

Django ORM Lookups Cheat Sheet with Example Models

Example Models

```
from django.db import models

class Author(models.Model):
    name = models.CharField(max_length=100)
    age = models.IntegerField()
    is_active = models.BooleanField(default=True)

class Tag(models.Model):
    name = models.CharField(max_length=50)

class Entry(models.Model):
    title = models.CharField(max_length=200)
    pub_date = models.DateField()
    published = models.BooleanField(default=False)
    author = models.ForeignKey(Author, on_delete=models.CASCADE)
    tags = models.ManyToManyField(Tag)
    price = models.DecimalField(max_digits=6, decimal_places=2)
```

Date / Time lookups

```
Entry.objects.filter(pub_date__year=2026)
Entry.objects.filter(pub_date__month=1)
Entry.objects.filter(pub_date__day=29)
Entry.objects.filter(pub_date__week=5)
Entry.objects.filter(pub_date__week_day=3) # Sunday=1, Monday=2, ...
Entry.objects.filter(pub_date__quarter=1)
Entry.objects.filter(pub_date__gte='2026-01-01')
Entry.objects.filter(pub_date__lte='2026-12-31')
Entry.objects.filter(pub_date__range=['2026-01-01', '2026-12-31'])
```

2 String / Text lookups

```
Entry.objects.filter(title__exact='Python')
Entry.objects.filter(title__iexact='python')
Entry.objects.filter(title__contains='Django')
Entry.objects.filter(title__icontains='django')
Entry.objects.filter(title__startswith='The')
Entry.objects.filter(title__istartswith='the')
Entry.objects.filter(title__endswith='Entry')
Entry.objects.filter(title__iendswith='entry')
Entry.objects.filter(title__regex=r'^Py.*$')
Entry.objects.filter(title__iregex=r'^py.*$')
```

3 Numeric lookups

```
Entry.objects.filter(id__gt=5)
Entry.objects.filter(id__gte=5)
Entry.objects.filter(id__lt=10)
Entry.objects.filter(id__lte=10)
Entry.objects.filter(id__exact=7)
Entry.objects.filter(id__in=[1,3,5])
Entry.objects.filter(price__range=[100,500])
```

4 Boolean / Null lookups

```
Entry.objects.filter(published__exact=True)
Entry.objects.filter(published__exact=False)
Entry.objects.filter(pub_date__isnull=True)
Entry.objects.filter(pub_date__isnull=False)
```

5 Relational lookups (ForeignKey / OneToOne / ManyToMany)

```
Entry.objects.filter(author__name='Faruk')
Entry.objects.filter(author__age__gte=18)
Entry.objects.filter(author__is_active=True)
Entry.objects.filter(tags__name__icontains='python')
```

6 Miscellaneous / PostgreSQL specific

```
Entry.objects.filter(title__length=10)
Entry.objects.filter(title__startswith='A')
Entry.objects.filter(title__contains='abc')
```

7 Combining lookups

```
# Multiple conditions
Entry.objects.filter(pub_date__year=2026, title__icontains='Python')

# Exclude
Entry.objects.exclude(pub_date__year=2026)

# Chaining
Entry.objects.filter(pub_date__year=2026).filter(title__icontains='Python')
```

Tip: All lookups are used in the format `field__lookup=value`. You can chain multiple filters and use `.exclude()` for negation.