Full Stack Developer Interview Questions and Answers – 2025

Full Stack Development is a crucial aspect of modern web applications, involving both frontend and backend technologies to build dynamic, scalable, and high-performance applications. Skilled Full Stack Developers proficient in **HTML, CSS, JavaScript, React, Node.js, Express, MongoDB, Spring Boot, Django, and Git** are in high demand, with companies offering competitive salaries for their expertise.

Full Satck Developer Interview Questions - Beginner Level

1. What is the Web?

The Web, also known as the World Wide Web (WWW), is a collection of web pages, images, videos, and other resources connected through links. It is accessed using the internet and web browsers like Google Chrome and Mozilla Firefox. The Web allows users to visit websites, read information, watch videos, and interact with online content.

2. What is Web Development?

Web development is the process of building websites and web applications. It includes designing how a website looks, writing code to make it work, and keeping it updated so it runs smoothly. The goal is to create websites that are easy to use and work properly on different devices.

3. What is full-stack development?

Full-stack development refers to the development of both the front-end (client-side) and back-end (server-side) of a web application. A full-stack developer works with HTML, CSS, JavaScript, and frameworks for the front end, while managing databases, server logic, and APIs using languages like Node.js, Python, PHP, or Java. They handle the entire development process, ensuring seamless integration between the user interface and server-side functionality.

4. What is a website?

A website is a collection of web pages that are connected under a common domain name and can be accessed through the internet. These pages contain various types of content like text, images, videos, and links. Websites

can be static (fixed content) or dynamic (content changes based on user interaction or data). Examples of websites include blogs, e-commerce stores, social media platforms, and educational sites.

5. What is the Internet?

The Internet is a vast global network that connects millions of computers, servers, and devices worldwide. It enables communication, data sharing, and access to various online services, such as websites, emails, and cloud storage. The Internet works through a system of protocols (such as HTTP, TCP/IP) that allow devices to communicate with each other. It is the backbone of modern digital communication, powering search engines, social media, streaming services, and more.

6. What is HTML?

HTML stands for HyperText Markup Language. It is used to design web pages using a markup language. HTML is a combination of Hypertext and Markup language. Hypertext defines the link between the web pages. The markup language is used to define the text document within the tag which defines the structure of web pages. HTML is used to structure the website and is therefore used for Web Development.

7. What are the various markup languages available?

- HTML: Hypertext Markup Language
- KML: Key whole Markup Language
- MathML: Mathematical Markup Language

- **SGML: Standard Generalized Markup Language**
- **XHTML: extensible Hypertext Markup Language**
- **XML: extensible Markup Language**

8. What is the difference between HTML and HTML 5?

HTML	HTML5
It didn't support audio and video without the use of Flash player support.	It supports audio and video controls with the use of <audio> and <video> tags.
It uses cookies to store temporary data.	It uses SQL databases and application cache to store offline data.
Does not allow JavaScript to run in the browser.	Allows JavaScript to run in the background. This is possible due to JS Web worker API in HTML5.
Vector graphics are possible in HTML with the help of various technologies such as VML, Silver-light, Flash, etc.	Vector graphics is additionally an integral part of HTML5 like SVG and canvas.
It does not allow drag-and-drop effects.	It allows drag-and-drop effects and support target blank attribute.

- **9. Difference between HTML and XHTML**

HTML	XHTML
HTML stands for Hypertext Markup Language.	XHTML stands for Extensible Hypertext Markup Language.
It was developed by Tim Berners-Lee.	It was developed by W3C i.e.lowercase World Wide Web Consortium.
It was developed in 1991.	It was released in 2000.
It is extended from SGML.	It is extended from XML and HTML.
The format is a document file format.	The format is a markup language.
All tags and attributes are not necessarily to be in lower or upper case.	In this every, tag and attribute should be in lower case.

10. What is the current version of HTML?

The current version of HTML is HTML5, which is the fifth and latest version of HTML. It introduced several new elements, attributes, and behaviors, providing more powerful tools for web developers. HTML5 supports multimedia elements such as audio and video

without the need for external plugins, improved semantic markup, better handling of forms, and enhanced features for designing responsive and accessible web applications.

11. What is !DOCTYPE?

A doctype or document-type declaration is an instruction that tells the web browser about the markup language in which the current page is written. The doctype is not an element or tag, it lets the browser know about the version of or standard of HTML or any other markup language that is being used in the document. The DOCTYPE for HTML5 is case-insensitive and can be written as shown below.

```
<!DOCTYPE html>
```

18. How are comments added in HTML?

The comment tag (<!– Comment –>) is used to insert comments in the HTML code.

Types of HTML Comments: There are two types of comments in HTML which are

- **Single-line comment**
- **Multi-lines comment**

19. What are the different formats in which colors in HTML can be declared?

The color of an element can be defined in the following ways

- **Built-In Color**
- **RGB Format**
- **RGBA Format**
- **Hexadecimal Notation**
- **HSL**
- **HSLA**

- **Hue: Hue is the degree of the color wheel. Its value lies between 0 to 360 where 0 represents red, 120 represents green and 240 represents blue color.**
- **Saturation: It takes a percentage value, where 100% represents completely saturated, while 0% represents completely unsaturated (gray).**
- **Lightness: It takes a percentage value, where 100% represents white, while 0% represents black.**

20. How to create a link in HTML?

A Link is a connection from one Web resource to another. A link has two ends, An anchor and a direction. The link starts at the "source" anchor and points to the "destination" anchor, which may be any Web resource such as an image, a video clip, a sound bite, a

program, an HTML document, or an element within an HTML document.

- **HTML Link Syntax**: Links are specified in HTML using the “a” tag.

```
<a href="url">Link Text<a>
```

Explanation

- **href**: The href attribute is used to specify the destination address of the link used.
- **Text link**: The text link is the visible part of the link.

21. What is the use of the target attribute in the <link> tag?

The HTML <link> target Attribute is used to specify the window or a frame where the linked document is loaded. It is not supported by HTML 5.

Syntax

<link target="_blank|_self|_parent|_top|framename">

Attribute Values

- _blank: It opens the link in a new window.
- _self: It opens the linked document in the same frame.
- _parent: It opens the linked document in the parent frameset.
- _top: It opens the linked document in the full body of the window.
- framename: It opens the linked document in the named frame.

22. What is the use of alt attribute in images?

The <img> alt attribute is used to specify the alternate text for an image. It is useful when

the image is not displayed. It is used to give alternative information for an image.

Syntax

```
<img alt="text">
```

23. What are the HTML tags used to display a table?

- <table>: It is used to define a table.
- <tr>: It is used to define a row in a table.
- <th>: It is used to define a header cell in a table.
- <td>: It is used to define a cell in a table.
- <caption>: It is used to define the table caption.
- <colgroup>: It is used to define a group of one or more columns in a table for formatting.

- **<col>: It is used with <colgroup> element to specify column properties for each column.**
- **<tbody>: It is used to define a group of body content in a table.**
- **<thead>: It is used to group the header content in a table.**
- **<tfooter>: It is used to group the footer content in a table.**

24. What is the difference between block and inline elements?

Feature	Block Elements	Inline Elements
Display	Takes up the full width available	Takes up only the

Feature	Block Elements	Inline Elements
		necessary width
Line Break	Starts on a new line	Stays in the same line with other elements
Common Examples	, <p>, <h1> to <h6>, <section>	, <a>, ,
Usage	Used for structuring content (like paragraphs, sections)	Used for styling or linking small parts of text

25. Are and tags similar?

Both the tags (and) are used to represent the part of the web page. The tag is used as the block component, and the tag is used as an inline component.

 A Computer Science Portal for code codeforcode

- tag: The div tag is known as the Division tag. It is a block-level tag & is used in HTML to make divisions of content on the web page (text, images, header, footer, navigation bar, etc). Div tag has both openings ( ) and closing ( ) tags, and it is mandatory to close the tag.
- tag: The HTML span element is a generic inline container for inline elements and content. It is used to group elements for styling purposes (by using the class or id attributes). A better way to

use it is when no other semantic element is available.

26. Differences between & tag?

tag	tag
The tag is a block level element.	The tag is an inline element.
It is best to attach it to a section of a web page.	It is best to attach CSS to a small section of a line on a web page.
It accepts align attribute.	It does not accept aligned attributes.
This tag should be used to wrap a section, for	This tag should be used to wrap any specific word that you

tag	tag
highlighting that section.	want to highlight on your webpage.

27. What is the difference between classes and id?

- id Attribute: The id attribute is a unique identifier that is used to specify the document. It is used by CSS and JavaScript to perform a certain task for a unique element. In CSS, the id attribute is written using the # symbol followed by id.

Syntax

<element id="id_name">In CSS Stylesheet#id_name { // CSS Property}

- class Attribute: The class attribute is used to specify one or more class names for an HTML element. The class attribute can be

used on any HTML element. The class name can be used by CSS and JavaScript to perform certain tasks for elements with the specified class name. The class name can be represented by using the “.” symbol.

Syntax

<element class="class_name>In CSS Stylesheet.class { // CSS Property} .

28. How can we create a nested webpage in HTML?

When the content of one completely different webpage is embedded into another webpage, it is called a nested webpage. The nested webpage can be created using the following 2 methods:

- **<iframe> tag:** The iframe in HTML stands for Inline Frame. The “iframe” tag defines

a rectangular region within the document in which the browser can display a separate document, including scrollbars and borders.

- **<embed> tag**: The <embed> tag in HTML is used for embedding external applications which are generally multimedia content like audio or video into an HTML document.

29. What are the tags that can be used inside the <head> tag?

The <head> element is like a container for metadata i.e. data about data and it also lies between the <html> tag and the <body> tag. Metadata is the data about the HTML document and is not displayed on the webpage. It defines the document title, style, script, and other meta information.

The HTML <head> element is a container for the following elements

- **<title>: It defines the title of the webpage.**
- **<link>: It is most often used to link an external CSS file.**
- **<meta>: It is used to specify the Character set, Page description, Keywords, Author of the document, and Viewport settings. It will not be displayed but is used by browsers on how to display content or reload pages and by search engines, and other web services.**
- **<base>: It is used to specify the base URL or target for relative URLs.**
- **<style>: It is used to make internal CSS within our HTML webpage.**
- **<script>: It is used to define within the HTML webpage.**

30. What are meta tags? How are they important?

The <meta> tag in HTML provides information about HTML Document or in simple words, it provides important information about a document. These tags are basically used to add name/value pairs to describe properties of HTML documents, such as expiry date, author name, list of keywords, document author, etc.

Syntax

<meta attribute-name="value">

Key Points

- The <meta> tag contents are not visible on your browser & is added inside the <head> tag.
- They are just used to give additional information about the HTML document.

- The <meta> tags are added to our HTML document for the purpose of Search Engine Optimisation.

31. What are HTML entities?

HTML entities are used to display reserved characters or symbols that either have special meanings in HTML or aren't available on a basic keyboard. For example, < is reserved for HTML tags, but to display it on a webpage, you use the entity <. Similarly, characters like £, ¥, €, and © can be represented using their entity names (e.g., ©) or entity numbers (e.g., ©). Entities help avoid ambiguity in HTML code.

32. How can we add symbols in HTML?

There are some characters in HTML that are reserved, & have special meaning when they

are used in an HTML document. Like if you used less than or greater than sign in your HTML document then the browser will treat them differently. So we will use HTML entities to insert symbols in a webpage.

Special Symbols	Syntax
©:copyright	©
®:registered trademark	®
™:trade mark	™
@: at	@
¶: paragraph	¶
§: section	§
ℂ: double-struck capital c	𝕔

Special Symbols	Syntax
%: care of	℅

33. What is HTML Canvas?

The HTML “canvas” element is used to draw graphics via JavaScript.The “canvas” element is only a container for graphics. One must use JavaScript to actually draw the graphics. Canvas has several methods for drawing paths, boxes, circles, text, and adding images.

Example: The canvas would be a rectangular area on an HTML page. By default, a canvas has no border and no content. An id attribute has been specified to refer to it in a script, and a width and height attribute to define the size of the canvas. The style attribute is used to add a border.

<html>

```
<body>
    <canvas id="myCanvas"
            width="400"
            height="200"
            style="border:2px solid #000000;">
    </canvas>
</body>
</html>
```

34. What is CSS?

Cascading Style Sheets fondly referred to as CSS, is a simply designed language intended to simplify the process of making web pages presentable. CSS allows you to apply styles to web pages. More importantly, CSS enables you to do this independent of the HTML that makes up each web page. CSS is

easy to learn and understood, but it provides powerful control over the presentation of an HTML document.

35. Why do we use CSS?

We use CSS because of the following reasons

- CSS saves time: You can write CSS once and reuse the same sheet on multiple HTML pages.
- Easy Maintenance: To make a global change simply change the style, and all elements in all the webpages will be updated automatically.
- Search Engines: CSS is considered a clean coding technique, which means search engines won't have to struggle to "read" its content.
- Superior styles to HTML: CSS has a much wider array of attributes than HTML, so

you can give a far better look to your HTML page in comparison to HTML attributes.

. Offline Browsing: CSS can store web applications locally with the help of an offline cache. Using of this we can view offline websites.

36. What is the current version of CSS?

CSS3 is the latest version of Cascading Style Sheets (CSS), introducing new features for better web design. It follows a modular approach, allowing independent use of features. Key enhancements include flexbox and grid layouts for responsive design, transitions, animations, and transformations for smooth effects, and custom fonts, gradients, and shadows for improved styling.

These features make modern web pages more dynamic, interactive, and visually appealing.

37. How is CSS different from CSS 3?

CSS	CSS3
CSS is capable of positioning texts and objects. CSS is somehow backward compatible with CSS3.	On the other hand, CSS3 is capable of making the web page more attractive and takes less time to create. If you write CSS3 code in CSS, it will be invalid.
Responsive designing is not supported in CSS	CSS3 is the latest version, hence it supports responsive design.

CSS	CSS3
CSS cannot be split into modules.	Whereas, whereas CSS3 can be breakdown into modules.
Using CSS, we cannot build 3D animation and transformation.	But in CSS3 we can perform all kinds of animation and transformations as it supports animation and 3D transformations.
CSS is very slow as compared to CSS3	Whereas, CSS3 is faster than CSS.

38. What is the syntax for CSS?

A CSS style rule consists of a selector, property, and its value. The selector points to the HTML element where CSS style is to be applied. The CSS property is separated by semicolons.

Syntax

```
selector {
    Property: value;
}
```

39. In how many ways can we add CSS to our HTML file?

Cascading Style Sheet(CSS) is used to set the style in web pages that contain HTML elements. It sets the background color, font size, font family, color, … etc properties of elements on a web page.
There are three types of CSS which are given below:

- **Inline CSS: Inline CSS contains the CSS property in the body section attached with the element known as inline CSS. This kind of style is specified within an HTML tag using the style attribute.**
- **Internal or Embedded CSS: This can be used when a single HTML document must be styled uniquely. The CSS ruleset should be within the HTML file in the head section i.e the CSS is embedded within the HTML file.**
- **External CSS: External CSS contains a separate CSS file which contains only style property with the help of tag attributes (For example class, id, heading, … etc). CSS property is written in a separate file with .css extension and should be linked to the HTML document using the link tag. This means that for each element, style can be**

set only once and that will be applied across web pages.

40. How can we add comments in CSS?

Comments are the statements in your code that are ignored by the compiler and are not executed. Comments are used to explain the code. They make the program more readable and understandable.

Syntax

/* content */

Comments can be single-line or multi-line.

41. What does the ‘a’ in rgba mean?

RGBA contains A (Alpha) which specifies the transparency of elements. The value of alpha lies between 0.0 to 1.0 where 0.0. represents fully transparent and 1.0 represents not

transparent.

Syntax

```
h1 {
    color:rgba(R, G, B, A);
}
```

42. What are CSS HSL Colors?

HSL: HSL stands for Hue, Saturation, and Lightness respectively. This format uses the cylindrical coordinate system.

- **Hue:** Hue is the degree of the color wheel. Its value lies between 0 to 360 where 0 represents red, 120 represents green and 240 represents a blue color.
- **Saturation:** It takes a percentage value, where 100% represents completely saturated, while 0% represents completely unsaturated (gray).

- Lightness: It takes a percentage value, where 100% represents white, while 0% represents black.

Syntax

```
h1 {
   color:hsl(H, S, L);
}
```

Example

```
<html>
   <head>
      <title>CSS hsl color property</title>
      <style>
         h1{
            color:hsl(120, 100%, 30%);
            text-align:center;
         }
      </style>
```

```
    </head>
    <body>
        <h1>
            codeforcode
        </h1>
    </body>
</html>
```

43. What are the different CSS border properties?

CSS border properties allow us to set the style, color, and width of the border.

- Border Style: The border-style property specifies the type of border. None of the other border properties will work without setting the border style.
- Border Width: Border width sets the width of the border. The width of the

border can be in px, pt, cm or thin, medium and thick.

- **Border Color: This property is used to set the color of the border. Color can be set using the color name, hex value, or RGB value. If the color is not specified border inherits the color of the element itself.**

44. What are Data Types in JavaScript?

JavaScript data types are categorized into two parts i.e. primitive and non-primitive types.

- **Primitive Data Type: The predefined data types provided by JavaScript language are known as primitive data type. Primitive data types are also known as in-built data types.**
  - **Numbers**
  - **Strings**

    - Boolean
    - Symbol
    - Undefined
    - Null
    - BigInt
- **Non-Premitive Data Type: The data types that are derived from primitive data types are known as non-primitive data types. It is also known as derived data types or reference data types.**
    - Objects
    - Functions
    - Arrays

45. Which symbol is used for comments in JavaScript?

Comments prevent the execution of statements. Comments are ignored while the

compiler executes the code. There are two type of symbols to represent comments in JavaScript:

- **Double slash: It is known as a single-line comment.**

// Single line comment

- **Slash with Asterisk: It is known as a multi-line comment.**

/*

Multi-line comments

. . .

***/**

46. What would be the result of 3+2+”7”?

Here, 3 and 2 behave like an integer, and “7” behaves like a string. So 3 plus 2 will be 5. Then the output will be 5+”7” = 57.

47. What is the use of the isNaN function?

The number isNan function determines whether the passed value is NaN (Not a number) and is of the type “Number”. In JavaScript, the value NaN is considered a type of number. It returns true if the argument is not a number, else it returns false.

48. Which is faster in JavaScript and ASP script?

JavaScript is faster compared to ASP Script. JavaScript is a client-side scripting language and does not depend on the server to execute. The ASP script is a server-side scripting language always dependable on the server.

49. What is negative infinity?

The negative infinity is a constant value represents the lowest available value. It means that no other number is lesser than this value. It can be generate using a self-made function or by an arithmetic operation. JavaScript shows the NEGATIVE_INFINITY value as -Infinity.

50. Is it possible to break JavaScript Code into several lines?

Yes, it is possible to break the JavaScript code into several lines in a string statement. It can be broken by using the '\n' (backslash n).

For example

```
console.log("A Online Computer Science Portal\n for Geeks")
```

The code-breaking line is avoid by JavaScript which is not preferable.

```
let gfg= 10, GFG = 5,
```

Geeks =

gfg + GFG;

51. Which company developed JavaScript?

Netscape developed JavaScript and was created by Brenden Eich in the year of 1995.

52. What are undeclared and undefined variables?

- **Undefined: It occurs when a variable is declare but not assign any value. Undefined is not a keyword.**
- **Undeclared: It occurs when we try to access any variable which is not initialize or declare earlier using the var or const keyword. If we use 'typeof' operator to get the value of an undeclare variable, we will face the runtime error with the return**

value as “undefined”. The scope of the undeclare variables is always global.

53. Write a JavaScript code for adding new elements dynamically.

```
<html>
<head>
</head>
<body>
   <button onclick="create()">
      Click Here!
   </button>

   <script>
      function create() {
         let geeks = document.createElement('geeks');
         geeks.textContent = "Geeksforgeeks";
```

```
        geeks.setAttribute('class', 'note');
        document.body.appendChild(geeks);
    }
  </script>
</body>
</html>
```

54. What are global variables? How are these variables declared, and what are the problems associated with them?

In contrast, global variables are the variables that define outside of functions. These variables have a global scope, so they can be used by any function without passing them to the function as parameters.

Example:

```
let petName = "Kaalingas"; // Global Variable
myFunction();
```

```
function myFunction() {
    console.log("Inside myFunction - Type of petName:", typeof petName);
    console.log("Inside myFunction - petName:", petName);
}

console.log("Outside myFunction - Type of petName:", typeof petName);
console.log("Outside myFunction - petName:", petName);
```

Output

```
Inside myFunction - Type of petName: string
Inside myFunction - petName: Kaalingas
Outside myFunction - Type of petName: string
```

Outside myFunction - petName: Kaalingas

It is difficult to debug and test the code that relies on global variables.

55. What do you mean by NULL in JavaScript?

The NULL value represents that no value or no object. It is known as empty value/object.

56. How to delete property-specific values?

The delete keyword deletes the whole property and all the values at once like

```
let gfg={Course: "DSA", Duration:30};

delete gfg.Course;
```

57. What is a prompt box?

The prompt box is a dialog box with an optional message prompting the user to input

some text. It is often used if the user wants to input a value before entering a page. It returns a string containing the text entered by the user, or null.

58. What is the ‘this’ keyword in JavaScript?

Functions in JavaScript are essential objects. Like objects, it can be assign to variables, pass to other functions, and return from functions. And much like objects, they have their own properties. ‘this’ stores the current execution context of the JavaScript program. Thus, when it use inside a function, the value of ‘this’ will change depending on how the function is defined, how it is invoked, and the default execution context.

59. Explain the working of timers in JavaScript. Also explain the drawbacks of using the timer, if any.

The timer executes some specific code at a specific time or any small amount of code in repetition to do that you need to use the functions setTimout, setInterval, and clearInterval. If the JavaScript code sets the timer to 2 minutes and when the times are up then the page displays an alert message "times up". The setTimeout() method calls a function or evaluates an expression after a specified number of milliseconds.

60. What is ReactJS?

ReactJS is a JavaScript library used to build reusable components for the view layer in MVC architecture. It is highly efficient and uses a virtual DOM to render components. It works on the client side and is written in JSX.

Important Features of React

- **Virtual DOM: React uses a virtual DOM to efficiently update and render components, ensuring fast performance by minimizing direct DOM manipulations.**
- **Component-Based Architecture: React builds UI using reusable, isolated components, making code more modular, maintainable, and scalable.**
- **Hooks: React hooks allow functional components to manage state and side effects, making them powerful and more flexible.**

61. Explain the MVC architecture.

The Model-View-Controller (MVC) framework is an architectural/design pattern that separates an application into three main logical components Model, View, and Controller. Each architectural component

is built to handle specific development aspects of an application. It isolates the business, logic, and presentation layer from each other

62. Explain the building blocks of React.

The five main building blocks of React are

- Components: These are reusable blocks of code that return HTML.
- JSX: It stands for JavaScript and XML and allows you to write HTML in React.
- Props and State: props are like function parameters and State is similar to variables.
- Context: This allows data to be passed through components as props in a hierarchy.

- **Virtual DOM: It is a lightweight copy of the actual DOM which makes DOM manipulation easier.**
-

63. What is virtual DOM in React?

The Virtual DOM in React is an in-memory representation of the actual DOM. It helps React efficiently update and render the user interface by comparing the current and previous virtual DOM states using a process called diffing.

How Virtual DOM Works

- **Efficient Rendering: The Virtual DOM is an in-memory representation of the actual DOM that React uses to optimize the process of updating and rendering UI changes.**
- **Diffing Algorithm: React compares the current and previous versions of the**

Virtual DOM using a diffing algorithm, identifying the minimal set of changes required to update the real DOM.

- Batch Updates: Instead of updating the real DOM immediately, React batches multiple changes to reduce unnecessary re-renders, improving performance.
- Faster Updates: Since updating the real DOM is slow, React minimizes direct DOM manipulations by only making updates where necessary after comparing the Virtual DOM.
- Declarative UI: With the Virtual DOM, React allows developers to write code in a declarative style, letting React handle when and how to efficiently update the UI.

64. What is JSX?

JSX is basically a syntax extension of regular JavaScript and is used to create React elements. These elements are then rendered to the React DOM. All the React components are written in JSX. To embed any JavaScript expression in a piece of code written in JSX we will have to wrap that expression in curly braces {}.

Example of JSX: The name written in curly braces { } signifies JSX

```
const name = "Learner";


const element = (
    <h1>
        Hello,
        {name}.Welcome to GeeksforGeeks.
    </h1>
);
```

65. What are components and their type in React?

A Component is one of the core building blocks of React. In other words, we can say that every application you will develop in React will be made up of pieces called components. Components make the task of building UIs much easier.

In React, we mainly have two types of components

- Functional Components: Functional components are simply javascript functions. We can create a functional component in React by writing a javascript function.
- Class Components: The class components are a little more complex than the functional components. The functional components are not aware of the other

components in your program whereas the class components can work with each other. We can pass data from one class component to another class component.

.

66. How do browsers read JSX?

In general, browsers are not capable of reading JSX and only can read pure JavaScript. The web browsers read JSX with the help of a transpiler. Transpilers are used to convert JSX into JavaScript. The transpiler used is called Babel.

67. Explain the steps to create a react application and print Hello World?

To install React, first, make sure Node is installed on your computer. After installing Node. Open the terminal and type the following command.

npx create-react-app <<Application_Name>>

Navigate to the folder.

cd <<Application_Name>>

This is the first code of ReactJS Hello World!

```
import React from "react";
import "./App.css";
function App() {
    return <div className="App">Hello World !</div>;
}
export default App;
```

Type the following command to run the application

npm start

68. How to create an event in React?

To create an event in React, attach an event handler like onClick, onChange, etc., to a JSX element. Define the handler function to specify the action when the event is triggered, such as updating state or executing logic.

```
function Component() {
   doSomething(e);
   {
      e.preventDefault();
      // Some more response to the event
   }
   return <button onEvent={doSomething}></button>;
}
```

69. Explain the creation of a List in react?

Lists are very useful when it comes to developing the UI of any website. Lists are

mainly used for displaying menus on a website, for example, the navbar menu. To create a list in React use the map method of array as follows.

```
import React from "react";
import ReactDOM from "react-dom";

const numbers = [1, 2, 3, 4, 5];

const updatedNums = numbers.map((number) => {
   return <li>{number}</li>;
});

ReactDOM.render(<ul>{updatedNums}</ul>, document.getElementById("root"));
```

70. What is a key in React?

A “key” is a special string attribute you need to include when creating lists of elements in React. Keys are used in React to identify which items in the list are changed, updated, or deleted. In other words, we can say that keys are used to give an identity to the elements in the lists.

71. What is MongoDB, and How Does It Differ from Traditional SQL Databases?

- MongoDB is a NoSQL database which means it does not use the traditional table-based relational database structure. Instead of it uses a flexible and document-oriented data model that stores data in BSON (Binary JSON) format.
- Unlike SQL databases that use rows and columns, MongoDB stores data as JSON-

like documents, making it easier to handle unstructured data and providing greater flexibility in terms of schema design.

72. Explain BSON and Its Significance in MongoDB.

BSON (Binary JSON) is a binary-encoded serialization format used by MongoDB to store documents. BSON extends JSON by adding support for data types such as dates and binary data and it is designed to be efficient in both storage space and scan speed. The binary format allows MongoDB to be more efficient with data retrieval and storage compared to text-based JSON.

73. What is Express.Js?

Express is a small framework that sits on top of Node.js's web server functionality to simplify its APIs and add helpful new features.

It makes it easier to organize your application's functionality with middleware and routing; it adds helpful utilities to Node.js's HTTP objects; it facilitates the rendering of dynamic HTTP objects.

Express is a part of MEAN stack, a full stack JavaScript solution used in building fast, robust, and maintainable production web applications.

74. Why use Express.Js?

Express.js is a lightweight Node.js framework that gives us ability to create server-side web applications faster and smarter. The main reason for choosing Express is its simplicity, minimalism, flexibility, and scalability characteristics. It provides easy setup for middlewares and routing.

75. What is Spring Boot?

Spring Boot is built on top of the Spring framework to create stand-alone RESTful web applications with very minimal configuration and there is no need of external servers to run the application because it has embedded servers like Tomcat and Jetty etc.

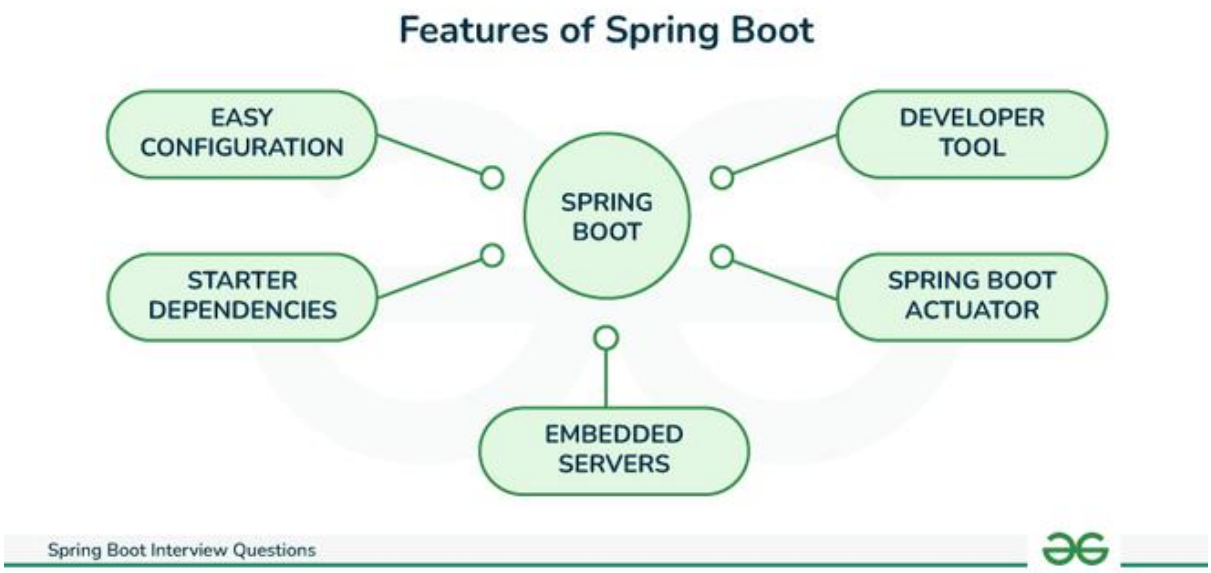
- Spring Boot framework is independent.
- It creates executable spring applications that are production-grade.

76. What are the Features of Spring Boot?

There are many useful features of Spring Boot. Some of them are mentioned below

- Auto-configuration – Spring Boot automatically configures dependencies by using @EnableAutoconfiguration annotation and reduces boilerplate code.

- **Spring Boot Starter POM – These Starter POMs are pre-configured dependencies for functions like database, security, maven configuration etc.**
- **Spring Boot CLI (Command Line Interface) – This command line tool is generally for managing dependencies, creating projects and running the applications.**
- **Actuator – Spring Boot Actuator provides health check, metrics and monitors the endpoints of the application. It also simplifies the troubleshooting management.**
- **Embedded Servers – Spring Boot contains embedded servers like Tomcat and Jetty for quick application run. No need of external servers.**



77. What is Django?

Django is a Full-stack web development framework that facilitates the creation and maintenance of high-quality Dynamic pages while also encouraging rapid development and a clean, pragmatic style. Django makes it easier to automate repeated operations, resulting in a more efficient development process with fewer lines of code.

78. What is the difference between Flask and Django?

Flask	Django
Flask is a WSGI framework	Django is a Full-stack web framework
It allows multiple types of databases.	It doesn't support multiple types of databases.
Use SQL Alchemy	Build-in ORM
Diversified Working Style	Monolithic Working Style
Arbitrary structure	Conventional Project Structure

Flask	Django
It supports API	It does not have any support for API
It does not support Dynamic HTML pages	Django accepts Dynamic Pages.
It has support for Visual debug	No support for Visual Debug
It doesn’t offer a built-in bootstrapping tool.	Django-admin enables us to start building web applications without any external input,

Flask	Django
URL dispatcher is a RESTful request.	URL dispatcher is Robust Documentation.

79. What is Git?

Git is a distributed version control system (DVCS) that is used to track changes in source code during software development. It permits multiple developers to work on a project together without interrupting each other's changes. Git is especially popular for its speed, and ability to manage both small and large projects capably.

80. What is a repository in Git?

A Git repository (or repo) is like a file structure that stores all the files for a project. It continues track changes made to these files over time, helping teams work together evenly. Git can control both local repositories (on your own machine) and remote repositories (usually hosted on platforms like GitHub, GitLab, or Bitbucket), allowing teamwork and backup.