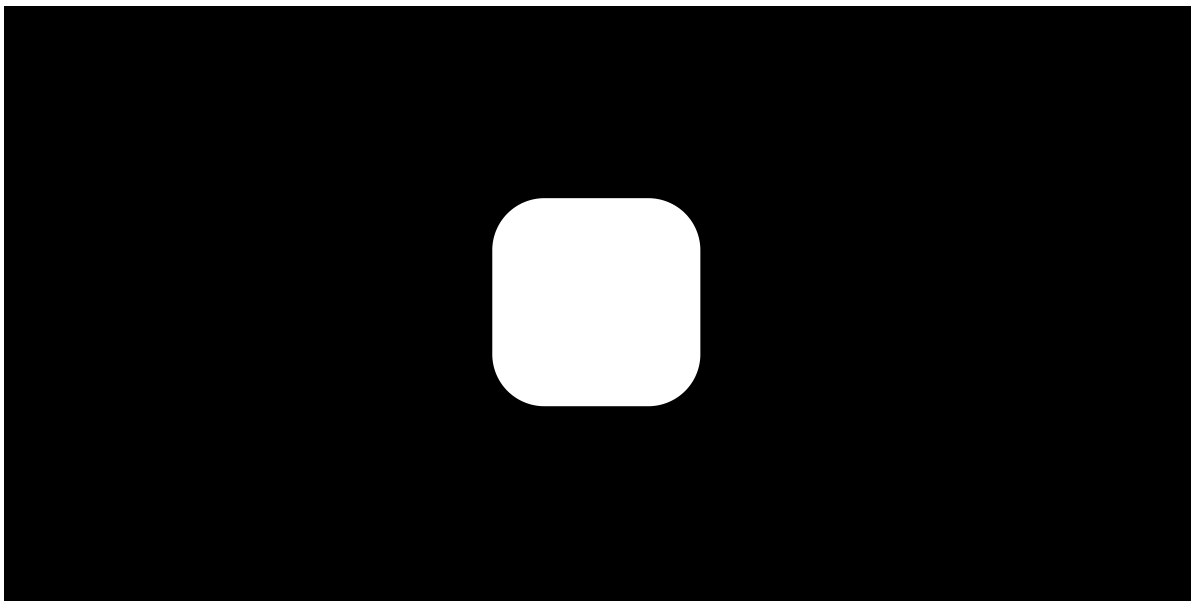
Part 3: Building the server side API

Part 3: Building the server side API

Flask Overview



Start of transcript. Skip to the end.

[MUSIC PLAYING]

To make our API work, we need to import several different packages.

And these will all obviously need to be pip installed prior to importing them.

So on line 1 from Flask, we import something called Flask.

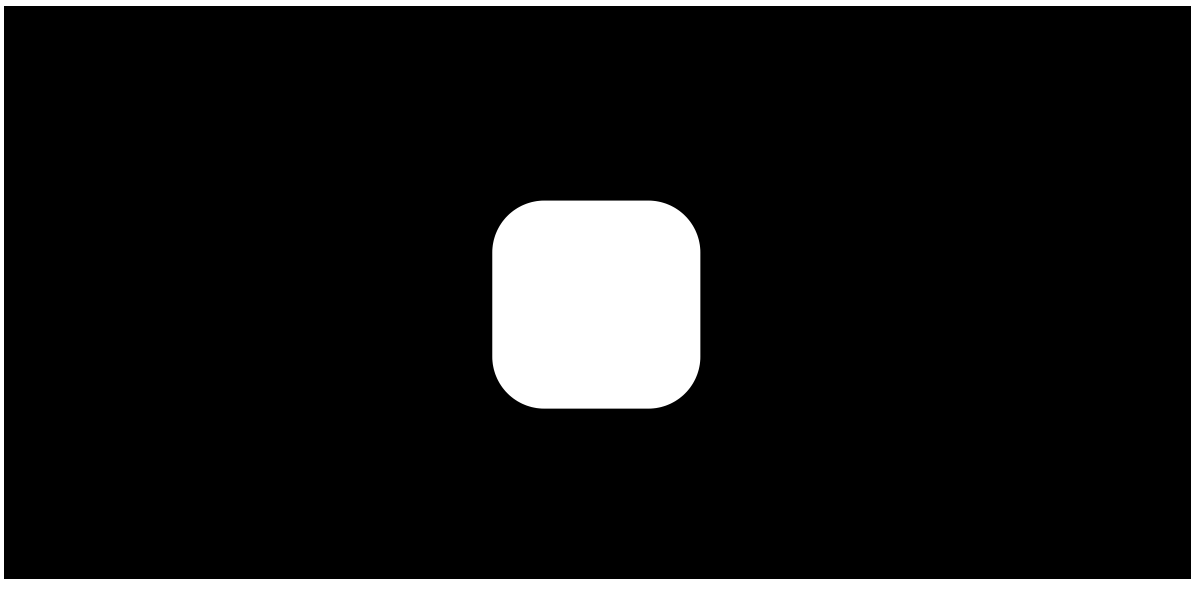
We then also need to import Resource and API

from something called Flask Restful

Here is the code used in the video:

```
if __name__ == "__main__":
    db_connect = create_engine("sqlite:///chinook.db")

    app = Flask(__name__)

    api = Api(app)
    api.add_resource(Employees, "/employees") # Route_1
    api.add_resource(Tracks, "/tracks") # Route_2
    api.add_resource(Employees_Name, "/employees/<employee_id>") # Route_3

    app.run(port="5002")

    db_connect.dispose()
```

API 3



Start of transcript. Skip to the end.

[MUSIC PLAYING]

Now we have the various different packages imported.

And we have the basic structure of our main logic.

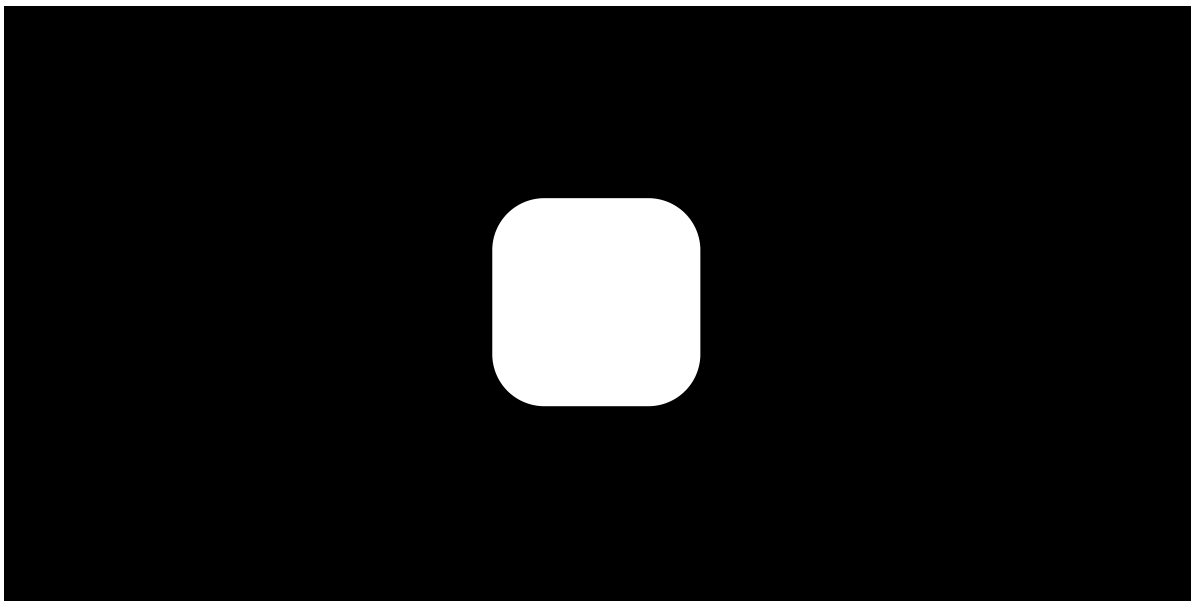
Let's look now at how we get data that's going to be available to those URLs

that we set up in the right section.

So you see here that we have three classes that I've created--

API 4

Previous Next



[MUSIC PLAYING]

So having looked at employees, let's now look at tracks.

So tracks is another class.

It inherits from resource.

It has a get method because all we're doing

is retrieving data, in a similar pattern.

It creates a connection, builds a query,