eSentire interview

* Introduction – keep notes of
  + Name, tech expertise, domain expertise, highlights from ZProfile – put security in the spotlight
  + Project information
* JS framework —- Node, React, Vue, Angular
* Python - Django, Flask
* The security side of the application – CORS, CSRF, OWASP 10
* AWS/Azure
* CSS
* Programming exercise
* DB related
* API security

—-----------—-----------—-----------—-----------—-----------—-----------—-----------—-----------—-----------

# JS interview questions:

* Explain how “this” works in JS?
* Pros and cons of using Promises instead of callbacks?
* Are you familiar with memoization for functions?

function multiply(x, y) {

console.log(‘RAN’)

return x\*y;

}

function memoize(fn) {

// implement the function here

return fn;

}

const m = memoize(multiply);

m(2, 3); // RAN, return 6

m(2, 3); // return 6

m(4, 5); // RAN, 20

m(2, 3); // return 6

* Give a comparison of current js frameworks and your experience with them.
* Describe your experience with unit test and automation frameworks.
* What is GraphQL and what problem does it solve?
* Write an iterator function:

let iterator = /// write a function

iterator(); // returns 1

iterator(); // returns 2

iterator(); // returns 3

iterator(); // returns 4

...

* If you were to design ReactJS, how would you do that at a high level?
* Difference between 1-way and 2-way binding.
* Explain HTTP requests as much as you can. Files vs body. Status codes. Pre-flight requests.
* CORS and DOM XSS attack.
* Explain js event propagation flow. How to override/manipulate it?

Javid:

* Questions based on previous projects.
* Life cycle methods in React.
* Questions based on hooks concepts (React).
* Converting class based components to functional components (React).
* Promises.
* Difference between shallow copy & deep copy.

Bhavesh

* How to create library using vanilla js.
* How to create web components using vanilla js.
* Security related question (XSS).
* Find the duplicate value form the Array.

Sreejith:

I)

* Tell about yourself
* What problem Typescript solves ?
* Http methods ? Option used for ?
* Content-type ? What types can be used ?
* CORS policy changes made at backend or frontend ?
* Status code (404, 500)
* Asked whether you have worked on any other frameworks/libraries other than react ? Differences between any two that you have worked or used ?
* Two way binding, one way binding pros cons
* Memoize problem from above
* Promises
* Callback
* prototype
* Graphql ? What does it solve ?
* Call, apply, bind
* Div, span difference
* What all display properties do we have ? Can you name a few ?
* Why flex? What does it solve ?
* Sass explain
* Event delegation, given a div structure and asked to explain how the events will be fired ? event bubbling, how to stop it ?
* Output of below code

function abc(){

return function(){

console.log(this);

}

}

var a = new abc();

a();

Last 10 minutes he explained about esentire, the product they are hiring for, tech stack, and way of working. Explained how they are using Vue and their plan to move to Vue3.

II)

* Write a function to Deep clone object.
* Write a function to Curry a function Sum(a,b…) to accommodate multiple parameters as well ?
* Write a function that repeats a string 1lakh (n) times depending on the counter being passed.
  + ‘Sreejith’.repeat(1000000)
  + o/p ‘SreejithSreejithSree……………………………..’
  + What will be the optimum solution in regards to space and time complexity ?
* Code review points to keep in mind.
* To call attention to an approach for a new functionality, what all points will you pitch in or how will you handle the situation with your teammates.
* Architect a new project from the start, what all points will you follow ?
  + [folder structure, security[what all security to keep in mind ?], maintainability, scalability, test, performance, code splitting, webpack ]

—-----------—-----------—-----------—-----------—-----------—-----------—-----------—-----------—-----------

# Python interview questions:

* What is multi-threading in Python?
* How we can handle simultaneous operation on the same column of the same table in the database?
* How does Book My Show manage a seat booking system? Does it mean that if one user is booking the ticket for the same show and at the same time another user is making a get request to check the seat availability of the same show then how do they handle such a thing?
* How did you manage authentication in REST api?
* How did you manage registration in your last project?
* Did you follow REST api standards? If yes then what kind of standards did you follow?
* When do we choose to pass the path parameter and when do we choose to pass the query parameter in REST api?
* In your last project in which scenarios you have used Azure?
* Have you managed session timeout in your project?
* If you face any issues in the production environment then did you provide any downtime page?
* If an error occurs in the production environment and you don't want to have a downtime page or you don't want to show a downtime page then how would you manage to production environment until you figure out your problem?
* In a microservice architecture, if any service falls down due to some error then how would you handle that fall down? like how would you track what error cause this fall down or something?
* If a large number of requests come to your application then how would you manage your traffic?
* What is the benefit of microservice architecture over monolithic architecture?
* Why did you choose only MySQL in your project? Why you did not use NoSQL or MongoDB?
* What is the difference between the POST and PUT methods?
* Which one is the idempotent method between the POST and PUT method?
* Have you ever worked with caching? (i.e. : Redis)
* Have you configured the expiry time of data in the cache? Like data older than 30 days will get clear automatically.
* If you have fetched the data in caching from the database and then someone has made a change to the data in the database then how would you update the cache with new data?
* What is LinkedList data structure?
* Do you have any idea about DB algorithms?
* Write a program to print a reverse string with recursion and explain it. Explain how we can optimize it?