

- A book can have a promotional price applied to it with an optional row in the Price- Offer, which is an example of a one-to-one relationship. (Technically, the relationship is one-to-zero-or-one, but EF Core handles it the same way.)
- You want to allow customers to review a book; they can give a book a star rating and optionally leave a comment. Because a book may have no reviews or many (unlimited) reviews, you need to create a table to hold that data. In this example, you'll call the table Review. The Books table has a one-to-many relationship to the Review table.
- Books can be written by one or more authors, and an author may write one or more books. Therefore, you need a table called Books to hold the books data and another table called Authors to hold the authors. The link between the Books and Authors tables is called a *many-to-many relationship*, which in this case needs a linking table to achieve this relationship. In this case, you create your own linking table with an Order value in it because the names of the authors in a book must be displayed in a specific order.
- Books can be tagged with different categories—such as Microsoft .NET, Linux, Web, and so on—to help the customer to find a book on the topic they are interested in. A category might be applied to multiple books, and a book might have one or more categories, so a many-to-many linking table is needed. But unlike in the previous BookAuthor linking table, the tags don't have to be ordered, which makes the linking table simpler.

