Tools the ORM provides

> Atomic transactions   **transaction.atomic()**

> Locking   **Model.objects.select_for_update()**

> Compare-and-swaps   **.filter(val=expected).update(val=new)**

Atomic Transactions

```python
with transaction.atomic():

    thing = SomeModel.objects.create(...)
    other_thing = SomeModel.objects.create(...)

    if error_condition(...):
        raise Exception('Rolls back entire transaction')
```

Exceptions roll back the entire transaction block.

Neither object will be saved to the DB.

Transactions can be nested.