Instructions:

1. This question paper consists of two sections, **A** of **60 marks** and **B** of **40 marks**.
2. All questions are mandatory.
3. The total time to complete is 4.5 hours.
4. For your submission, you are required to create a GitHub repository. The repository should contain a .doc or .pdf file for the theoretical questions and the project based questions should have their own folder.

Section-A (60 Marks)

Question 1-5 contains 1 mark each

1. Which of the following is the correct way to create a list using the lowercase letters?

   a. <ol alpha=”a”> b. <ol type=”a”> c. <ol letter=”a”> d. None

2. In HTML5, which of the following tag is used to initialize the document type?

   a. <Doctype HTML> b. <\Doctype html> c. <DOCTYPE> d. <!DOCTYPE html>

3. Which of the following has the highest priority in CSS?

   a. #id b. HTML tags c. .my-class d. :hover

4. The default value of “position” property is _____.

   a. fixed b. absolute c. static d. relative

5. How to stop event bubbling in JavaScript?

   a. e.stopBubble() b. e.stop() c. e.pauseBubble() d. e.stopPropagation()

6. Explain Closure, Hoisting and Currying (Function Composition) in JavaScript, with the help of examples. (**10 Marks**)

7. Demonstrate using code sample, multiple ways of using CSS in HTML. What is the priority order of the same? (**10 Marks**)

8. Find and explain the output of the following code snippet. (**5 marks**)

```
const arr = [10, 12, 15, 21];
for (var i = 0; i < arr.length; i++) {
  setTimeout(function() {
    console.log('Index: ' + i + ', element: ' + arr[i]);
  }, 3000);
}
```

9. Find and explain the output of the following code snippet. (**5 marks**)

```
const promise1 = new Promise((resolve, reject) => {
  console.log(1);
  resolve('success')
});
promise1.then(() => {
  console.log(3);
});
console.log(4);
```

10. Design and develop a random quote project as below, **(25 marks)**

GIF

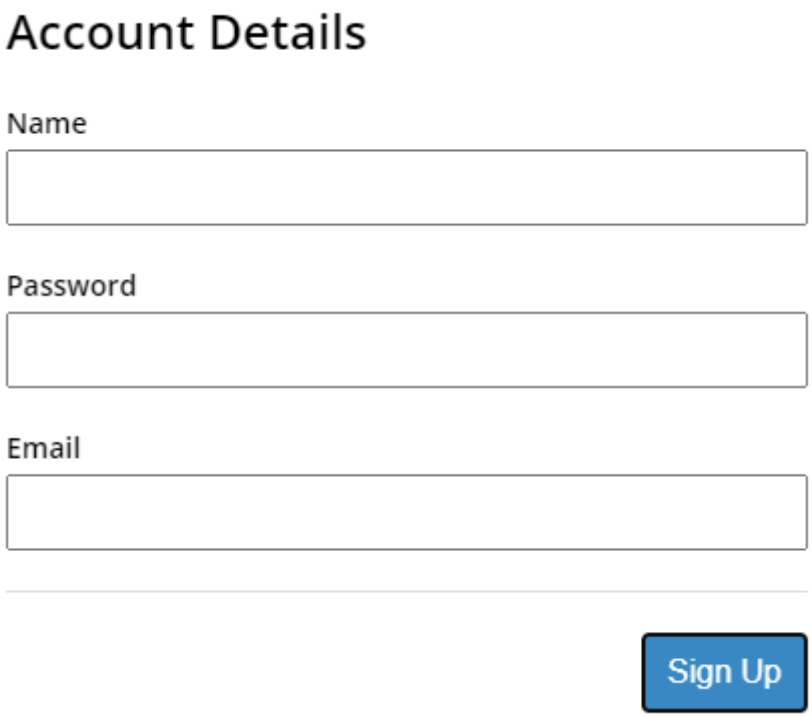
You can use this API(Type fit) to get a list of quotes.

User Stories:

- Clicking on the “Generate Quote” button will show me a random quote.
- The first time load of the page should also contain a random quote by default.

Section-B (40 Marks)

11. Design and develop a simple user sign up form as such: (**20 Marks**)



- Accept all the user input in browser based JavaScript and send that data as JSON to your backend server on the route **/userSignup** as a **POST** request.
- In your server, save the incoming user sign up data in a **users.json** file.
- Note: Your HTML should also be served through the server.
- Note: If more user sign ups are done all those user information should go inside that json file.
- Here is a sample users.json file after the process.

```
[
  {
    "name": "John",
    "email": "John.Doe@gmail.com",
    "password": "123432"
  },
  {
    "name": "Foo Bar",
    "email": "Foo.bar@gmail.com",
    "password": "77858asd"
  },
  ...
]
```

12. Download **netflix_titles.csv** from this link. Create a node.js program which will read the content of this CSV file and convert it into a JSON/JS object which you need to write to a **netflix.json** file as an array.

You need to go through the CSV file and understand all the fields and the type of data it stores and design your JSON accordingly. Make sure you DO NOT store comma separated values as string in your JSON object, use appropriate data structure. For ex, if the **director** has multiple director names, then it should be an array in you JSON object

Your program should also handle any exceptions if it arises.

Next, serve the created **netflix.json** through a route in your server as '**/netflix/{show_id}'** where the show_id is a path parameter, and will give the correct Netflix show as a JSON object**.**

(20 Marks)