

EXPERIMENT 4.3

AIM:

E-commerce Catalog with Nested Document Structure in MongoDB

CODE:

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <title>E-commerce Catalog (MongoDB Nested Structure)</title>

  <style>

    body { font-family: Arial, sans-serif; margin: 20px; }

    h1, h2 { color: #2c3e50; }

    .category { margin-bottom: 20px; }

    table { border-collapse: collapse; width: 100%; margin-top: 10px; }

    table, th, td { border: 1px solid #444; }

    th, td { padding: 8px; text-align: left; }

    th { background: #f4f4f4; }

  </style>

</head>

<body>

  <h1>E-commerce Catalog</h1>

  <div id="catalog"></div>

  <script>

    // Simulating MongoDB nested document structure

    const catalog = [

      {

        category: "Electronics",
```

```
products: [  
  {  
    name: "Smartphone",  
    brand: "Samsung",  
    price: 25000,  
    specs: { storage: "128GB", camera: "64MP", battery: "5000mAh" }  
  },  
  {  
    name: "Laptop",  
    brand: "Dell",  
    price: 60000,  
    specs: { ram: "16GB", storage: "512GB SSD", processor: "i7" }  
  }  
]  
,  
{  
  category: "Fashion",  
  products: [  
    {  
      name: "T-Shirt",  
      brand: "Nike",  
      price: 1200,  
      specs: { size: "M", color: "Black", material: "Cotton" }  
    },  
    {  
      name: "Shoes",  
      brand: "Adidas",  
      price: 3500,  
      specs: { size: "9", color: "White", type: "Running" }  
    }  
  ]  
}
```

```

    }
  ];

  // Render catalog
  const container = document.getElementById("catalog");

  catalog.forEach(cat => {
    const section = document.createElement("div");
    section.className = "category";

    section.innerHTML = `<h2>${cat.category}</h2>
    <table>
    <thead>
    <tr>
    <th>Product</th>
    <th>Brand</th>
    <th>Price (₹)</th>
    <th>Details</th>
    </tr>
    </thead>
    <tbody>
    ${cat.products.map(p => `
    <tr>
    <td>${p.name}</td>
    <td>${p.brand}</td>
    <td>${p.price}</td>
    <td>
    ${Object.entries(p.specs).map(([k,v]) => `${k}: ${v}`).join(", ")}
    </td>
    </tr>
    `).join("")}
  `);

```

```

        </tbody>
    </table>`;

    container.appendChild(section);
});
</script>

</body>
</html>
```

OUTPUT:

E-commerce Catalog

Electronics

Product	Brand	Price (₹)	Details
Smartphone	Samsung	25000	storage: 128GB, camera: 64MP, battery: 5000mAh
Laptop	Dell	60000	ram: 16GB, storage: 512GB SSD, processor: i7

Fashion

Product	Brand	Price (₹)	Details
T-Shirt	Nike	1200	size: M, color: Black, material: Cotton
Shoes	Adidas	3500	size: 9, color: White, type: Running