# Know how to retrieve all users.

*User.objects.all()*

# Know how to get the last user.

*User.objects.last()*

# Create a few records in the users

User.objects.create({{field1}}="{{value}}", {{field2}}="{{value}}", etc)

# Know how to get the first user.

*User.objects.first()*

# Know how to get the users sorted by their first name (order by first\_name DESC)

*User.objects.order\_by(‘-first\_name’)*

# Get the record of the user whose *id* is 3 and UPDATE the person's *last\_name* to something else. Know how to do this directly in the console using *.get* and *.save.*

*x = User.objects.get(id=3)*

*x.last\_name = “Newname”*

*x.save()*

# Know how to delete a record of a user whose id is 4 (use something like *User.objects.get(id=2).delete*...)

delete = User.objects.filter(first\_name="Bobby")

delete.delete()

# (optional) Ninja:

# Find a way to validate the data coming into the shell.  For example, make sure that "name" fields are a minimum length, "email" is a valid email, or that "email" doesn't already exist in the db.

There are three steps involved in validating a model:

1. Validate the model fields - **[Model.clean\_fields()](https://docs.djangoproject.com/en/2.0/ref/models/instances/" \l "django.db.models.Model.clean_fields" \o "django.db.models.Model.clean_fields)**
2. Validate the model as a whole - **[Model.clean()](https://docs.djangoproject.com/en/2.0/ref/models/instances/" \l "django.db.models.Model.clean" \o "django.db.models.Model.clean)**
3. Validate the field uniqueness - **[Model.validate\_unique()](https://docs.djangoproject.com/en/2.0/ref/models/instances/" \l "django.db.models.Model.validate_unique" \o "django.db.models.Model.validate_unique)**

All three steps are performed when you call a model’s **[full\_clean()](https://docs.djangoproject.com/en/2.0/ref/models/instances/" \l "django.db.models.Model.full_clean" \o "django.db.models.Model.full_clean)** method.