

```
    },  
    description: 'Add tags such as "new arrival", "bestseller", or "limited edition".',  
  },  
  {
```

page 4

```
    // Array field for available product sizes  
    name: 'sizes',  
    type: 'array',  
    title: 'Sizes',  
    of: [{ type: 'string' }], // Each size is a string  
    options: {  
      layout: 'tags', // Allows for tag-style input  
    },  
    description: 'Available sizes for the product (e.g., S, M, L, XL, XXL).',  
  },  
  {  
    // Field for the product image  
    name: 'image',  
    type: 'image',  
    title: 'Product Image',  
    description: 'High-quality image of the product.',  
    options: {  
      hotspot: true, // Enables cropping and focal point selection  
    },  
    validation: (Rule) => Rule.required().error('Product image is required.'),  
  },  
  {  
    // Field for the SEO-friendly title  
    name: 'seoTitle',  
    type: 'string',  
    title: 'SEO Title',  
    description: 'Title for SEO optimization (max 60 characters).',  
    validation: (Rule) => Rule.max(60).error('SEO title cannot exceed 60 characters.'),  
  },  
  {  
    // Field for the SEO-friendly description  
    name: 'seoDescription',  
    type: 'text',  
    title: 'SEO Description',  
    description: 'Meta description for SEO optimization (max 160 characters).',  
    validation: (Rule) => Rule.max(160).error('SEO description cannot exceed 160 characters.'),  
  },  
],  
};
```

```
1 export default {
2   // Define the document type and its name
3   name: 'product',
4   type: 'document',
5   title: 'Product', // The title displayed in the CMS for this document
6   fields: [
7     {
8       // Field for the product's name
9       name: 'name',
10      type: 'string', // Basic text field
11      title: 'Product Name', // Label for the field in the CMS
12      validation: (Rule) =>
13        Rule.required() // Makes this field mandatory
14          .max(100) // Restricts the maximum length to 100 characters
15          .error('Product name is required and cannot exceed 100 characters.'),
16    },
17    {
18      // Slug field for generating a URL-friendly identifier
19      name: 'slug',
20      type: 'slug',
21      title: 'Slug',
22      description: 'URL-friendly identifier for the product.', // Helper text in the CMS
23      options: {
24        source: 'name', // Automatically generates the slug based on the product name
25        maxLength: 200, // Limits the slug length
26      },
27      validation: (Rule) =>
28        Rule.required().error('Slug is required for product identification.'),
29    },
30    {
31      // Field for a detailed description of the product
32      name: 'description',
33      type: 'text', // Multi-line text field
34      title: 'Description',
35      description: 'Detailed description of the product.',
36      validation: (Rule) =>
37        Rule.required() // Makes this field mandatory
38          .min(20) // Requires at least 20 characters
39          .max(500) // Limits to a maximum of 500 characters
40          .error('Description must be between 20 and 500 characters.'),
41    },
42  ],
43 }
```



```
40     .error('Description must be between 20 and 500 characters.'),
41 },
42 {
43     // Field for the product's price
44     name: 'price',
45     type: 'number', // Numeric field
46     title: 'Product Price',
47     validation: (Rule) =>
48         Rule.required() // Makes this field mandatory
49         .min(0) // Ensures the price is non-negative
50         .error('Product price must be a positive value.'),
51 },
52 {
53     // Field for the discount percentage
54     name: 'discountPercentage',
55     type: 'number',
56     title: 'Discount Percentage',
57     description: 'Percentage discount on the product.',
58     validation: (Rule) =>
59         Rule.min(0) // Ensures the discount is not negative
60         .max(100) // Ensures the discount does not exceed 100%
61         .error('Discount percentage must be between 0 and 100.'),
62 },
63 {
64     // Field for the price before the discount
65     name: 'priceWithoutDiscount',
66     type: 'number',
67     title: 'Price Without Discount',
68     description: 'Original price of the product before discount.',
69     readOnly: true, // Makes this field non-editable in the CMS
70     // Optional: Auto-calculate this field based on price and discount
71     initialValue: (doc) => doc.price / (1 - doc.discountPercentage / 100),
72 },
73 {
74     // Field for the product's rating
75     name: 'rating',
76     type: 'number',
77     title: 'Rating',
78     description: 'Average rating of the product.',
79     validation: (Rule) =>
80         Rule.min(0) // Ensures the rating is not negative
81         .max(5) // Limits the rating to a maximum of 5
```

```
.max(5) // Limits the rating to a maximum of 5
.precision(1) // Allows one decimal place
.error('Rating must be between 0 and 5.'),
```

page 3

```
},
{
  // Field for the number of ratings the product has received
  name: 'ratingCount',
  type: 'number',
  title: 'Rating Count',
  description: 'Total number of ratings received by the product.',
  validation: (Rule) => Rule.min(0).error('Rating count must be a non-negative number.'),
},
{
  // Array field for tags associated with the product
  name: 'tags',
  type: 'array',
  title: 'Tags',
  of: [{ type: 'string' }], // Each tag is a string
  options: {
    layout: 'tags', // Allows for tag-style input
  },
  description: 'Add tags such as "new arrival", "bestseller", or "limited edition".',
},
{
  // Array field for available product sizes
  name: 'sizes',
  type: 'array',
  title: 'Sizes',
  of: [{ type: 'string' }], // Each size is a string
  options: {
    layout: 'tags', // Allows for tag-style input
  },
  description: 'Available sizes for the product (e.g., S, M, L, XL, XXL).',
},
{
  // Field for the product image
  name: 'image',
  type: 'image',
  title: 'Product Image',
  description: 'High-quality image of the product.',
  options: {
    hotspot: true, // Enables cropping and focal point selection
  },
},
```