MERN Assignment Questions

Week 1: Q.1 to Q.8
Week 2: Q.9 to Q.13
Week 3: Q.14 to Q.20
Week 4: Q.21 to Q.26
Week 5-6: Q.27 to Q.38
Week 7-8: Q.39 to Q.50

Note : For placement training batches after week-6 check for project implementation including
Express-Mongodb-React

Javascript: (10 days)

Week 1:

1. Wap to print swapping of 2 no's using javascript.

2. Wap to reverse three digit number.

3. Calculate factorial of a Given number

4. Wap to check prime no or not

5. Wap to matrix multiplication

6. Program to find the simple interest using function with argument & With return type.

7. Wap to show list of functions of array in JS

8. Explain JS object in breif and implement an login to fetch following JS objects

emp_details=[{‘eno’:1001001,’enm’:’amit’,’esal’:10000},{‘eno’:1001002,’enm’:’pankaj’,’esal’:20000},{‘eno’:1001003,’enm’:’ravi’,’esal’:30000}]

Week 2:

9. Have a form with two check boxes: "Chocolate" and "Vanilla" and a "Check order" button. Have the button open an alert box that states either "You like chocolate", "You like Vanilla", "You like both chocolate and vanilla", or "PICK ONE!" if neither was checked. Uncheck all boxes after any of the alerts are cleared.

10. Display 3 images and a single form text area on the page. Have 3 different words display in the text area, depending on which image you click.
Put the focus in the text area when the page has loaded.

11. Have a select list of ice cream flavors and a textbox. When you select a flavor, the textbox should display: "You picked [flavor chosen]."

12. Explain AJAX life cycle

13. Wap to generate an AJAX code to take response from a URL and display output on webpage

URL: https://api.myjson.com/bins/ms2y6

Node JS: (10 days)

Week 3:

14. Explain Blocking and Non Blocking functions with a examples in nodejs

16. Write a program to print Fibonacci Series using blocking function.

17. Write a program to check whether a number is Armstrong or not using non blocking function

18. Wap to create a funtional prototype in JS to take details of employee and print details
Details required : employee number, employee name, employee salary

18. Create custom module and upload on npm server

17. Create custom module calculate first number is divisible by second or not? and use in nodejs program

18. Read a file and count all word in file

19. Write to a program to save Academic Details(Name, Subject, Course) in file.

20. Write a program to send message from client to server with server port 8888.

Week 4:

21. Wap to create an http server listening at port 8081 and showing a messages for URL path

path	message
/	this is routing for home page
/home	this is routing for home page
/about	this is routing for about page
/contact	this is routing for contact page
/service	this is routing for service page

22. Wap to create database connectivity code for Nodejs-mysql

Express JS: (15 days)

23. Explain MiddleWare function , its type in breif

24. Implement Q.21 , in express and server port must be 3000

25. Generate an express app by express generator tool

26. Create an application with employee registration form containing following fields Employee_name, Employee_username, Employee_password, Employee_address,Employee_city, Employee_gender, Employee_hobbies insert details to MYSQL DATABASE and containing all JS form validations

Week 5:

27. Create an grid to show all registration records , and implement edit and delete operation in that grid

28. Implement login for above registration records and implement session in app for state management (User tracking & State management)

29. Implement file uploading in express including ,database implementation

30. In Q26 implement email sending ,for email sending

MongoDB: (7 days)

31. Write difference between Mysql and Mongodb

32. Create a document name **mean**

33. Create collection to store user details like(id, login, firstname, lastname, gender, state, city, emailid, contactno).

34. Create Employee collection with (empid,name,depart,salary,age)

Week 6:

35. Explain 10 operator variable with query

36. Write 5 aggregate function in mongodb

37. Write a command to fetch all user record

38. Write a command to mean document backup and restore

ReactJS: (18 days)

Week 7-8

39. Write difference between Reactjs and Angular cli

40. Create an react app and explain **index.js** file in breif

41. Implement following by using **ES6**

i. Wap to print swapping of 2 no's

ii. Wap to reverse three digit number.

iii. Calculate factorial of a Given number

iv. Wap to check prime no or not

v. Wap to matrix multiplication

vi. Program to find the simple interest using function with argument & With return type.

vii. Create a program to generate class in ES6 to take employee details like empid, empnm, empsal and to show them

vii. WAP to justify multilevel inheritance in ES6

viii. WAP to justify hybrid inheritance in ES6 , justify interface in ES6

42. Wap to create an react routing to load different component in UI

path	message
/	this is routing for home page(home component)
/home	this is routing for home page(home component)
/about	this is routing for about page((about component))
/contact	this is routing for contact page(contact component)
/service	this is routing for service page(service component)

43. Explain state & its types?

44. Explain props with example?

45. Create an state to display current date and time on ‘service’ component loaded in Q.42

46. Create an http service to call web service and display output on component

URL: https://api.myjson.com/bins/ms2y6

47. Explain with an example one way and two way binding.

48. Create an MERN app containing

React UI

Express Backend

MongoDB Database

49. Create an application with employee registration form containing following fields

Employee_name, Employee_username, Employee_password, Employee_address,Employee_city, Employee_gender, Employee_hobbies insert details to database using Express and mongodb

50. Create an grid to show all registration records , and implement edit and delete operation in that grid

51. Implement login for above registration records

52. Implement React validation in application