

## Short Answer:

Answer the following questions with complete sentences in **your own words**. You are encouraged to conduct your own research online or through other methods before answering the questions. If you research online, please consult multiple sources before you write down your answers.

1. What is Node.js?
2. What are some differences between Node.js and JavaScript?
3. What is npm?
4. What is CommonJS? How does it differ from ES Modules?
  1. How do we import files into other files in Node.js?
5. How can you make the server automatically restart when files are modified?
6. What are some global objects in Node.js?
7. Explain how the Node.js architecture is event-driven in terms of event emitters and other core modules.
8. What are streams? In Node.js, what are the different kinds of streams?
9. What is the difference between fs module's `readFile` and `createReadStream`?
10. What are the different timing or scheduling functions in Node.js?
11. How does the event loop work in Node.js?
12. How do you manage multiple node versions?

## Coding Questions:

Use HTML/CSS/JS to solve the following problems. You are highly encouraged to present more than one way to answer the questions. Please follow best practices when you write the code so that it would be easily readable, maintainable, and efficient. Clearly state your assumptions if you have any. You may discuss with others on the questions, but please write your own code.

1. Sort the numbers:  
Write 100 random number (from 1-100) to a file.txt file, then read from the file.txt and sort all the number in ascending order, finally write all the sorted number back into the file.txt replacing the original content
2. To-Do App  
Implement a to-do app. The app should include following requirements. (Hints: some library might be helpful: yargs, fs).
  - 1) User should be able to login with fixed username(admin) and password(admin) using command `node app.js login --username='admin' --password='admin'`. If the username or the password is not correct, display "Wrong Username or Password"
  - 2) User can read from a file that includes all the to-do item and display them in console. If the file not exist, throw an error says 'file not exists'. Assume there is only one file.  
`node app.js read`
  - 3) User can add a to-do item, including title, description, status(NOT FINISH, FINISH), time(current timestamp). Every data field should be required except for the description. Assume the note title is unique  
`node app.js add --title='XXX' --desc='XXX' --status='XXX' --time='XXX'`
  - 4) User can finish an to-do item by changing the status from 'NOT FINISH' to 'FINISH'

```
node app.js finish --title='XXX'
```

5) User can delete a to-do item by providing the item title

```
node app.js delete --title='XXX'
```