

# Module 2 Cheat Sheet: ORM: Bridging the Gap Between the Real World and Relational Model

Package/ Method	Description	Code Example
<b>django.db.models.Model</b>	Define a model.	<pre> from django.db import models class MyModel(mod els.Model):     field1 = models.CharFi eld(max_lengt h=100)     field2 = models.Integer Field() python manage.py makemigration s python manage.py migrate </pre>
<b>makemigrations/migrate</b>	Create database tables based on models.	
<b>all()</b>	Retrieves all instances of the 'MyModel' model from the database.	<pre> MyModel.objects.all() MyModel.objects.filter(field1="value") </pre>
<b>filter()</b>	Filter objects using conditions.	<pre> MyModel.objects.filter(field2__gt=5) </pre>
<b>get()</b>	Retrieves a single instance of the 'MyModel' model from the database where the value of 'field1' is "value".	<pre> MyModel.objects.get(field1="value") </pre>

<b>obj = MyModel(field1="value", field2=5) obj.save()</b>	Creates a new instance of the 'MyModel' model with the values "value" for 'field1' and 5 for 'field2', and then saves the instance to the database.	obj = MyModel(field1="value", field2=5) obj.save()
-----------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------

<b>obj.field1 = "new value" obj.save()</b>	Updates the value of 'field1' for the 'obj' instance to "new value" and saves the changes to the database.	obj.field1 = "new value" obj.save()
----------------------------------------------------	------------------------------------------------------------------------------------------------------------	-------------------------------------------

<b>delete()</b>	Deletes an object.	obj.delete()
-----------------	--------------------	--------------

<b>obj.related_model</b>	Retrieves the related model associated with the 'obj' instance. Access related objects (Foreign Key or OneToOneField)	obj.related_model
--------------------------	--------------------------------------------------------------------------------------------------------------------------	-------------------

<b>obj.model_set.all()</b>	Fetches all related objects associated with the 'obj' instance. Access related objects in reverse (ForeignKey)	obj.model_set.all()
----------------------------	-------------------------------------------------------------------------------------------------------------------	---------------------

<b>field</b>	Performs a filtering operation on the 'MyModel' model instances based on a related model's field value.	MyModel.objects.filter(related_model__field="value")
--------------	---------------------------------------------------------------------------------------------------------	------------------------------------------------------

**exact** Retrieves instances of the 'MyModel' model from the database where the value of the 'field' attribute MyModel.obje is exactly cts.filter(field\_ equal to \_exact="value" "value". )

**ixact** The iexact lookup is case-insensitive, meaning it will match values regardless of whether they are uppercase or lowercase and provide a MyModel.obje case- cts.filter(field\_ insensitive \_ixact="value match. ")

**contains** Checks if the value is a substring within the field. MyModel.obje cts.filter(field\_ \_contains="val ue")

**startswith** Determines whether a string begins with the characters of a specified string. MyModel.obje cts.filter(field\_ \_startswith="v alue")

**endswith** Determines whether a string ends with the specified suffix. MyModel.obje cts.filter(field\_ \_endswith="va lue")

**in** Checks if the value of the field is present in the given list of values. MyModel.obje cts.filter(field\_ \_in=["value1", "value2"])

**gt** Checks if the value of 'field' MyModel.obje is numerically cts.filter(field\_ greater than 5. \_gt=5)

It Checks if the  
value of 'field' MyModel.obje  
is numerically cts.filter(field\_  
less than 10. \_lt=10)