**Model Relationships in Django**

Samantha Matthews

**Resource**

<https://www.scaler.com/topics/django/relationships-in-django-models/>

**Basic Summary**

This resource goes through what model relationships are, the purpose of model relationships, the types of relationships, data integrity, and code examples.

**What is a Model Relationship?**

Model relationships are a way to link different models so that they can react to changes within each other. For example, linking a student model to a model that stores what major they are in. So that when you remove a major it will change it for every student model. These allow for having changes in one model reflect on other models.

**Why Model Relationships?**

Relationships make it so that you do not have to manually change each model when it contains information that depends on a different model. In the student and major example, if a relationship was not created between the model that contains student info and the model that contains the majors, if you removed one you would have to manually get rid of that in every student. By linking the two models through a relationship you are able to make it so that on delete, it will go through each student and delete that major. Making a change in one table can reflect on the other.

**One-to-One Relationships**

One record of a model is related to exactly one record of a different model. This is used for security purposes. For example, having only one user id relate to only one student. One-to-one can be implemented so that when a student is removed from the system their corresponding user id will also be removed.

**Many-to-Many Relationships**

Each record in the first model is related to several records in the second model and vice versa. For example, a product can have multiple customers and a customer can have multiple models. These types of relationships come into play when you have models where several of their elements are related to each other.

**Many-to-One Relationships**

A record in one model can only be related to one in the other, but the other model relates to several records in the one model. For example, a song can have one album, but an album has multiple songs. So, the “one” is the album model and the “many” is the song model. These work in situations where you have something that belongs under one category, but that category can contain multiple things.

**Data Integrity**

Data integrity is the accuracy of a database. Defining the behavior of a model maintains database integrity. When you delete something that another model relies on, then you want to make sure the reliant record also gets deleted.