**Associations**

In the object persistence world, understanding association and relationships is essential. If we cannot handle complexity, we cannot solve real-life problems. In this chapter, we’ll concentrate on the fundamentals of associations and their mappings. You’ll also learn about the directionality and multiplicity of these associations.

In life, simplicity is not always the case! Objects cannot just linger on their own; they need to be able to associate with other objects representing real-life problems and solutions. Until now, we have seen how to persist and access simple Java objects such as a single Movie or Trade, If you dig in deep, a movie rarely exists without content – story, script, actors, and technicians, to name a few examples. The same is true about a trade: it’s always associated with an underlying counterparty, quantity, reference data, and so on. These associations are crucial and need to be accounted for when we are persisting and retrieving data from a relational database. This is a humongous task for any ORM tool. Hibernate implements this functionality with elegance. The real power of Hibernate comes from its ability to help us manage these associations and relationships.