## Stage 1

Make them comfortable

#### Introduction

#### Hobbies (Guitar, Football, Travel, Computer Games)

Do you have any hobbies besides coding?

Choose one hobby and ask any question among them [Add more if there are any hobbies besides this]# NOTE: only doing this in development as some production environments (Heroku)

# NOTE: are sensitive to local FS writes, and besides -- it's just not proper

# NOTE: to have a dev-mode tool do its thing in production.

if Rails.env.development?

RailsERD.load\_tasks

end

1. Guitar

* How proficient are you in playing Guitar?
* Which song can you play flawlessly?

1. Football

* Who is your favorite football player?
* Do you watch football matches? Which league do you watch?
* Which club do you support?

1. Travel

* Where have you traveled to recently?
* What is your dream destination for travel?

1. Computer Games

* Which computer game is your favorite till now?
* Will you be buying a new PS or Xbox?
* Do you play FIFA or PES? Rate yourself on the scale of 1 to 10.

## Stage 2

Get to the technical aspect

#### 

#### Recent projects

Ask about the recent project they did

1. Stacks
2. Challenges
3. Learning

#### General Programming

1. What is the one thing that you will do differently if you were to do the same project again? *(Checking spontaneity, presence of mind and how they handle the pressure!)*
2. How do you keep up with the latest trends and technologies? *-> Helps us in identifying if they stay up to date with current affairs in tech world which is crucial for Seniors devs*
3. Have you attended any JS meetups or conferences? *-> Helps in knowing how much they like learning from others and eager to find out how others build their applications or how other think*
4. Follow up if they have with “What was your objective in going to those events?”
5. What are some things that you look at when searching for new packages to integrate in the project? This should be asked with scenarios like “Let’s say your senior management wants you to research a package to integrate in the existing project, how do you go on with researching and finding the right package for your project?
6. Have you run on the bug which has made you feel helpless for days or even a week? Follow up: How did you tackle the anxiety that comes up with it?
7. One of the complex features you have developed in your career? What were the learnings from that feature?

#### Projects mentioned in CV

Ask some questions on other projects they have mentioned in the CV

1. What is one of the projects that you found very difficult when you were building it? What were the difficulties?
2. Which of your projects are you proud of?
3. Is building projects on the side (other than the job) important? Why?

## Technical questions regarding stacks

Choose one or two from each stack and check how technically sound they are:

#### HTML

##### Trainee

1. What is the difference between a div and a span element in HTML?
   1. Follow-up: Can you give an example of when you would use a div element and when you would use a span element in your HTML code?
2. Can you explain the difference between a block-level element and an inline element in HTML?
   1. Follow-up: Can you give an example of how you would use a block-level element versus an inline element in your HTML code?
   2. Follow-up: Does margin-top or margin-bottom have an effect on inline elements?
3. Are you familiar with semantic HTML?
   1. Follow-up: Can you give an example of a non-semantic HTML element and how you would replace it with a semantic element?
4. What is the syntax difference between a bulleted list and numbered list?
   1. Follow-up: how to nest one type of list inside another type of list?
5. What is the use of the required attribute?

##### ASE

1. Difference between block-level-element and inline element ?
2. How to insert an image in HTML?
3. Explain the use of an iframe tag.
4. What is the difference between LocalStorage and SessionStorage Objects?
5. What is the use of the figure tag in HTML 5?
6. How do you create text on a webpage that allows you to send an email when clicked?
7. In how many ways can we position an HTML element? Or what are the permissible values of the position attribute?
8. Difference between link tag <link> and anchor tag <a>
9. What is the use of datalist tag?
10. What is an SVG tag in HTML? Give Example.
11. Is the <!DOCTYPE html> tag considered as a HTML tag?
12. Can you explain the CSS box model?
13. What does the pseudo-class **:root** refer to?
14. What are viewport height and viewport width in CSS? When shouldn’t they be used?
15. Does margin-top or margin-bottom have an effect on inline elements?
16. Describe the various CSS positioning methods (static, relative, absolute, fixed, and sticky) and their use cases.

##### Software Engineer

1. What is semantic HTML? Why is it important?
2. Describe the purpose and usage of the HTML <meta> tag.
3. What is the data- attribute in HTML5, and how can it be used?
4. What is the purpose of the <noscript> tag, and when would you use it?
5. How does the picture element work in HTML5, and how can you use it to serve different image formats based on device capabilities?
6. What is the use of an iframe tag? When is it appropriate to use it?
7. What is the canvas element in HTML5?

##### Senior Engineer

##### Common Questions

1. Are you familiar with semantic HTML?

E.g. Header, Footer, Nav, Section, Article

1. How important is semantic HTML in web development? Why?
2. Are you familiar with <b> and <strong> tags? Can you tell the difference between these two tags?  
   There is only semantic difference, <strong> is a semantic tag while <b> is not
3. How do you create buttons with links in html?
4. Are you familiar with HTML5? Tell me some new tags that were introduced in HTML5
5. When should you use <div> tag? How frequently do you use it?
6. What is the use of <table> tag?
7. What are some html tags that you use frequently?

#### CSS

##### Trainee

1. Can you explain the difference between absolute positioning and relative positioning in CSS, and when you would use each one?
   * Follow-up: Can you give an example of a situation where you would use absolute positioning instead of relative positioning?
2. Can you explain the difference between a class and an ID in HTML, and when you would use each one?
   * Follow-up: Can you give an example of when you would use a class instead of an ID in your HTML code?
3. Difference between padding, margin and border
   * Follow-up: Can the padding, margin, and border properties be set separately for different sides of an element?
4. What is the purpose of the z-index and how to use it?
   * Follow-up: In which position properties z-index works
5. What are different position properties?
6. margin : 10px 20px 30px 40px, What is the value of the margin-top?
7. Explain the difference between visibility: hidden and display: none?

##### ASE

1. What is the difference between display: none and visibility: hidden?
2. Is it possible to change an inline element into a block level element? How?
3. What happens if you open the external CSS file in a browser?
4. What is VH/VW (viewport height/ viewport width) in CSS?
5. What are Pseudo elements and Pseudo classes?
6. Does margin-top or margin-bottom have an effect on inline elements?
7. What are the different ways to hide the element using CSS?
8. What does !important mean in CSS?
9. What would be the color of text “I am awesome” for the following rules?

<ul class="shopping-list" id="awesome">

<li><span>Milk</span></li>

<li class="favorite" id=”must-buy”>

<span class=”highlight”>I am awesome</span>

</li>

</ul>

<style>

ul#awesome {

color: red;

}

li.favorite span {

color: blue;

}

.favorite{

color: green

}

</style>

1. What is the z-index in CSS?
2. What are functionsand mixins in sass?

##### Software Engineer

1. What does !important mean in CSS? Is it good practice to add !important?
2. Can you explain the concept of CSS specificity and how it determines which styles are applied when there are conflicting rules?
3. How good are you with animations and transitions? Can you give their use cases?
4. Can you describe the CSS flexbox and grid layout systems. When would you choose one over the other?
5. What is a CSS Preprocessor? What are Sass, Less, and Stylus? Why do people use them?
6. Which CSS frameworks have you used? If the candidate knows at least two frameworks, ask him which framework does s/he prefer and why?
7. If the candidate knows at least one framework, ask him if he prefers a framework or vanilla styling and why?
8. How do you handle cross-browser compatibility issues in CSS, and what techniques can you use to address them?
9. When do you prefer css frameworks like tailwind over custom css classes?
10. What do you feel you need to improve on at the moment?
11. What does pixel perfect design mean to you? How do you implement pixel perfect designs in projects?

##### Senior Engineer

1. When do you prefer css frameworks like tailwind over custom css classes?
2. What do you feel you need to improve on at the moment?
3. What does pixel perfect design mean to you? How do you implement pixel perfect designs in projects?

##### Common Questions

1. Explain the importance of CSS in web development. Why do we use it?
2. Are you familiar with responsive web design?
3. How can we achieve responsive designs with CSS?
4. Difference between padding, margin and border
5. Have you used any CSS frameworks? What is the one feature you like about it?

E.g. Bootstrap, Foundation, Tailwind CSS, Materialize (Material Designs)

1. SASS
   * Trainee
   * ASE
   * Software Engineer
   * Senior SE

* Are you familiar with SASS?
* Are there any advantages for using SASS over CSS?

#### Vanilla JS

##### Trainee

1. What is the DOM?
2. Can you explain what an event listener is in JavaScript?
3. What are callbacks?
4. Explain Hoisting in javascript
5. What is the difference between null and undefined
6. Difference between “ == “ and “ === “ operators.
7. Is javascript a statically typed or a dynamically typed language?
8. What is NaN property in JavaScript?
9. Can you tell some array methods
10. Can you tell some String manipulation function
11. What are the types of errors in javascript?

Optional

1. What is Event Loop?
2. What is JavaScript's Immediately Invoked Function Expressions?
3. What is Closures function ?

##### 

##### Interview Problems

1. What does this return?

!"";

Ans: True because "" is falsy and ! makes it true.

1. Why this function doesn't print 1?

function printOne() {

const arr = [1, 5, 0, 3];

return arr[1];

}

Ans: Because array is zero indexed and index 1 returns 5.

1. Why is it returning old string?

const capitalize = str => {

str.toUpperCase();

return str;

}

Ans: str.toUpperCase() returns a new string, so need to store its result in a variable.

1. Why this function doesn't add el?

const addEl = (arr, el) => {

arr.pop(el);

return arr;

}

Ans: pop removes the last element of array.

1. Swap values without using a third variable

const a = 2, b = 3;

Ans: [b, a] = [a, b]

1. What's the result?

3 + 3; // returns 33

3 + "3"; // returns "33"

1. Why doesn't it work?

const sum = (a. b) => a + b;

Ans: There is a dot after "a". It should be a comma.

##### ASE

1. .forEach loop vs a .map() loop :

Can you describe the main difference between a .forEach loop and a .map() loop and why you would pick one versus the other?

1. Map vs Filter vs Reduce
2. How many parameters can forEach take?
3. Define Higher Order Functions
4. How can you explain Type Coercion in JS?
5. Do you know the difference between While and do while loop
6. Difference between for in loop and for of
7. What is OOP
8. Explain the 4 pillars of OOP
9. Async / Await
   1. Difference between Sync programming and async programming ?
   2. Can you explain the concept of Async programming?
10. What's the difference between a variable that is: null, undefined or undeclared?

##### Software Engineer

1. What is an event loop ?
2. What is a web API? List of examples for web API? Is `console.log` web API or javascript feature?
3. What is an Immediately Invoked Function in JavaScript?
4. Explain “this” keyword
5. What is currying in JavaScript?
6. Explain Closures in JavaScript
7. What is memoization?
8. Why do we use Template literals. Can you tell me the syntax.

##### Senior SE

* None

**Common Questions**

1. What is hoisting?
2. Do you know about event loop and its flow?
3. What is difference between null and undefined
4. Dom manipulations using js
5. Callbacks vs promises vs async/await
6. JS conventions (clean coding)

#### ReactJS

##### Trainee

1. What is ReactJS?
2. s
3. State vs Props in React.
4. What is a react hook?
5. Can you list out some of the react hooks that you have used?
6. Can you tell the difference between useState and useEffect hook?
7. Can you tell me about the dependency array in the useEffect hook?
8. How can you identify hooks in ReactJS?
9. What are components in ReactJS?What are the types of components present in ReactJS?
10. Can you list some differences between class-based components and functional components?
11. Can you tell me about stateful and stateless components in React?
12. Can you tell us about props drilling in ReactJS?
13. Can you tell us about contextAPI in React?
14. Can you tell us about virtual DOM in React?

##### ASE

1. What is the main difference between props and state?
2. Why should we not update the state directly?
3. What is context?
4. What are React events?
5. What is JSX?
6. What is Virtual DOM?

##### Software Engineer

1. Why is there a need for using keys in Lists?
2. Explain the lifecycle methods of components? How can we achieve a function based component?
3. Difference between useMemo and useCallback?
4. Higher order components
5. Can you tell which algorithm does ReactJS use in order to sync the Virtual DOM with the Real DOM?
6. Can you please explain the process, how react updates the real DOM using virtual DOM?

##### Senior SE

* + - None  
      …..
* Functional Vs Class Components
* What is a pure component?
* Higher order components
* Component life cycle
* React conventions (clean coding)
* Using hooks (useState, useEffect)

1. ExpressJS

* Add \*\*\*\*

1. GIT

* Have you used git in your projects? How proficient are you with it?
* Can you explain the concept of git? Why do we use it?
* What are some GIT commands you use frequently?
* Is there any difference between git and GitHub?

git is a version control system while GitHub is a hosting service for storing git repositories

* How do you download projects on GitHub to your local machine?
* When pushing to any branches: you do `git push origin master`. What is “origin” in this command?
* How many “origin” can one project have?
* Can you store one project in multiple repositories? How can you do it if so?
* Difference between using SSH and HTTPS url when cloning git projects

### Authentication and Authorization

* Can you walk me through the process you used for authenticating users in your app?
* What is authorization?

### SQL Vs No SQL

* Difference
* Advantages of one over the other
* How do you know which one is needed in a project?

1. Linting
   * Trainee
   * ASE
   * Software Engineer
   * Senior SE

* How do you maintain code quality in the projects?
* What is the use of linting in the project?

### Ruby

* Is Ruby suitable for aspiring developers who have just started their programming journey?
* What is the main philosophy of Ruby?
* Have you heard of "Almost Everything is an object in Ruby"? Can you give some examples?
* If 'almost everything is an object in Ruby', what is not an object?
* Have you used a module in Ruby? Can you describe it?
* Difference between module and class
* Including vs extending a module
* Can you tell me something about blocks in ruby?
* Block, Proc and Lambdas
* Difference between private and protected methods? When to use which?

### Rails

* Advantages of Rails
* MVC Pattern
* Explain what happens when Client/Browser makes the Rails API call [Modify this question after research, we want to know if they know the rails ecosystem and what happens behind the scene] -> Client makes api call which is routed to controller, controller fetches data from Model and passes data to View. View is parsed and HTML is served to the client.
* Have you heard about Rails' principle ‘convention over configuration’? Can you explain what it means? Is it important? Why?
* Difference between methods include and includes
* What do you add in the lib folder?
* Have you implemented authentication in Rails? Give an example on how you did it?
* If you have very limited time and Client wants you to implement an admin dashboard, how will you do it? -> use gem to add the dashboard instead of building it by themselves
* Let's assume you have an app that stores all your bills. At the start of the new month at 10AM you want to receive an email reminder to pay all those bills, how will you implement this with Rails?
* Can you give me a few examples of naming conventions used in Rails?
* Difference between collection and member in Routing
* How do you add custom validations to the rails model? Which method do you use for it?
* Have you used Rubocop in Rails projects? What does it do?
* Is testing important in projects? Why or why not?
* <https://career.guru99.com/top-34-ruby-on-rail-interview-questions/> || <https://github.com/afeld/rails_interview_questions> || <https://www.upwork.com/i/interview-questions/ruby-on-rails/> || <https://www.indeed.com/career-advice/interviewing/ruby-on-rails-interview-questions> || <https://arc.dev/interview/ruby-on-rails-interview-questions-and-answers> || <https://mindmajix.com/ruby-rails-interview-questions> || <https://medium.com/ruby-daily/53-ruby-on-rails-interview-questions-and-answers-eb99eed1aeb7> || <https://blog.imocha.io/ruby-on-rails-interview-questions-for-experienced> || <https://www.toptal.com/ruby-on-rails/interview-questions> || <https://rubygarage.org/blog/how-to-interview-your-ruby-on-rails-developer> || <https://dev.to/devinterview/ruby-interview-questions-and-answers-for-backend-developers-96d> || <https://www.bestinterviewquestion.com/ruby-on-rails-interview-questions>

### Backend

1. Explain the purpose of the backend?
2. How would you implement authentication and authorization on a new project?
3. Typical workflow for implementing a new feature on the backend?
4. DRY and DIE principles?

Don't Repeat Yourself and Duplication Is Evil

1. Difference between a PUT and a PATCH request?
2. When should you use asynchronous programming?
3. What are high-order functions?
4. How would you design an API?
5. How do you handle errors when making API calls?
6. What is the difference between a relational and a non-relational database?
7. Are NULL values in a database the same as that of blank space or zero?
8. Normalization and Denormalization

#### Node JS

##### Trainee

1. What is Node.js?
2. How many types of API functions are available in Node.js?
3. List some modules used in Node js
4. How do you use this module in node js
5. What is typically the first argument passed to a Node.js callback handler?  
   - Error object

**OR**

1. Have you heard about error-first callback in Node.js?  
   function(err, data) {  
    if(err){  
    console.error(error)  
    // do something  
    }  
    // do something with the data object  
   }
2. Why is Node.js Single-threaded?
3. Timing features of Node.js: setTimeout, setImmediate, setInterval  
   **Output**?  
    console.log("first");

setTimeout(() => {

console.log("second");

}, 0);

console.log("third");

**What about this?**

console.log("first");

setImmediate(() => {

console.log("second");

});

console.log("third");

1. Why Node.js over other backend techs like PHP and JAVA?
2. Which database is popular with Node.js?
3. Common libraries in Node js?
4. Why to use module.exports?
5. How does Node.js support Cryptography?
6. What is an asynchronous API?

##### ASE (In addition to general Trainee Questions)

1. What are middlewares?
2. Do you know about Callback hell?
3. What are the advantages of Promises over callbacks?
4. What do you know about Event loops?
5. If Node.js is single threaded then how does it handle concurrency?
6. How does Node.js overcome the problem of blocking of I/O operations?
7. Why should you separate Express app and server?
8. Can you access DOM in node?

### UI UX

Ask some questions on other projects they have mentioned in the CV

1. What is one of the projects that you found very difficult when you were building it? What were the difficulties?
2. Which of your projects are you proud of?
3. Is building projects on the side (other than the job) important? Why?
4. Can you explain how you start your design project and the process that you use?

Technical questions regarding designing

#### Designing

1. What is a design system?
2. What is the difference between a vector and raster image?
3. What is UI/UX?
4. What are the basic principles of color theory, and how do they apply to design?
5. What is typography, and how do you choose the appropriate fonts for a design?
6. What is white space, and how do you use it effectively in design?
7. What is a moodboard, and how do you use it in the design process?
8. What is the importance of visual hierarchy in design, and how do you achieve it?
9. What is the difference between serif and sans-serif fonts, and when would you use each in a design?

#### Figma

1. How much experience do you have in Figma?
2. What are frames in Figma, and how do you use them to organize your designs?
3. What is the difference between frames and group in figma?
4. What is prototyping in Figma?
5. Can we prototype using groups in figma? Why not?
6. What are components in Figma, and how do you create and use them effectively?
7. What happens when I change child component? Will it reflect back on the parent component?
8. What is auto layout in Figma?
9. What is the difference between HIFI and LOFI design?
10. What is a style guide? Is it necessary?
11. What is grid and why it is used?
12. What is gutter and margin?
13. How do we use grid in different screen sizes?
14. Will the parent component change if i make changes in child component?
15. How do you collaborate with other team members in Figma?
16. How do you ensure that a design is accessible and meets the needs of all users, including those with disabilities? Or what is web accessibility?

## Stage 3

Project Leadership, Personality, Career Plans and Goals, Culture Fit

## Project Leadership (For Mid-level and Seniors Devs Only)

1. How do you share your knowledge or learnings with your team members in the company? *-> Checks eagerness of sharing knowledge and helping others*
2. How do you estimate hours when you have a research task for e.g. searching for a new package to integrate in the project? What do you do if you can’t complete these type of tasks on time?

## Personality

1. How do you describe your personality?
2. What type of people do you go along with best and worst? -> Must Ask
3. What does deadline mean to you?
4. Have you ever failed to meet a deadline? If you have, what action did you take? -> Ask
5. How did your last performance appraisal go? What could you have done better?
6. What do you like to do outside of work?
7. Given the chance. What would you do differently in your career?
8. Do you plan your work day? How do you plan your tasks and priorities? Do you do things randomly or write down the tasks somewhere and complete them on priority?
9. Do you identify yourself as a hardworking individual? Why?

## Career Plans and Goals

1. What are your future goals in your professional career? *-> Checks their eagerness to grow and move beyond their comfort zone to be best in their craft*
2. What are your plans for further study and going abroad? *-> Helps us in planning well inside the company if we know their plans to match that with company visions and goals in the long term.*

## Culture Fit

These questions define who we are as a company.

1. What do you want to achieve here if you are hired in our company? What are your objectives to join us? *-> Very few people are able to answer this, helps us how eager they are in making an impact in their new role*
2. What would you say is a reasonable time to spend in one job before moving on?
3. What’s the hardest thing you've ever worked on?
4. How many hours were you working a week?
5. Why did you work so hard? Why did you care?
6. When were you the most unmotivated in your life?
7. What’s the thing you’re the most proud of?
8. Do you think everything you did in the last job was worth it? When you think about your career goals, do you prioritize the company or yourself?

## Stage 4

## Conclusion

1. Any particular reason for searching for a new job?
2. How did you find out about our company?
3. Why do you want to leave your current job?
4. What do you like/dislike most about your current job?
5. Why do you want to join this company?
6. If you are hired, what value can you add to our company?

**Do you have any questions for us?**

## After Interview

1. Intern

* Give them task to see if they can be taken and taught

1. Trainee (<1 years experience) || Has done intern but not professional yet

* Give them a test task, hire or dismiss depending on the result

1. Has experience, Junior or Middle Level Developer (1-3 years experience)

* Give them task and pay them fixed money to solve those tasks, hire or dismiss depending on the result

1. Senior Level Developer (Someone who has great portfolio, knows what they are doing OR more than 3 yrs experience)

* See what other things can be done to check other various things before going for a hire

1. Developer coming from referral

* Check if they can culturally fit, make them work with us for 6 months probation. Don’t onboard the core team. Hire or fire as required.
* See what other things can be done to check other various things before going for a hire

## References

* [121 Interview Questions with Answers](https://books.google.com.np/books?id=70vvMG_HGzkC&pg=PP10&lpg=PP10&dq=reactjs+interview+questions&source=bl&ots=5uc5Y21nfR&sig=ACfU3U1NN4_lm1yVv4qFluzbjLzfqhGxYw&hl=en&sa=X&ved=2ahUKEwjNsO7d7aXoAhXcxzgGHW0SBLs4ggEQ6AEwAnoECAoQAQ#v=onepage&q&f=false)