Right now, our web API exposes the database entities to the client. The client receives data that maps directly to your database tables. However, that's not always a good idea. Sometimes you want to change the shape of the data that you send to client. For example, you might want to:

- Remove circular references (see previous section).
- Hide particular properties that clients are not supposed to view.
- Omit some properties in order to reduce payload size.
- Flatten object graphs that contain nested objects, to make them more convenient for clients.
- Avoid "over-posting" vulnerabilities. (See Model Validation for a discussion of over-posting.)
- Decouple your service layer from your database layer.

To accomplish this, you can define a *data transfer object* (DTO). A DTO is an object that defines how the data will be sent over the network.