2.2.6 Dealing with Richer Predicates

We can think of a triple (x, P, y) as a logical formula $P(x, y)$, where the binary predicate P relates the object x to the object y. In fact, *RDF offers only binary predicates (properties)*. However, in some cases we may need predicates that have more than two arguments. Luckily, such predicates can be simulated by a number of binary predicates. We illustrate this technique for a predicate *broker* with three arguments. The intuitive meaning of *broker(X, Y, Z)* is

> *X* is the broker in a home sale between seller *Y* and purchaser *Z*.

We now introduce a new auxiliary resource *home-sale* and the binary predicates *broker*, *seller*, and *purchaser*. Then we can represent *broker(X, Y, Z)* as follows:

> *broker(home-sale, X)*
>
> *seller(home-sale, Y)*
>
> *purchaser(home-sale, Z)*

While the predicate with three arguments is more succinct to write, the use of binary predicates does simplify the overall data model.

2.3 RDF Syntaxes

We have already seen one syntax for RDF, namely, a graphical syntax. This syntax is, however, neither machine interpretable nor standardized. Here, we introduce a standard machine interpretable syntax called *Turtle*, and briefly discuss some alternatives.

2.3.1 Turtle

Terse RDF Triple Language (Turtle) is a text-based syntax for RDF. The file extension used for Turtle text files is ".ttl". We have already seen how to write a statement in Turtle earlier. Here's an example: