Branch for this assignment is **assignment/get-api**

1. Create an API for **GET /movies** that returns a list of movies. Define an array of movies in your code and return the value in response.

1. Create an API **GET /movies/:indexNumber** (For example **GET /movies/1** is a valid request and it should return the movie in your array at index 1). You can define an array of movies again in your api

[‘Rang de basanti’, ‘The shining’, ‘Lord of the rings’, ‘Batman begins’]

Example of a request url -> localhost:3000/movies/2

1. Handle a scenario in problem 2 where if the index is greater than the valid maximum value or smaller that the valid minimum value, a message is returned that tells the user to use a valid index in an error message.
2. Write another api called **GET /films**. Instead of an array of strings define an array of movie objects this time. Each movie object should have attributes - id and name. An example of movies array is

[ {

“id”: 1,

“name”: “The Shining”

}, {

“id”: 2,

“name”: “Incendies”

}, {

“id”: 3,

“name”: “Rang de Basanti”

}, {

“id”: 4,

“name”: “Finding Nemo”

}]

Return the entire array in this api’s response

1. Write api **GET /films/:filmId** where filmId is the value received in request path params. Use this value to return a movie object with this id. In case there is no such movie present in the array, return a suitable message in the response body. Example for a request **GET /films/3** should return the movie object

{

“id”: 3,

“name”: “Rang de Basanti”

}

Similarly for a request **GET /films/9** the response can be something like - **‘No movie exists with this id’**