A data model is more or less what it sounds like. It’s a way to model what database tables look like in a way that makes sense to humans.

Salesforce supports several different types of objects. There are standard objects, custom objects, external objects, platform events, and BigObjects.

**Standard objects** are objects that are included with Salesforce. Common business objects like Account, Contact, Lead, and Opportunity are all standard objects.

**Custom objects** are objects that you create to store information that’s specific to your company or industry. For DreamHouse, D’Angelo wants to build a custom Property object that stores information about the homes his company is selling.

**Get to Know Fields**

Every standard and custom object has fields attached to it. Let’s get familiar with the different types of fields.

| **Field Type** | **What is it?** | **Can I get an example?** |
| --- | --- | --- |
| Identity | A 15-character, case-sensitive field that’s automatically generated for every record. You can find a record’s ID in its URL. | An account ID looks like 0015000000Gv7qJ. |
| System | Read-only fields that provide information about a record from the system, like when the record was created or when it was last changed. | CreatedDate, LastModifiedById, and LastModifiedDate. |
| Name | All records need names so you can distinguish between them. You can use text names or auto-numbered names that automatically increment every time you create a record. | A contact’s name can be Julie Bean. A support case’s name can be CA-1024. |
| Custom | Fields you create on standard or custom objects are called custom fields. | You can create a custom field on the Contact object to store your contacts’ birthdays. |

## The Wide World of Object Relationships

There are two main types of object relationships: lookup and master-detail.

**Lookup Relationships**

In our Account to Contact example above, the relationship between the two objects is a **lookup relationship**. A lookup relationship essentially links two objects together so that you can “look up” one object from the related items on another object.

Lookup relationships can be one-to-one or one-to-many. The Account to Contact relationship is one-to-many because a single account can have many related contacts. For our DreamHouse scenario, you could create a one-to-one relationship between the Property object and a Home Seller object.

**Master-Detail Relationships**

While lookup relationships are fairly casual, **master-detail relationships** are a bit tighter. In this type of relationship, one object is the master and another is the detail. The master object controls certain behaviors of the detail object, like who can view the detail’s data.

**More on Relationships**

Just like in real life, relationships are complicated. Here’s a bit more information to help you differentiate between lookup and master-detail relationships.

Typically, you use lookup relationships when objects are only related in some cases. Sometimes a contact is associated with a specific account, but sometimes it’s just a contact. Objects in lookup relationships usually work as stand-alone objects and have their own tabs in the user interface.

In a master-detail relationship, the detail object doesn’t work as a stand-alone. It’s highly dependent on the master. In fact, if a record on the master object is deleted, all its related detail records are deleted as well. When you’re creating master-detail relationships, you always create the relationship field on the detail object.

Finally, you could run into a third relationship type called a hierarchical relationship. Hierarchical relationships are a special type of lookup relationship. The main difference between the two is that hierarchical relationships are only available on the User object. You can use them for things like creating management chains between users.

Schema Builder is a tool that lets you visualize and edit your data model. It’s useful for designing and understanding complex data models