

Practice Implement Stateless Communication Using REST APIs and JSON



Practices

- Practice 1: List installation and start up details of json-server
- Practice 2: Make an HTTP Request to Fetch Movies
- Practice 3: Make an HTTP Request to Add a Movie
- Practice 4: Make an HTTP Request to Update a Movie
- Practice 5: Make an HTTP Request to Delete a Movie

Implementation Environment

- Perform the following tasks to setup the implementation environment:
 - Task 1: Install Node.js
 - Task 2: Install json-server
 - Task 3: Install Postman app

Note: Task details are given in the upcoming slides.

An illustration of a woman with dark hair and glasses, wearing a red top, and a man with brown hair and glasses, wearing an orange top. They are sitting at a desk with a large blue computer monitor. The woman is holding a yellow clipboard. On the desk, there is a coffee cup, a pencil, and a notepad. The background is light green with some abstract shapes and a large green plant on the right.

PRACTICE

Practice 1: List the Installation and Startup Details of json-server

List version information of Node.js, npm and json-server.

Also, list down the json-server startup details.

Instructions for Practice 1

- Start json-server from the folder **p1-list-details**, to read data from **db.json** file.
- Note down the installation and the startup details as mentioned in the file **p1-submission.md** located inside the same folder.
- The details should be noted in the same file.
- Submit the solution for evaluation.

An illustration of a woman with dark hair and glasses, wearing a red top, and a man with brown hair and glasses, wearing an orange top. They are sitting at a desk with a large blue monitor. The woman is holding a yellow clipboard. On the desk, there is a coffee cup, a pencil, and some papers. The background is light green with some abstract shapes and a large green leaf on the right.

PRACTICE

Practice 2: Make an HTTP Request to Fetch Movies

To retrieve movie data from the **movies.json** file using json-server, list down the HTTP request URLs to fetch movies based on the criteria specified in the upcoming slide.

Note:

You may write the request URLs in the Postman app directly and test it.

Instructions for Practice 2

- Start the json-server. The server should read the data from **movies.json** file which is provided in the boilerplate.
- Specify the request and response details in the **p2-submission.csv** file located inside folder **p2-make-http-fetch-request** of the boilerplate.
- Write HTTP requests to fetch:
 - all movies' data
 - a movie having the id `502`
 - a movie having the id `600`
 - a movie having the title `365 Days`
 - movies having the genre `Action`
 - movies having the genre `Action` and released in the year `2022`
- Test the request URLs using Postman and note the response details.
- Submit the file for evaluation.

An illustration of a woman with dark hair and glasses, wearing a red top, and a man with brown hair and glasses, wearing an orange top. They are sitting at a desk with a large blue computer monitor. The woman is holding a yellow clipboard. On the desk, there is a coffee cup, a pencil, and a notepad. The background is light green with some abstract shapes and a large green plant on the right.

PRACTICE

Practice 3: Make an HTTP Request to Add a Movie

The list of movies stored in `movies.json` file is accessible through the json-server. Write HTTP request URL to add a movie with the details provided in upcoming slide .

Note:

You may directly enter the request URLs and test them in the Postman app.

Instructions for Practice 3

- Start the json-server. The server should read the data from **movies.json** file which is provided in the boilerplate.
- Specify the request and response details in the **p3-submission.csv** file located inside folder **p3-make-http-save-request** of the boilerplate, to add movies with the details given in the table below:

Id	MovieTitle	Year of Release	Genre	Director	Actor-1	Actor-2	Actor-3	Rating	On OTT	Ott
514	Father of the Bride	2021	Adventure	Gary Alazraki	Actor name: Andy Garcia Screen name: Billy Herrera	Actor name: Gloria Estefan Screen name: Ingrid Herrera	Actor name: Adria Arjona Screen name: Sofia Herrera	5.9 / 10	No	
503	Spiderhead	2022	Action	Joseph Kosinski	Actor name: Chris Hemsworth Screen name: Abnesti	Actor name: Miles Teller Screen name: Jeff	Actor name: Jurnee Smollett Screen name: Lizzy	5.4 / 10	No	

- Test the request URLs using Postman.
- Submit the file for evaluation.

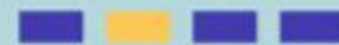
PRACTICE

Practice 4: Make an HTTP Request to Update a Movie

The list of movies stored in `movies.json` file is accessible through the `json-server`. Write HTTP request URL to update an existing movie with the details provided in the upcoming slide.

Note:

You may directly enter the request URLs and test them in the Postman app.



Instructions for Practice 4

- Start the json-server. The server should read the data from **movies.json** file which is provided in the boilerplate.
- Specify the request and response details in the **p4-submission.csv** file located inside folder **p4-make-http-update-request** of the boilerplate, to update movies with the details given in the table below:

ID	Movie Title	Year of Release	Genre	Director	Actor-1	Actor-2	Actor-3	Rating	On OTT	Ott
514	Father of the Bride	2022	Comedy	Gary Alazraki	Actor name: Andy Garcia Screen name: Billy Herrera	Actor name: Gloria Estefan Screen name: Ingrid Herrera	Actor name: Adria Arjona Screen name: Sofia Herrera	5.9 / 10	Yes	Netflix
500	Top Gun	1986	Action	Tony Scott	Actor name: Tom Cruise Screen name: Maverick	Actor name: Tim Robbins Screen name: Merlin	Actor name: Kelly McGillis Screen name: Charlie	6.9 / 10	Yes	Prime Video

- Test the request URLs using Postman.
- Submit the file for evaluation.

An illustration of a woman with dark hair and glasses, wearing a red top, and a man with brown hair and glasses, wearing an orange top. They are sitting at a desk with a large blue monitor. The woman is holding a yellow clipboard. On the desk, there is a coffee cup, a pencil, and a notepad. The background is light green with some abstract shapes and a large green leaf on the right.

PRACTICE

Practice 5: Make an HTTP Request to Delete a Movie

The list of movies stored in `movies.json` file is accessible through the `json-server`. Write HTTP request URL to delete an existing movie.

Note:

You may directly enter the request URLs and test them in the Postman app.

Instructions for Practice 5

- Start the json-server. The server should read the data from movies.json file which is provided in the boilerplate.
- Specify the request and response details in the **p5-submission.csv** file located inside folder **p5-make-http-delete-request** of the boilerplate, to delete movies with IDs 501 and 550.
- Test the request URLs using Postman.
- Submit the file for evaluation.