# *Hackathon3 Day:2 Task.*

**1.Product schema:**

```
export const product = {
  name: 'product',
  title: 'Product',
  type: 'document',
  fields: [
    { name: 'name', title: 'Name', type: 'string' },
    { name: 'price', title: 'Price', type: 'number' },
    { name: 'description', title: 'Description', type: 'text' },
    { name: 'image', title: 'Image', type: 'image' },
    { name: 'category', title: 'Category', type: 'string' },
    {name: 'stock', title: 'Stock', type: 'number' },
    {name: 'tags', title: 'Tags', type: 'array', of: [{ type: ‘string’}] },
  ],
};
```

**2.Order schema:**

```
export const order = {
  name: 'order',
  title: 'Order',
  type: 'document',
  fields: [
    {
      name: 'cartItems',
      title: 'Cart Items',
      type: 'array',
      of: [
        {
          type: 'object',
          fields: [
            { name: 'product', title: 'Product', type: 'reference', to: [{ type: 'product' }] },
            { name: 'quantity', title: 'Quantity', type: 'number' },
          ],
        },
      ],
    },
    { name: 'user', title: 'User Details', type: 'reference', to: [{ type: 'customer' }] },
    { name: 'status', title: 'Status', type: 'string' },
    { name: 'shipping', title: 'Shipping', type: 'object', fields: [{ name: 'address', title: 'Address', type: 'string' }] },
    { name: 'billing', title: 'Billing', type: 'object', fields: [{ name: 'address', title: 'Address', type: 'string' }] },
    { name: 'totalAmount', title: 'Total Amount', type: 'number' },
    { name: 'paymentMethod', title: 'Payment Method', type: 'string' },
    { name: 'createdAt', title: 'Created At', type: 'datetime' },
  ],
};
```

### 3.Customer schema:

```
export const customer = {
  name: 'customer',
  title: 'Customer',
  type: 'document',
  fields: [
    { name: 'name', title: 'Name', type: 'string' },
    { name: 'email', title: 'Email', type: 'string' },
    { name: 'password', title: 'Password', type: 'string' }, // Use hashing at the application level
    { name: 'address', title: 'Address', type: 'text' },
    { name: 'phone', title: 'Phone', type: 'string' },
  ],
};
```

### 4.Shipment schema:

```
export const shipment = {
  name: 'shipment',
  title: 'Shipment',
  type: 'document',
  fields: [
    { name: 'order', title: 'Order', type: 'reference', to: [{ type: 'order' }] },
    { name: 'trackingNumber', title: 'Tracking Number', type: 'string' },
    { name: 'shipmentStatus', title: 'Shipment Status', type: 'string' },
    { name: 'estimatedDeliveryDate', title: 'Estimated Delivery Date', type: 'datetime' },
  ],
};
```

### 5.Payment schema:

```
export const payment = {
  name: 'payment',
  title: 'Payment',
  type: 'document',
  fields: [
    { name: 'user', title: 'User/Customer', type: 'reference', to: [{ type: 'customer' }] },
    { name: 'order', title: 'Order', type: 'reference', to: [{ type: 'order' }] },
    { name: 'amount', title: 'Amount', type: 'number' },
    { name: 'paymentMethod', title: 'Payment Method', type: 'string' },
    { name: 'paymentStatus', title: 'Payment Status', type: 'string' },
  ],
};
```

Groq Language:

### Fetch All Products

```
*[_type == "product"] {
  _id,
  name,
  price,
  description,
  "imageUrl": image.asset->url,
  category,
  stock,
  tags
}
```

### Fetch Orders with Related Data

```
*[_type == "order"] {
  _id,
  cartItems[] {
    product-> {
      name,
      price
    },
    quantity
  },
  user-> {
    name,
    email
  },
  status,
  shipping,
  billing,
  totalAmount,
  paymentMethod,
  createdAt
}
```

### Fetch Customers

```
*[_type == "customer"] {
  _id,
  name,
  email,
  address,
  phone
}
```

### Fetch Shipment Data

```
*[_type == "shipment"] {
  _id,
  order-> {
    _id,
    totalAmount
  },
  trackingNumber,
  shipmentStatus,
  estimatedDeliveryDate
}
```

### Fetch Payments

```
*[_type == "payment"] {
  _id,
  user-> {
    name,
    email
  },
  order-> {
    _id,
    totalAmount
  },
  amount,
  paymentMethod,
  paymentStatus
}
```