

Test 1

In this task you have to design a web API application that will support the database presented in the diagram. We create a website that should allow you to view data on prescriptions, doctors and patients.

1. Design an endpoint that will return data about a particular doctor and a list of prescriptions. Endpoint should take doctor's id as parameter. Prescription data should be returned in descending order by the Date column. Remember to return the appropriate error codes.

```
2. GET /api/doctors/5 HTTP/1.1  
3. Host: localhost:5000
```

2. Prepare an endpoint that will allow you to delete data about a given doctor. In a situation when a given doctor has issued some prescriptions, then we also want to delete them (together with data from the association table). Remember that the endpoint should use the transaction to maintain database consistency.

```
1. DELETE /api/doctors/10 HTTP/1.1  
2. Host: localhost:44391
```

1. Remember about:
 - a. Name correctly the variables, controllers etc.
 - b. Remember to maintain proper code structure.
 - c. Remember to inject dependencies.
 - d. Remember to handle errors.
 - e. Place the code in the new repository. Remember about the .gitignore file and push changes to the server.

