Oracle MOOC: Node JS Microservices for Oracle Cloud

Session 1

Week 2 Homework

You will learn:

- Modularize Node.JS microservices in layers.
- Create Oracle Database users, tables and data.
- Connect Node.JS applications to Oracle databases
- Use promises in your code to simplify asynchronous code.

You will need:

- Download and setup your Oracle pre-built Developer VM
- Complete previous homework

Note:

- Discuss among students in the forum for further learning
- The homework is NOT mandatory
- You may download a solution to the previous homework from the course materials although I encourage you to try it on your own.

Oracle MOOC: Node JS Microservices for Oracle Cloud

Application Overview

The application you must build in your homework is a Movie Collection and Reviews RESTful web service.

The application stores and lets you manage a movie collection.

Movies and Review objects are discussed in the Week 1 Homework.

This time you are tasked with completing the RESTful API of the application:

HTTP METHODS	JSON OBJECTS
GET / returns a list of movie summaries REQUEST: empty RESPONSE: Array<OUT_SUMMARY> **GET /movieId** returns a movie with its reviews REQUEST: empty RESPONSE: OUT_MOVIE **GET /movieId/reviewId** returns a review of a movie REQUEST: empty RESPONSE: OUT_REVIEW **POST /** add a movie REQUEST: IN_MOVIE RESPONSE: the id of the movie added. **POST /movieId** add a review to the movie REQUEST: IN_REVIEW RESPONSE: the id of the review added. **PUT /movieId** Update a movie REQUEST: IN_MOVIE RESPONSE: empty **PUT /movieId/reviewId** Update a review REQUEST: IN_REVIEW RESPONSE: empty	**<OUT_MOVIE>** id : number name : string year : number studio : string genre : string rating : string runtime : number director : string description : string reviews : array<OUT-REVIEW> **<IN_MOVIE>** name : string year : number studio : string genre : string rating : string runtime : number director : string description : string **<OUT_REVIEW>** id : number name : string date : string score : number description : string **<IN_REVIEW>** name : string date : string score : number description : string

Oracle MOOC: Node JS Microservices for Oracle Cloud

DELETE /movieId Deletes a movie and all its reviews REQUEST: empty RESPONSE: empty **DELETE /movieId/reviewId** Deletes a movie a review from the movie REQUEST: empty RESPONSE: empty	**<OUT_SUMMARY>** id : number name : string year : number studio : string genre : string rating : string runtime : number director : string description : string reviews : number score : number

Oracle MOOC: Node JS Microservices for Oracle Cloud

Assignment 2-1: Implement the Movies API

You are part of the development team that is adopting NodeJS to create microservices. Your first assignment is to provide a microservice to query data in your NodeJS application from a JSON file.

Separate into layers and implement all routes

1. Start by separating you application in modules. The easiest one to start with is the configuration, then do the server since that dictates the structure, then the router, controller and data.
2. You may use the data in memory, or start with the database.
3. Use promises when possible, they will save a lot of time.
4. Focus on having all the routes and controller methods.

Oracle MOOC: Node JS Microservices for Oracle Cloud

Assignment 2-2: Connect the Application to a Database

Now that you have all the routes and controller methods focus on connecting to the Oracle Database.

Connect to the Database

1. Install the Oracle database, REMEMBER NOT to save the dependency in the package.json file.
2. Create the database module and use the environment variables to open connections.
3. Use promises to query data.
4. Use the code in this week as a guide but try to figure out how to build your own solution.

Database configuration environment variables.

```
db: {
  connectString: process.env.DBAAS_DEFAULT_CONNECT_DESCRIPTOR || "localhost/orcl",
  user: process.env.DBAAS_USER_NAME || "movies_usr",
  password: process.env.DBAAS_USER_PASSWORD || "oracle"
}
```

Tip:

In the course videos I show how to build most of the homework, if you get stuck watch the lessons and follow along.

Congratulations, you have successfully completed homework for Week 2 of NodeJS Microservices!