The convention is not to capitalize the first letter of the **[verbose\_name](https://docs.djangoproject.com/en/2.1/ref/models/fields/" \l "django.db.models.Field.verbose_name" \o "django.db.models.Field.verbose_name)**. Django will automatically capitalize the first letter where it needs to.

It’s suggested, but not required, that the name of a **[ForeignKey](https://docs.djangoproject.com/en/2.1/ref/models/fields/" \l "django.db.models.ForeignKey" \o "django.db.models.ForeignKey)** field (**manufacturer** in the example above) be the name of the model, lowercase. You can, of course, call the field whatever you want. For example:

It’s suggested, but not required, that the name of a **[ManyToManyField](https://docs.djangoproject.com/en/2.1/ref/models/fields/" \l "django.db.models.ManyToManyField" \o "django.db.models.ManyToManyField)** (**toppings** in the example above) be a plural describing the set of related model objects.

Generally, **[ManyToManyField](https://docs.djangoproject.com/en/2.1/ref/models/fields/" \l "django.db.models.ManyToManyField" \o "django.db.models.ManyToManyField)** instances should go in the object that’s going to be edited on a form. In the above example, **toppings** is in **Pizza** (rather than **Topping** having a **pizzas** **[ManyToManyField](https://docs.djangoproject.com/en/2.1/ref/models/fields/" \l "django.db.models.ManyToManyField" \o "django.db.models.ManyToManyField)** ) because it’s more natural to think about a pizza having toppings than a topping being on multiple pizzas. The way it’s set up above, the **Pizza** form would let users select the toppings.

**Extra fields on many-to-many relationships**