, name: "Toaster", price: 29, category: "Appliances", stock: 40, rating: 4.2 },  
 { id: 703, name: "Tennis Racket", price: 129, category: "Sports", stock: 60, rating: 4.5 },  
 { id: 704, name: "Dining Table", price: 799, category: "Furniture", stock: 5, rating: 4.6 },  
 { id: 705, name: "VR Headset", price: 599, category: "Electronics", stock: 15, rating: 4.9 }  
];

**Question:**  
Display a grid of product cards with details including name, price, and rating. Ensure the layout is responsive, showing 2 cards per row on tablets and 3 cards per row on desktops.

### 3.

**Fake Dataset:**

let products = [  
 { id: 401, name: "Wireless Mouse", price: 29, category: "Accessories", stock: 150, rating: 4.5 },  
 { id: 402, name: "Portable Fan", price: 19, category: "Appliances", stock: 40, rating: 4.3 },  
 { id: 403, name: "Resistance Bands", price: 15, category: "Fitness", stock: 120, rating: 4.9 },  
 { id: 404, name: "Floor Lamp", price: 99, category: "Furniture", stock: 8, rating: 4.2 },  
 { id: 405, name: "Tablet", price: 199, category: "Electronics", stock: 25, rating: 4.7 }  
];

**Question:**  
Create a webpage displaying product cards with details like name, price, and stock. Use CSS to create a responsive layout with cards stacking vertically on small screens.

### 4.

**Fake Dataset:**

let products = [  
 { id: 901, name: "Monitor", price: 199, category: "Electronics", stock: 50, rating: 4.6 },  
 { id: 902, name: "Microwave Oven", price: 149, category: "Appliances", stock: 20, rating: 4.4 },  
 { id: 903, name: "Fitness Tracker", price: 99, category: "Fitness", stock: 80, rating: 4.7 },  
 { id: 904, name: "Bookshelf", price: 129, category: "Furniture", stock: 10, rating: 4.3 },  
 { id: 905, name: "Gaming Keyboard", price: 99, category: "Electronics", stock: 35, rating: 4.8 }  
];

**Question:**  
Display product cards with a clean design using CSS. Include details like name, price, and category. Ensure the grid adjusts to 1 card per row on mobile, 2 on tablets, and 4 on larger screens.

### 5.

**Fake Dataset:**

let products = [  
 { id: 1001, name: "Noise Cancelling Headphones", price: 349, category: "Electronics", stock: 25, rating: 4.9 },  
 { id: 1002, name: "Air Fryer", price: 129, category: "Appliances", stock: 15, rating: 4.6 },  
 { id: 1003, name: "Cycling Helmet", price: 79, category: "Fitness", stock: 60, rating: 4.8 },  
 { id: 1004, name: "TV Stand", price: 199, category: "Furniture", stock: 12, rating: 4.4 },  
 { id: 1005, name: "Smartphone Case", price: 19, category: "Accessories", stock: 100, rating: 4.5 }  
];

**Question:**  
Build a webpage that displays product cards with a simple and consistent style. Focus on responsiveness to show 1 card per row on small screens, 3 cards per row on medium screens, and 5 cards per row on large screens.