



## Product Schema for Clothing

schemas/singleProduct.js

```

export default {
  name: 'singleProduct',
  title: 'Product',
  type: 'document',
  fields: [
    { name: 'name', type: 'string', title: 'Name' },
    { name: 'description', type: 'text', title: 'Description' },
    { name: 'price', type: 'number', title: 'Price' },
    { name: 'image', type: 'image', title: 'Image' },
    { name: 'size', type: 'string', title: 'Size' },
    { name: 'color', type: 'string', title: 'Color' },
    { name: 'stock', type: 'number', title: 'Stock' },
  ],
};

```

Fields:

- 1 Name: Product name.
- 2 Description: Product description.
- 3 Price: Price of the product.
- 4 Image: Image of the product.
- 5 Size: Size of the clothing.
- 6 Color: Color options available.
- 7 Stock: Quantity in stock.

### Product API Endpoints:

- **GET** /api/products : Fetch all products.
- **GET** /api/products/{id} : Fetch a specific product by ID.
- **POST** /api/products : Create a new product (e.g., clothing item).
- **PUT** /api/products/{id} : Update an existing product by ID.
- **DELETE** /api/products/{id} : Delete a product by ID.

## Customer Schema schemas/customer.js

```
export default {
  name: 'customer',
  title: 'Customer',
  type: 'document',
  fields: [
    { name: 'name', type: 'string', title: 'Name' },
    { name: 'email', type: 'string', title: 'Email' },
    { name: 'phone', type: 'string', title: 'Phone' },
    { name: 'address', type: 'string', title: 'Address' },
    { name: 'orderHistory', type: 'array', title: 'Order History', of: [{ type: 'reference', to: [{ type:
'order' }] }] },
  ],
};
```

### Fields:

- 1 Name: Customer's name.
- 2 Email: Customer's email address.
- 3 Phone: Customer's phone number.
- 4 Address: Shipping address of the customer.
- 5 Order History: Reference to the customer's past orders.

### 1. Customer API Endpoints:

- **GET** /api/customers : Fetch all customers.
- **GET** /api/customers/{id} : Fetch a specific customer by ID.
- **POST** /api/customers : Create a new customer.
- **PUT** /api/customers/{id} : Update an existing customer by ID.
- **DELETE** /api/customers/{id} : Delete a customer by ID.

## Order Schema

schemas/order.js

```
export default {
  name: 'order',
  title: 'Order',
  type: 'document',
  fields: [
    { name: 'orderNumber', type: 'string', title: 'Order Number' },
    { name: 'customer', type: 'reference', title: 'Customer', to: [{ type: 'customer' }] },
    { name: 'items', type: 'array', title: 'Items', of: [{ type: 'reference', to: [{ type: 'singleProduct'
}] }] },
    { name: 'totalAmount', type: 'number', title: 'Total Amount' },
    { name: 'status', type: 'string', title: 'Status', options: { list: ['Pending', 'Shipped', 'Delivered',
'Cancelled'] } },
    { name: 'shippingAddress', type: 'string', title: 'Shipping Address' },
    { name: 'paymentMethod', type: 'string', title: 'Payment Method' },
  ],
};
```

```
    { name: 'orderDate', type: 'datetime', title: 'Order Date' },  
  ],  
};
```

### **Fields:**

- Order Number: A unique identifier for the order.
- Customer: Reference to the customer who placed the order.
- Items: References to the products in the order.
- Total Amount: Total cost of the order.
- Status: Current status of the order (e.g., Pending, Shipped, Delivered, Cancelled).
- Shipping Address: Address where the order will be shipped.
- Payment Method: Method used for payment (e.g., Credit Card, PayPal).
- Order Date: Date when the order was placed.

### **. Order API Endpoints:**

- **GET** /api/orders : Fetch all orders.
  - **GET** /api/orders/{id} : Fetch a specific order by ID.
  - **POST** /api/orders : Create a new order.
  - **PUT** /api/orders/{id} : Update an existing order by ID (e.g., change order status).
  - **DELETE** /api/orders/{id} : Delete an order by ID.
-