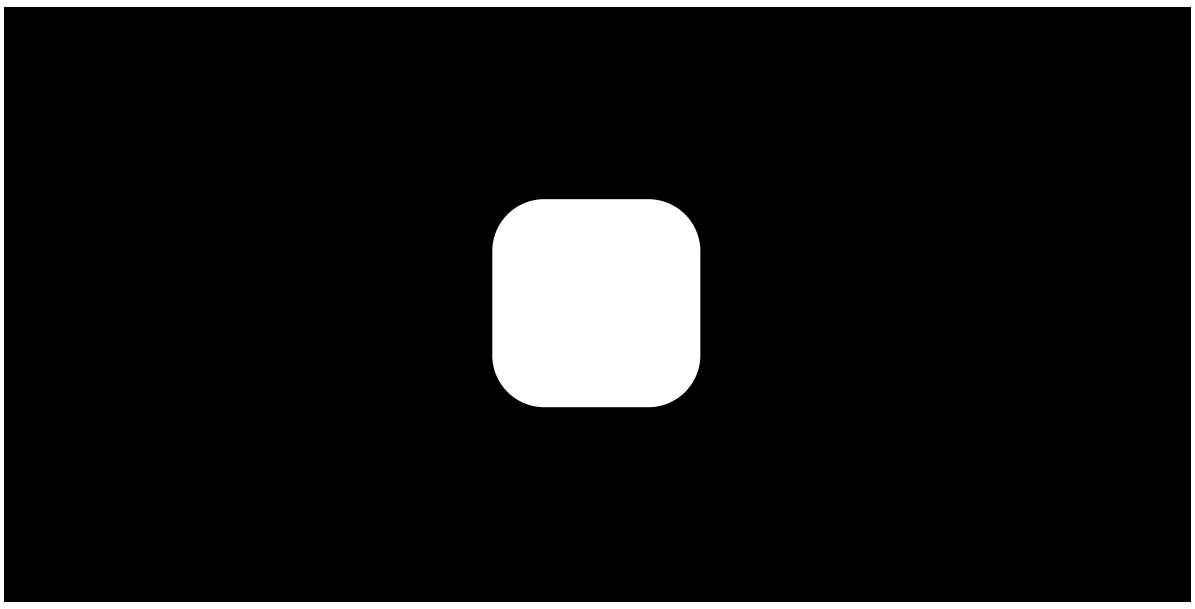
Part 1: Introduction

Bookmark this page

Part 1: Introduction

Mongo DB



Start of transcript. Skip to the end.

In this lesson, we're going to move on from looking at relational databases

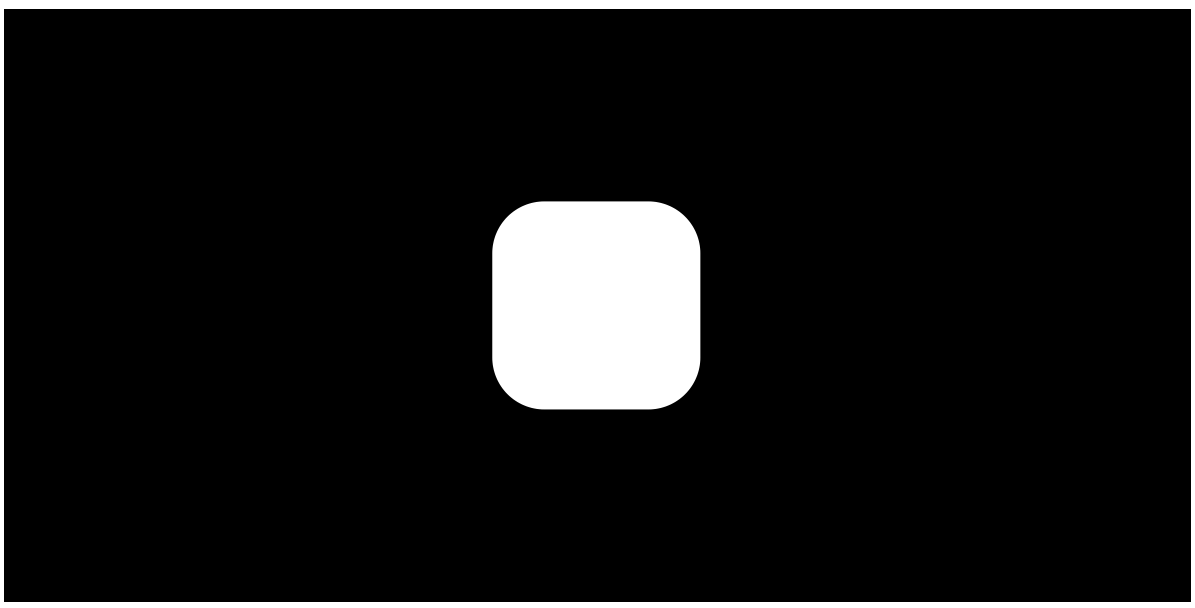
and we're going to look at NoSQL databases,

specifically one called MongoDB.

Peewee does a great job of making relational databases easy to use,

but MongoDB seems to take it to an extra level.

Introduction - an API perspective



Start of transcript. Skip to the end.

As with any language, when using Python to develop business applications,

you nearly always have to deal with additional runtime software

to help get your job done.

Examples of such software include message queue

in packages, such as Rabbit MQ, business rows engines, and of course databases

Video

Download video file

Transcripts

Download SubRip (.srt) file

Download Text (.txt) file

Code from the video

```
class BaseModel(Model):
    class Meta:
        database = database

class Person(BaseModel):
```

```
class Person(BaseModel):
    """
      This class defines Person, which maintains details of someone
      for whom we want to research career to date.
    """

    person_name = CharField(primary_key = True, max_length = 30)
    lives_in_town = CharField(max_length = 40)
    nickname = CharField(max_length = 20, null = True)
```



```
class Job(BaseModel):
    """
      This class defines Job, which maintains details of past Jobs
      held by a Person.
    """
    job_name = CharField(primary_key = True, max_length = 30)
    start_date = DateField(formats = 'YYYY-MM-DD')
    end_date = DateField(formats = 'YYYY-MM-DD')
    salary = DecimalField(max_digits = 7, decimal_places = 2)
    person_employed = ForeignKeyField(Person, related_name='was_filled_by', null = False)
new_person = Person.create(
   person_name = 'Fred',
   lives_in_town = 'Seattle',
   nickname = 'Fearless')
    new_person.save()

aperson = Person.get(Person.person_name == 'Fred')
```