MERN – FULL STACK INTERVIEW QUESTIONS

HTML

1) What is HTML?

HTML is short for HyperText Markup Language and is the language of the World Wide Web. It is the standard text formatting language used for creating and displaying pages on the Web. HTML documents are made up of two things: the content and the tags that format it for proper display on pages.

---

2) What are tags?

Content is placed in between HTML tags in order to properly format it. It makes use of the less than symbol (<) and the greater than symbol (>). A slash symbol is also used as a closing tag. For example:

```
<strong>sample</strong>
```

---

3) Do all HTML tags come in a pair?

No, there are single HTML tags that do not need a closing tag. Examples are the <img> tag and
 tags.

---

4) What are some of the common lists that can be used when designing a page?

You can insert any or a combination of the following list types:
– ordered list
– unordered list
– definition list
– menu list

– directory list
Each of this list types makes use of a different tag set to compose

---

5) How do you insert a comment in HTML?

Comments in HTML begins with "<!–"nd ends with "–>". For example:

---

6) Do all character entities display properly on all systems?

No, there are some character entities that cannot be displayed when the operating system that the browser is running on does not support the characters. When that happens, these characters are displayed as boxes.

---

7) What is an image map?

Image map lets you link to many different web pages using a single image. You can define shapes in images that you want to make part of an image mapping.

---

8) What is the advantage of collapsing white space?

White spaces are a blank sequence of space characters, which is treated as a single space character in HTML. Because the browser collapses multiple spaces into a single space, you can indent lines of text without worrying about multiple spaces. This enables you to organize the HTML code into a much more readable format.

---

9) Can attribute values be set to anything or are there specific values that they accept?

Some attribute values can be set to only predefined values. Other attributes can accept any numerical value that represents the number of pixels for a size.

10) How do you insert a copyright symbol on a browser page?

To insert the copyright symbol, you need to type © or & #169; in an HTML file.

---

11) How do you create links to sections within the same page?

Links can be created using the <a> tag, with referencing through the use of the number (#) symbol. For example, you can have one line as BACK TO TOP, which would result in the words "BACK TO TOP" appearing on the webpage and links to a bookmark named topmost. You then create a separate tag command like <a name="topmost"> somewhere on the top of the same webpage so that the user will be linked to that spot when he clicked on "BACK TO TOP".

---

12) Is there any way to keep list elements straight in an HTML file?

By using indents, you can keep the list elements straight. If you indent each subnested list in further than the parent list that contains it, you can at a glance determine the various lists and the elements that it contains.

---

13) If you see a web address on a magazine, to which web page does it point?

Every web page on the web can have a separate web address. Most of these addresses are relative to the top-most web page. The published web address that appears within magazines typically points this top-most page. From this top level page, you can access all other pages within the website.

---

14) What is the use of alternative text in image mapping?

When you use image maps, it can easily become confusing and difficult to determine which hotspots correspond to which links. Using alternative text lets, you put a descriptive text on each hotspot link.

---

15) Do older HTML files work on newer browsers?

Yes, older HTML files are compliant to the HTML standard. Most older files work on the newer browsers, though some features may not work.

---

16) Does a hyperlink apply to text only?

No, hyperlinks can be used in the text as well as images. That means you can convert an image into a link that will allow users to link to another page when clicked. Surround the image within the … tag combinations.

---

17) If the user's operating system does not support the needed character, how can the symbol be represented?

In cases wherein their operating system does not support a particular character, it is still possible to display that character by showing it as an image instead.

---

18) How do you change the number type in the middle of a list?

The <li> tag includes two attributes – type and value. The type attribute can be used to change the numbering type for any list item. The value attribute can change the number index.

---

19) What are style sheets?

Style sheets enable you to build consistent, transportable, and well-defined style templates. These templates can be linked to several different web pages, making it easy to maintain and change the look and feel of all the web pages within site.

---

20) State bullet types available in HTML

With ordered lists, you can select to use some different list types including alphabetical and Roman numerals. The type attribute for unordered lists can be set to disc, square, or circle.

---

21) How do you create multicolored text in a webpage?

To create text with different colors, use the … tags for every character that you want to apply color. You can use this tag combination as many times as needed, surrounding a single character or an entire word.

---

22) Why are there both numerical and named character entity values?

The numerical values are taken from the ASCII values for the various characters, but these can be difficult to remember. Because of this, named character entity values were created to make it easier for web page designers to use.

---

23) Write an HTML table tag sequence that outputs the following:
50 pcs 100 500
10 pcs 5 50

Answer:

```
<table>
<tr>
<td>50 pcs</td>
<td>100</td>
<td>500</td>
```

```
</tr>
<tr>
<td>10 pcs</td>
<td>5</td>
<td>50</td>
</tr>
</table>
```

---

24) What is the advantage of grouping several checkboxes together?

Although checkboxes don't affect one another, grouping checkboxes together help to organize them. Checkbox buttons can have their name and do not need to belong to a group. A single web page can have many different groups of checkboxes.

---

25) What will happen if you overlap sets of tags?

If two sets of HTML tags are overlapped, only the first tag will be recognized. You will find this problem when the text does not display properly on the browser screen.

---

26) What are applets?

Applets are small programs that can be embedded within web pages to perform some specific functionality, such as computations, animations, and information processing. Applets are written using the Java language.

---

27) What if there is no text between the tags or if a text was omitted by mistake? Will it affect the display of the HTML file?

If there is no text between the tags, then there is nothing to format. Therefore no formatting will appear. Some tags, especially tags without a closing tag like the <img> tag, do not require any text between them.

---

28) Is it possible to set specific colors for table borders?

You can specify a border color using style sheets, but the colors for a table that does not use style sheets will be the same as the text color.

---

29) How do you create a link that will connect to another web page when clicked?

To create hyperlinks, or links that connect to another web page, use the href tag. The general format for this is: text
Replace “site” with the actual page URL that is supposed to be linked to when the text is clicked.

---

30) What other ways can be used to align images and wrap text?

Tables can be used to position text and images. Another useful way to wrap text around an image is to use style sheets.

---

31) Can a single text link point to two different web pages?

No. The <a> tag can accept only a single href attribute, and it can point to only a single web page.

---

32) What is the difference between the directory and menu lists and the unordered list?

The key difference is that the directory and menu lists do not include attributes for changing the bullet style.

---

33) Can you change the color of bullets?

The bullet color is always the same as that of the first character in the list item. If you surround the <li> and the first character with a set of <font> tags with the color attribute set, the bullet color, and the first character will be a different color from the text.

---

34) What are the limits of the text field size?

The default size for a text field is around 13 characters. However, if you include the size attribute, you can set the size value to be as low as 1. The maximum size value will be determined by the browser width. If the size attribute is set to 0, the size will be set to the default size of 13 characters.

---

35) Do <th> tags always need to come at the start of a row or column?

Any <tr> tag can be changed to a <th> tag. This causes the text contained within the <th> tag to be displayed as bold in the browser. Although <th> tags are mainly used for headings, they do not need to be used exclusively for headings.

---

36) What is the relationship between the border and rule attributes?

Default cell borders, with a thickness of 1 pixel, are automatically added between cells if the border attribute is set to a nonzero value. Likewise, If the border attribute is not included, a default 1-pixel border appears if the rules attribute is added to the <table> tag.

---

37) What is a marquee?

A marquee allows you to put a scrolling text in a web page. To do this, place whatever text you want to appear scrolling within the <marquee> and </marquee> tags.

---

38) How do you create text on a webpage that will allow you to send an email when clicked?

To change text into a clickable link to send email, use the mailto command within the href tag. The format is as follows:

```
<A HREF="mailto:youremailaddress">text to be clicked</A>
```

---

39) Are
 tags the only way to separate sections of text?

No. The
 tag is only one way to separate lines of text. Other tags, like the <p> tag and <blockquote> tag, also separate sections of text.

---

40) Are there instances where the text will appear outside of the browser?

By default, the text is wrapped to appear within the browser window. However, if the text is part of a table cell with a defined width, the text could extend beyond the browser window.

---

41) How are active links different from normal links?

The default color for normal and active links is blue. Some browsers recognize an active link when the mouse cursor is placed over that link; others recognize active links when the link has the focus. Those that don't have a mouse cursor over that link is considered a normal link.

---

42) Do style sheets limit the number of new style definitions that can be included within the brackets?

Style sheets do not limit the number of style definitions that can be included within the brackets for a given selector. Every new style definition, however, must be separated from the others by a semicolon symbol.

---

43) Can I specify fractional weight values such as 670 or 973 for font weight?

Implementation largely depends on the browser, but the standard does not support fractional weight values. Acceptable values must end with two zeroes.

---

44) What is the hierarchy that is being followed when it comes to style sheets?

If a single selector includes three different style definitions, the definition that is closest to the actual tag takes precedence. Inline style takes priority over embedded style sheets, which takes priority over external style sheets.

---

45) Can several selectors with class names be grouped together?

You can define several selectors with the same style definition by separating them with commas. This same technique also works for selectors with class names.

---

46) What happens if you open the external CSS file in a browser?

When you try to open the external CSS file in a browser, the browser cannot open the file, because the file has a different extension. The only way to use an external CSS file is to reference it using <link/> tag within another HTML document.

---

47) How do you make a picture into a background image of a web page?

To do this, place a tag code after the </head> tag as follows:

```
<body background = “image.gif”>
```

Replace image.gif with the name of your image file. This will take the picture and make it the background image of your web page.

---

48) What happens if the list-style-type property is used on a non-list element like a paragraph?

If the list-style-type property is used on a non-list element like a paragraph, the property will be ignored and do not affect the paragraph.

---

49) When is it appropriate to use frames?

Frames can make navigating a site much easier. If the main links to the site are located in a frame that appears at the top or along the edge of the browser, the content for those links can be displayed in the remainder of the browser window.

---

50) What happens if the number of values in the rows or cols attribute doesn’t add up to 100 percent?

The browser sizes the frames relative to the total sum of the values. If the cols attribute is set to 100%, 200% the browser displays two vertical frames with the second being twice as big as the first.

---

51) Which browsers support HTML5?

The latest versions of Google Chrome, Apple Safari, Mozilla Firefox, and Opera all support most of the HTML5 features.

---

52) Name two new tags included in the HTML 5

<Video> and <Audio> are new tags which are included in HTML5 version. They are mainly used as a replacement for Flash, Silverlight, and similar technologies to play multimedia items.

---

53) Do you know which are two semantic tags are included in HTML5 version?

The <article> and <section> tags are two new tags that are included in HTML5. Articles can be composed of multiple sections that can have multiple articles. An article tag represents a full block of content which is a section of a bigger whole.

---

54) What is <figure> in HTML5?

This tag represents a piece of self-contained flow content. It is mostly used as a single unit as a reference the main flow of the document.

---

55) What is the use of Canvas element?

The canvas element helps to build charts, graphs, bypass Photoshop to create 2D images and place them directly into HTML5 code.

---

56) What are the new FORM elements which are available in HTML5?

The new Form elements in HTML5 offers much better functionality than the earlier versions.

The tags given provided to carry out these functions are:

1) <datalist> – This tag is use to specify a list of options for input controls.

2) <keygen> – This tag represents a key-pair generator field.

3) <output> – It represents the result of any scripting calculation.

---

57) Tell me two benefits of HTML5 Web Storage

Two main benefits of HTML5 Web Storage:

- It can store up to 10 MB data which is certainly more than what cookies have.
- Web storage data cannot be transferred with the HTTP request. It helps to increase the performance of the application.

---

58) What are two types of Web Storage in HTML5?

Two storage types of HTML5 are:

Session Storage: It stores data of current session only. It means that the data stored in session storage clear automatically when the browser is closed.

Local Storage: Local storage is another type of HTML5 Web Storage. In local storage, data is not deleted automatically when the current browser window is closed.

---

59) What is the Application Cache in HTML5 and why it is used?

The Application Cache concept means that a web application is cached. It can be accessible without the need for internet connection.

Some advantages of Application Cache:

1. **Offline browsing –** Web users can also use the application when they are offline.
2. **Speed –** Cached resources load quicker

3. **Reduce the server load –** The web browser will only download updated resources from the server.

---

60) Explain five new input types provided by HTML5 for forms?

Following are the important, new data types offered by HTML5:

1. **Date:** It allows the user to select a date.
2. **datetime-local:** This input type allows the user to select a date and time without time zone.
3. **datetime:** This input type allows the user to select a date and time with time zone.
4. **month:** It enables the user to select a month and year
5. **email:** These input fields used to contain an e-mail address.

----------xxxxxxx----------

CASCADING STYLE SHEETS

1. What is CSS ?

The full form of CSS is Cascading Style Sheets. It is a styling language which is simple enough for HTML elements. It is popular in web designing, and its application is common in XHTML also.

---

2. What is the origin of CSS ?

Standard Generalized Markup Language marked the beginning of style sheets in 1980s.

---

3. What are the different variations of CSS ?

The variations for CSS are:

- CSS 1
- CSS 2
- CSS 2.1
- CSS 3
- CSS 4

---

4. What are the limitations of CSS ?

Limitations are:

- Ascending by selectors is not possible
- Limitations of vertical control
- No expressions
- No column declaration
- Pseudo-class not controlled by dynamic behavior
- Rules, styles, targeting specific text not possible

---

5. What are the advantages of CSS ?

Advantages are:

- Bandwidth
- Site-wide consistency
- Page reformatting
- Accessibility
- Content separated from presentation

---

6. What are CSS frameworks?

It is a pre-planned libraries, which allows easier and more standards-compliant webpage styling, using CSS language.

---

7. How block elements can be centered with CSS1?

Block level elements can be centered by:

The margin-left and margin-right properties can be set to some explicit value:

```
BODY {

width: 40em;

background: fluorescent;

}

P {

width: 30em;

margin-right: auto;

margin-left: auto

}
```

In this case, the left and right margins will be each, five ems wide since they split up the ten ems left over from (40em-30em). It was unnecessary for setting up an explicit width for the BODY element; it was done here for simplicity.

---

8. Who maintains the CSS specifications?

World Wide Web Consortium maintains the CSS specifications.

---

9. In how many ways can a CSS be integrated as a web page?

CSS can be integrated in three ways:

- Inline: Style attribute can be used to have CSS applied HTML elements.
- Embedded: The Head element can have a Style element within which the code can be placed.
- Linked/ Imported: CSS can be placed in an external file and linked via link element.

---

10. What benefits and demerits do External Style Sheets have?

Benefits:

- One file can be used to control multiple documents having different styles.
- Multiple HTML elements can have many documents, which can have classes.
- To group styles in composite situations, methods as selector and grouping are used.

Demerits:

- Extra download is needed to import documents having style information.
- To render the document, the external style sheet should be loaded.
- Not practical for small style definitions.

---

11. Discuss the merits and demerits of Embedded Style Sheets?

Merits of Embedded Style Sheets:

- Multiple tag types can be created in a single document.
- Styles, in complex situations, can be applied by using Selector and Grouping methods.
- Extra download is unnecessary.

Demerits of Embedded Style Sheets:

- Multiple documents cannot be controlled.

---

12. What does CSS selector mean?

A string equivalent of HTML elements by which declarations or a set of it, is declared and is a link that can be referred for linking HTML and Style sheet is CSS selector.

---

13. Enlist the media types CSS allows?

The design and customization of documents are rendered by media. By applying media control over the external style sheets, they can be retrieved and used by loading it from the network.

---

14. Differentiate logical tags from physical tags?

- While physical tags are also referred to as presentational mark-up, logical tags are useless for appearances.
- Physical tags are newer versions while logical tags are old and concentrate on content.

---

15. Differentiate Style Sheet concept from HTML?

While HTML provides easy structure method, it lacks styling, unlike Style sheets. Moreover, style sheets have better browser capabilities and formatting options.

---

16. Describe ‘ruleset’?

Ruleset : Selectors can be attached to other selectors to be identified by ruleset.

It has two parts:

- Selector, e.g. R and
- declaration {text-indent: 11pt}

---

17. Comment on the Case-sensitivity of CSS ?

Although, there are no case-sensitivity of CSS, nevertheless font families, URL’s of images, etc is. Only when XML declarations along with XHTML DOCTYPE are being used on the page, CSS is case -sensitive.

---

18. Define Declaration block?

A catalog of directions within braces consisting of property, colon and value is called declaration block.
e.g.: [property 1: value 3]

---

19. Enlist the various fonts' attributes?

They are:

- Font-style
- Font-variant
- Font-weight
- Font-size/line-height
- Font-family
- Caption
- Icon

---

20. Why is it easy to insert a file by importing it?

Importing enables combining external sheets to be inserted in many sheets. Different files and sheets can be used to have different functions. Syntax:

@import notation, used with <Style> tag.

---

21. What is the usage of Class selector?

Selectors that are unique to a specific style, are called CLASS selectors. Declaration of style and association with HTML can be made through this. Syntax:

Classname
it can be A-Z, a-z or digits.

.top {font: 14em ;}, class selector
<Body class= “top”> this class is associated with element </body>

---

22. Differentiate Class selector from ID selector?

While an overall block is given to class selector, ID selector prefers only a single element differing from other elements. In other words, ID are uniques while classes are not. Its possible that an element has both class and ID.

---

23. Can more than one declaration be added in CSS?

Yes, it can be achieved by using a semicolon.

---

24. What is Pseudo-elements ?

Pseudo-elements are used to add special effects to some selectors. CSS in used to apply styles in HTML mark-up. In some cases when extra mark-up or styling is not possible for the document, then there is a feature available in CSS known as pseudo-elements. It will allow extra mark-up to the document without disturbing the actual document.

---

25. How to overrule underlining Hyperlinks?

Control statements and external style sheets are used to overrule underlining Hyperlinks.

E.g.:

```
B {

text-decoration: none;

}

<B href="career.html" style="text-decoration: none">link text</B>
```

26. What happens if 100% width is used along with floats all across the page?

While making the float declaration, 1 pixel is added every time it is used in the form of the border, and even more float is allowed thereafter.

---

27. Can default property value be restored through CSS? If yes, how?

In CSS, you cannot revert back to old values due to lack of default values. The property can be re- declared to get the default property.

---

28. Enlist the various Media types used?

Different media has different properties as they are case insensitive.

They are:

- Aural – for sound synthesizers and speech
- Print – gives a preview of the content when printed
- Projection- projects the CSS on projectors.
- Handheld- uses handheld devices.
- Screen- computers and laptop screens.

---

29. What is CSS Box Model and what are its elements?

This box defines design and layout of elements of CSS. The elements are:

Margin: the top most layer, the overall structure is shown
Border: the padding and content option with a border around it is shown. Background color affects the border.

Padding: Space is shown. Background colour affects the border.
Content: Actual content is shown.

---

30. What is contextual selector?

Selector used to select special occurrences of an element is called contextual selector. A space separates the individual selectors. Only the last element of the pattern is addressed in this kind of selector. For e.g.: TD P TEXT {color: blue}

---

31. Compare RGB values with Hexadecimal color codes ?

A color can be specified in two ways:

- A color is represented by 6 characters i.e. hexadecimal color coding. It is a combination of numbers and letters and is preceded by #. e.g.: g {color: #00cjfi}
- A color is represented by a mixture of red, green and blue. The value of a color can also be specified. e.g.: rgb(r,g,b): In this type the values can be in between the integers 0 and 255. rgb(r%,g%,b%): red, green and blue percentage is shown.

---

32. Define Image sprites with context to CSS ?

When a set of images is collaborated into one image, it is known as 'Image Sprites'. As the loading every image on a webpage consumes time, using image sprites lessens the time taken and gives information quickly.

CSS coding:

```
img.add { width: 60px; height: 55px; background: url (image.god) 0 0; }
```

In this case, only the part needed is used. The user can save substantial margin and time through this.

---

33. Compare Grouping and Nesting in CSS ?

Grouping: Selectors can be grouped having the same values of property and the code be reduced.
E.g. :

```
h1 {

color: blue;

}

h2 {

color: blue;

}

p {

color: blue;

}
```

It can be seen from the code that every element shares the same property. Rewriting can be avoided by writing each selector separated by a comma.

Nesting: Specifying a selector within a selector is called nesting.

```
P

{

color: red;

text-align: left;

}

.marked

{

background-color: blue;

}

.marked p

{
```

```
color: green;

}
```

---

34. How can the dimension be defined of an element ?

Dimension properties can be defined by:

- Height
- Max-height
- Max-width
- Min-height
- Min-width
- Width

---

35. Define float property of CSS?

By float property, the image can be moved to the right or the left along with the text to be wrapped around it. Elements before this property is applied do not change their properties.

---

36. How does Z index function?

Overlapping may occur while using CSS for positioning HTML elements. Z index helps in specifying the overlapping element. It is a number which can be positive or negative, the default value being zero.

---

37. What is graceful degradation?

In case the component fails, it will continue to work properly in the presence of a graceful degradation. The latest browser application is used when a webpage is designed. As it is not available to everyone, there is a basic functionality, which enables its use to a wider audience. In case the image is unavailable for viewing, text is shown with the alt tag.

---

38. What is progressive enhancement?

It’s an alternative to graceful degradation, which concentrates on the matter of the web. The functionality is same, but it provides an extra edge to users having the latest bandwidth. It has been into prominent use recently with mobile internet connections expanding their base.

---

39. How can backward compatibility be designed in CSS?

HTML sheet methods is collaborated with CSS and used accordingly.

---

40. How can the gap under the image be removed?

As images being inline elements are treated same as texts, so there is a gap left, which can be removed by:

CSS

```
img { display: block ; }
```

---

41. Why is @import only at the top?

@import is preferred only at the top, to avoid any overriding rules. Generally, ranking order is followed in most programming languages such as Java, Modula, etc. In C, the # is a prominent example of a @import being at the top.

---

42. Which among the following is more precedent: CSS properties or HTML procedures?

CSS is more precedent over HTML procedures. Browsers, which do not have CSS support, display HTML attributes.

43. What is Inline style?

The Inline style in a CSS is used to add up styling to individual HTML elements.

---

44. How comments can be added in CSS?

The comments in CSS can be added with /* and */.

---

45. Define Attribute Selector ?

It is defined by a set of elements, value and its parts.

---

46. Define property?

A style, that helps in influencing CSS. E.g. FONT. They have corresponding values or properties within them, like FONT has different style like bold, italic etc.

---

47. What is Alternate Style Sheet?

Alternate Style Sheets allows the user to select the style in which the page is displayed using the view>page style menu. Through Alternate Style Sheet, user can see a multiple version of the page on their needs and preferences.

---

48. Are quotes mandatory in URL's?

Quotes are optional in URL's, and it can be single or double.

---

49. What is at-rule?

Rule, which is applicable in the entire sheet and not partly, is known as at-rule. It is preceded by @ followed by A-Z, a-z or 0-9.

---

50. How can CSS be cascaded to mix with user's personal sheet?

Properties can be a set in recommended places and the document modified for CSS to mix with user's personal sheet.

These interview questions will also help in your viva(orals). In every industry, the use of websites and web applications is increasing day by day, and CSS is an essential part to build attractive websites. So, there is a huge demand for UI/UX and Web designers having a good knowledge of CSS with HTML.

----------xxxxxxxxx----------

BOOTSTRAP

1) Explain what is Bootstrap?

Bootstrap is a HTML, CSS, and JS framework for building the rich web applications with minimal effort. This framework emphasis more on building mobile web applications.

---

2) Explain why to choose Bootstrap for building the websites?

There are few reason why we choose Bootstrap for building websites

- **Mobile Support**: For mobile devices it provides full support in one single file rather than in separate file. It supports the responsive design including adjusting the CSS based on the different types of device, size of the screen etc. It reduces extra effort for developers.
- **Easy to Learn:** Writing application in bootstrap is easy if you know CSS and HTML

Browser Support: It supports all the popular browsers like Firefox, Opera, Safari, IE etc.

---

3) What are the key components of Bootstrap?

The key components of Bootstrap are

- **CSS :** It comes with plenty of CSS files
- **Scaffolding :** It provides a basic structure with Grid system , link styles and background
- **Layout Components :** List of layout components
- **JavaScript Plugins :** It contains many jQuery and JavaScript plugins
- **Customize :** To get your own version of framework you can customize your components

---

4) Explain what are class loaders in Bootstrap?

Class loader is a part of JRE (Java Runtime Environment) which loads Java classes into Java virtual environment. Class loaders also does the process of converting a named class into its equivalent binary form.

---

5) What are the types of layout available in Bootstrap?

In Bootstrap there are two types of Layout available

- **Fluid Layout:** Fluid layout is used when you want to create a app that is 100% wide and use up all the width of the screen

Fixed Layout: For a standard screen you will use fixed layout (940 px) option

---

6) Explain what is Bootstrap Grid System?

For creating page layout through a series of rows and columns that house your content Bootstrap Grid Sytem is used.

---

7) What are offset columns in Bootstrap?

For more specialized layouts offsets are a useful feature. For more spacing they can be used by pushing column over.

For example, .**col-xs=* classes** do not support offset but they are easily replicated using an empty cell

---

8) What is column ordering in Bootstrap?

Column ordering is one of the feature available in bootstrap and you can easily write columns in an order and show them in another one. With **.col-md-push-*** and **.col-md-pull-***

the order of the column can be easily changed.

---

9) What function you can use to wrap a page content?

To wrap a page content you can use **.container** and using that you can also center the content.

---

10) Explain what pagination in bootstrap is and how they are classified?

Pagination is the handling of an unordered list by bootstrap. To handle pagination bootstrap provides following classes

- **.pagination:** To get pagination on your page you have to add this class

- **.disabled, .active:** Customize links by .disabled for unclickable links and .active to indicate the current page
- **.pagination-lg, .pagination-sm:** Use these classes to get different size item

---

11) What is the use of Jumbotron in Bootstrap?

In bootstrap, Jumbotron is generally used for content that you want to highlight like some slogan or marketing headline etc. in other words it is used to enlarge the size of the headings and to add a margin for landing page content

To use the Jumbotron in Bootstrap

- Create a container ** ** with the class of .jumbotron

---

12) What is the difference between Bootstrap and Foundation?

Bootstrap	Foundation
– Bootstrap offers unlimited number of UI elements	– In Foundation UI element options are very limited in numbers
– Bootstraps uses pixels	– Foundation use REMs
– Bootstrap encourages to design for both desktop and mobile.	– Foundation encourages to design mobile first
– Bootstrap support LESS as its preprocessor	– Foundation support Sass and Compass as its preprocessor

---

13) In Bootstrap what are the two ways you can display the code?

In bootstrap you can display code in two ways

- **<code> tag :** If you are going to display code inline, you should use <code> tag

- **<pre> tag:** If you want to display the code as a standalone block element or it has multiple lines then you should use <pre> tag

---

14) Explain what are the steps for creating basic or vertical forms?

The steps for creating basic or vertical forms are

- Add a role ***form*** to the parent <form> element
- Wrap labels and controls in a with class ***.form-group***. To achieve optimum spacing this is needed
- Add a class of ***.form-control*** to all texturl <input> , <textarea> , and <select> elements

---

15) Explain what is Modal plugin used for in Bootstrap?

A modal is a child window that is layered over its parent window. Using a custom Jquery Plugin, Bootstrap Modal are created. To enrich user experience and to add functionality to users, modal windows are created with the help of Modal plugin.

---

16) Explain what is Bootstrap Container?

Bootstrap container is a class which is useful and creates a centred area within the page where our site content can be put within. The advantage of the bootstrap .container is that it is responsive and will place all our other HTML code.

---

17) Explain what is Bootstrap collapsing elements?

Bootstrap collapsing elements enables you to collapse any particular element without writing any JavaScript code or the accordion markup. In Bootstrap to apply collapsing elements you have to add data-toggle= “collapse” to the controller element along with a data-target or href to automatically assign control of a collapsible element. Likewise, you can use .collapse (options), .collapse (‘show’) or .collapse (‘hide’)

18) Explain what is list group in Bootstrap and what is the use of it?

List groups are components to display both simple and complex element with custom content

For example, a simple list group is created using class **.list-group** to address the list, and class .list-group-item to address individual item.

---

19) How you can add badge to list group in Bootstrap?

To add badge to list group in Bootstrap you have to simply add **** within the <li> element.

---

20) Explain what media object in Bootstrap is and what are their types?

Media objects in Bootstrap enables to put media object like image, video or audio to the left or right of the content blocks. Media element can be created using the class **.media** and the source is specified in using the class **.media-object.** Media-objects are of two types,

They are of two types

- .media
- .media-list

---

21) Explain what is Bootstrap well?

Bootstrap well is a container that makes the content to appear sunken or an inset effect on the page. In order to create a well, wrap the content that you would like to appear in the well with a containing the class of .well.

22) Explain how you can create Nav elements in Bootstrap?

Bootstrap offers various options for styling navigation elements all of them use the same markup and base class .nav.

To create Tabular Navigation or Tabs

- Start with a basic unordered list with the base class of **.nav**
- Then add class **.nav-tabs**

---

23) Explain what is the use of Bootstrap Carousel plugin?

The Carousel plugin is used to add a slider to your site. It is useful in condition where you want to display huge amount of contents within a small space on the web pages. Some of the standard carousel includes

- .carousel (options)
- .carousel ('cycle')
- .carousel ('pause')
- .carousel ('number')
- .carousel ('prev')
- .carousel ('next')

These interview questions will also help in your viva(orals)

---xxxxxxxx---

JAVASCRIPT

1. What is JavaScript?

JavaScript is a very powerful client-side scripting language. JavaScript is used mainly for enhancing the interaction of a user with the webpage. In other words, you can make your webpage more lively and interactive, with the help of JavaScript. JavaScript is also being used widely in game development and Mobile application development.

2. Enumerate the differences between Java and JavaScript?

Java is a complete programming language. In contrast, JavaScript is a coded program that can be introduced to HTML pages. These two languages are not at all inter-dependent and are designed for different intent. Java is an object-oriented programming (OOPS) or structured programming languages like C++ or C, whereas JavaScript is a client-side scripting language.

3. What are JavaScript Data Types?

Following are the JavaScript Data types:

- Number
- String
- Boolean
- Object

Undefined

4. What is the use of isNaN function?

isNan function returns true if the argument is not a number; otherwise, it is false.

5. Which is faster between JavaScript and an ASP script?

JavaScript is faster. JavaScript is a client-side language,, and thus it does not need the assistance of the webserver to execute. On the other hand, ASP is a server-side language and hence is always slower than JavaScript. Javascript now is also a server-side language (nodejs).

6. What is negative Infinity?

Negative Infinity is a number in JavaScript which can be derived by dividing negative number by zero.

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

Breaking within a string statement can be done by using a backslash, '\,' at the end of the first line.

Example:

```
document. Write ("This is \a program,");
```
And if you change to a new line when not within a string statement, then javaScript ignores the break in the line.

Example:

```
var x=1, y=2,
z=
x+y;
```
The above code is perfectly fine, though not advisable as it hampers debugging.

---

8. Which company developed JavaScript?

Netscape is the software company that developed JavaScript.

---

9. What are undeclared and undefined variables?

Undeclared variables are those that do not exist in a program and are not declared. If the program tries to read the value of an undeclared variable, then a runtime error is encountered.

Undefined variables are those that are declared in the program but have not been given any value. If the program tries to read the value of an undefined variable, an undefined value is returned.

---

10. Write the code for adding new elements dynamically?

```
<html>
<head>
<title>t1</title>
<script type="text/javascript">
    function addNode () { var newP = document. createElement("p");
    var textNode = document.createTextNode(" This is a new text node");
    newP.appendChild(textNode); document.getElementById("firstP").appendChild(newP); }
</script> </head>
<body> <p id="firstP">firstP<p> </body>
</html>
```

---

11. What are global variables? How are these variable declared?

Global variables are available throughout the length of the code so that it has no scope. The var keyword is used to declare a local variable or object. If the var keyword is omitted, a global variable is declared.

Example:

// Declare a global: globalVariable = “Test”;

The problems faced by using global variables are the clash of variable names of local and global scope. Also, it is difficult to debug and test the code that relies on global variables.

---

12. What is a prompt box?

A prompt box is a box that allows the user to enter input by providing a text box. A label and box will be provided to enter the text or number.

---

13. What is ‘this’ keyword in JavaScript?

‘This’ keyword refers to the object from where it was called.

---

14. What is the working of timers in JavaScript?

Timers are used to execute a piece of code at a set time or repeat the code in a given interval. This is done by using the functions **setTimeout, setInterval,** and **clearInterval**.

The **setTimeout(function, delay)** function is used to start a timer that calls a particular function after the mentioned delay. The **setInterval(function, delay)** function repeatedly executes the given function in the mentioned delay and only halts when canceled.
The **clearInterval(id)** function instructs the timer to stop.

Timers are operated within a single thread, and thus events might queue up, waiting to be executed.

---

15. Which symbol is used for comments in Javascript?

// for Single line comments and

/* Multi

Line

Comment

*/

---

16. What is the difference between ViewState and SessionState?

- ‘ViewState’ is specific to a page in a session.
- ‘SessionState’ is specific to user-specific data that can be accessed across all web application pages.

---

17. What is === operator?

=== is called a strict equality operator, which returns true when the two operands have the same value without conversion.

---

18. How you can submit a form using JavaScript?

To submit a form using JavaScript use

```
document.form[0].submit();
document.form[0].submit();
```

---

19. Does JavaScript support automatic type conversion?

Yes, JavaScript does support automatic type conversion. It is the common way of type conversion used by JavaScript developers

---

20. How can the style/class of an element be changed?

It can be done in the following way:

```
document.getElementById("myText"). style. fontSize = "20";
```

or

```
document. getElementById ("myText"). className = "anyclass";
```

---

21. How to read and write a file using JavaScript?

There are two ways to read and write a file using JavaScript

- Using JavaScript extensions
- Using a web page and Active X objects

---

22. What are all the looping structures in JavaScript?

Following are looping structures in Javascript:

- For

- While
- Do-while loops

23. What is called Variable typing in Javascript?

Variable typing is used to assign a number to a variable. The same variable can be assigned to a string.

Example:

```
i = 10;
i = "string;"
```

This is called variable typing.

24. How can you convert the string of any base to an integer in JavaScript?

The parseInt() function is used to convert numbers between different bases. parseInt() takes the string to be converted as its first parameter. The second parameter is the base of the given string.

To convert 4F (or base 16) to integer, the code used will be –

```
parseInt ("4F", 16);
```

25. Difference between “==” and “===”?

“==” checks only for equality in value, whereas “===” is a stricter equality test and returns false if either the value or the type of the two variables are different.

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

Since 3 and 2 are integers, they will be added numerically. And since 7 is a string, its concatenation will be done. So the result would be 57.

27. How to detect the operating system on the client machine?

In order to detect the operating system on the client machine, the navigator. Platform string (property) should be used.

28. What do you mean by NULL in Javascript?

The NULL value is used to represent no value or no object. It implies no object or null string, no valid boolean value, no number, and no array object.

29. What is the function of the delete operator?

The delete keyword is used to delete the property as well as its value.

Example

```
var student= {age:20, batch:"ABC"};
Delete student. age;
```

30. What is an undefined value in JavaScript?

Undefined value means the

- Variable used in the code doesn't exist
- Variable is not assigned to any value
- Property does not exist.

31. What are all the types of Pop up boxes available in JavaScript?

- Alert
- Confirm and
- Prompt

32. What is the use of Void (0)?

Void(0) is used to prevent the page from refreshing, and parameter "zero" is passed while calling.

Void(0) is used to call another method without refreshing the page.

33. How can a page be forced to load another page in JavaScript?

The following code has to be inserted to achieve the desired effect:

```
<script language="JavaScript" type="text/javascript" >
<!-- location. href="https://www.guru99.com/javascript-interview-questions-
answers.html"; //--></script>
```

34. What is the data type of variables in JavaScript?

All variables in JavaScript are object data types.

35. What is the difference between an alert box and a confirmation box?

An alert box displays only one button, which is the OK button.

But a Confirmation box displays two buttons, namely OK and cancel.

36. What are escape characters?

Escape characters (Backslash) is used when working with special characters like single quotes, double quotes, apostrophes, and ampersands. Place backslash before the characters to make it display.

Example:

```
document. write "I m a "good" boy."
document. write "I m a \"good\" boy."
```

37. What are JavaScript Cookies?

Cookies are the small test files stored in a computer, and they get created when the user visits the websites to store information that they need. Examples could be User Name details and shopping cart information from previous visits.

38. What a pop()method in JavaScript is?

The pop() method is similar to the shift() method, but the difference is that the Shift method works at the array's start. The pop() method takes the last element off of the given array and returns it. The array on which it is called is then altered.

Example:

```
var cloths = ["Shirt", "Pant", "TShirt"];
cloths.pop();
//Now cloth becomes Shirt,Pant
```

39. Does JavaScript has concept level scope?

No. JavaScript does not have concept-level scope. The variable declared inside the function has scope inside the function.

40. What are the disadvantages of using innerHTML in JavaScript?

If you use innerHTML in JavaScript, the disadvantage is

- Content is replaced everywhere
- We cannot use it like “appending to innerHTML
- Even if you use +=like “innerHTML = innerHTML + ‘html'” still the old content is replaced by html
- The entire innerHTML content is re-parsed and builds into elements. Therefore, it’s much slower
- The innerHTML does not provide validation, and therefore we can potentially insert valid and broken HTML in the document and break it

---

41. What is break and continue statements?

Break statement exits from the current loop.

Continue statement continues with next statement of the loop.

---

42. What are the two basic groups of data types in JavaScript?

- They are as—Primitive
- Reference types

Primitive types are number and Boolean data types. Reference types are more complex types like strings and dates.

---

43. How can generic objects be created?

Generic objects can be created as:

```
var I = new object();
```

---

44. What is the use of a type of operator?

‘Typeof’ is an operator used to return a string description of the type of a variable.

---

45. Which keywords are used to handle exceptions?

Try… Catch—finally is used to handle exceptions in the JavaScript

```
Try{
    Code
}
Catch(exp){
    Code to throw an exception.
}
Finally{
    Code runs either it finishes successfully or after catch
}
```

46. Which keyword is used to print the text on the screen?

Document. Write ("Welcome") is used to print the text–Welcome on the screen.

47. What is the use of the blur function?

Blur function is used to remove the focus from the specified object.

48. What is variable typing?

Variable typing assigns a number to a variable and then assigns a string to the same variable. An example is as follows:

```
i= 8;
i="john";
```

49. How to find an operating system in the client machine using JavaScript?

The 'Navigator. the app version is used to find the operating system's name in the client machine.

50. What are the different types of errors in JavaScript?

There are three types of errors:

- **Load time errors**: Errors that come up when loading a web page, like improper syntax errors, are known as Load time errors and generate the errors dynamically.
- **Runtime errors**: Errors that come due to misuse of the command inside the HTML language.
- **Logical errors**: These are the errors that occur due to the bad logic performed on a function with a different operation.

51. What is the use of the Push method in JavaScript?

The push method is used to add or append one or more elements to an Array end. Using this method, we can append multiple elements by passing multiple arguments.

52. What is the unshift method in JavaScript?

Unshift method is like the push method, which works at the beginning of the array. This method is used to prepend one or more elements to the beginning of the array.

53. What is the difference between JavaScript and Jscript?

Both are almost similar. Netscape and Jscript develop JavaScript was developed by Microsoft.

54. How are object properties assigned?

Properties are assigned to objects in the following way –

```
obj ["class"] = 12;
or
obj.class = 12;
```

55. What is the 'Strict Mode in JavaScript, and how can it be enabled?

Strict Mode adds certain compulsions to JavaScript. Under the strict Mode, JavaScript shows errors for a piece of code, which did not show an error before, but might be problematic and potentially unsafe. Strict Mode also solves some mistakes that hamper the JavaScript engines from working efficiently.

Strict mode can be enabled by adding the string literal "use strict" above the file. This can be illustrated by the given example:

```
function myfunction() {
    "use strict;"
    var v = "This is a strict mode function";
}
```

56. What is the way to get the status of a CheckBox?

The status can be acquired as follows –

```
 alert(document.getElementById('checkbox1').checked);
```

If the CheckBox is checked, this alert will return TRUE.

57. How can the OS of the client machine be detected?

The navigator. appVersion string can be used to detect the operating system on the client machine.

58. What is a window.onload and onDocumentReady?

The onload function is not run until all the information on the page is loaded. This leads to a substantial delay before any code is executed.

onDocumentReady loads the code just after the DOM is loaded. This allows early manipulation of the code.

59. How closures work in JavaScript?

The closure is a locally declared variable related to a function that stays in memory when it has returned.

For example:

```
function greet(message) {
    console.log(message);
}
function greeter(name, age) {

    return name + " says howdy!! He is " + age + " years old";
}
// Generate the message
var message = greeter("James", 23);
// Pass it explicitly to greet
greet(message);
This function can be better represented by using closures
function greeter(name, age) {
    var message = name + " says howdy!! He is " + age + " years old";
    return function greet() {
        console.log(message);
    };
}
// Generate the closure
var JamesGreeter = greeter("James", 23);
// Use the closure
JamesGreeter();
```

60. How can a value be appended to an array?

A value can be appended to an array in the given manner –

```
arr[arr.length] = value;
```

61. What is for-in loop in Javascript?

The for-in loop is used to loop through the properties of an object.

The syntax for the for-in loop is –

```
for (variable name in object){
    statement or block to execute
}
```

In each repetition, one property from the object is associated with the variable name. The loop is continued till all the properties of the object are depleted.

---

62. What are the important properties of an anonymous function in JavaScript?

A function that is declared without any named identifier is known as an anonymous function. In general, an anonymous function is inaccessible after its declaration.

Anonymous function declaration –

```
var anon = function() {
    alert('I am anonymous');
};
anon();
```

---

63. What is the difference between .call() and .apply()?

The function .call() and .apply() are very similar in their usage except a little difference. .call() is used when the number of the function's arguments are known to the programmer, as they have to be mentioned as arguments in the call statement. On the other hand, .apply() is used when the number is not known. The function .apply() expects the argument to be an array.

The basic difference between .call() and .apply() is in the way arguments are passed to the function. Their usage can be illustrated by the given example.

```
var someObject = {
myProperty : 'Foo',

myMethod : function(prefix, postfix) {

    alert(prefix + this.myProperty + postfix);
}
};
someObject.myMethod('<', '>'); // alerts '<Foo>'
var someOtherObject  = {

    myProperty : 'Bar.'
```

```
};
someObject.myMethod.call(someOtherObject, '<', '>'); // alerts '<Bar>'

someObject.myMethod.apply(someOtherObject, ['<', '>']); // alerts '<Bar>'
```

64. What is event bubbling?

JavaScript allows DOM elements to be nested inside each other. In such a case, if the handler of the child is clicked, the handler of the parent will also work as if it were clicked too.

65. Is JavaScript case sensitive? Give its example.

Yes, JavaScript is case-sensitive. For example, a function parseInt is not the same as the function Parseint.

66. What boolean operators can be used in JavaScript?

The ‘And’ Operator (&&), ‘Or’ Operator (||), and the ‘Not’ Operator (!) can be used in JavaScript.

*Operators are without the parenthesis.

67. How can a particular frame be targeted, from a hyperlink, in JavaScript?

This can be done by including the name of the required frame in the hyperlink using the ‘target’ attribute.

```
<a href="/newpage.htm" target="newframe">>New Page</a>
```

68. What is the role of break and continue statements?

The break statement is used to come out of the current loop. In contrast, the continue statement continues the current loop with a new recurrence.

69. Write the point of difference between a web garden and a web farm?

Both web-garden and web-farm are web hosting systems. The only difference is that web-garden is a setup that includes many processors in a single server. At the same time,web-farm is a larger setup that uses more than one server.

70. How are object properties assigned?

Assigning properties to objects is done in the same way as a value is assigned to a variable. For example, a form object's action value is assigned as 'submit' in the following manner – Document. form.action="submit"

71. What is the method for reading and writing a file in JavaScript?

This can be done by Using JavaScript extensions (runs from JavaScript Editor), for example, for the opening of a file –

```
fh = fopen(getScriptPath(), 0);
```

72. How are DOM utilized in JavaScript?

DOM stands for Document Object Model and is responsible for how various objects in a document interact with each other. DOM is required for developing web pages, which includes objects like paragraphs, links, etc. These objects can be operated to include actions like add or delete. DOM is also required to add extra capabilities to a web page. On top of that, the use of API gives an advantage over other existing models.

73. How are event handlers utilized in JavaScript?

Events are the actions that result from activities, such as clicking a link or filling a form by the user. An event handler is required to manage the proper execution of all these events. Event handlers are an extra attribute of the object. This attribute includes the event's name and the action taken if the event takes place.

74. What is the role of deferred scripts in JavaScript?

The HTML code's parsing during page loading is paused by default until the script has not stopped executing. If the server is slow or the script is particularly heavy, then the web page is delayed.

While using Deferred, scripts delays execution of the script till the time the HTML parser is running. This reduces the loading time of web pages, and they get displayed faster.

75. What are the various functional components in JavaScript?

The different functional components in JavaScript are-

- **First-class functions:** Functions in JavaScript are utilized as first-class objects. This usually means that these functions can be passed as arguments to other functions, returned

as values from other functions, assigned to variables, or can also be stored in data structures.

- **Nested functions:** The functions, which are defined inside other functions, are called Nested functions. They are called ‘every time the main function is invoked.

---

76. Write about the errors shown in JavaScript?

JavaScript gives a message as if it encounters an error. The recognized errors are –

- **Load-time errors:** The errors shown at the time of the page loading are counted under Load-time errors. The use of improper syntax encounters these errors and is thus detected while the page is getting loaded.
- **Runtime errors:** This is the error that comes up while the program is running. For example, illegal operations cause the division of a number by zero or access a non-existent area of the memory.
- **Logic errors:** It is caused by syntactically correct code, which does not fulfill the required task—for example, an infinite loop.

---

77. What are Screen objects?

Screen objects are used to read the information from the client’s screen. The properties of screen objects are –

- AvailHeight: Gives the height of the client’s screen
- AvailWidth: Gives the width of the client’s screen
- ColorDepth: Gives the bit depth of images on the client’s screen
- Height: Gives the total height of the client’s screen, including the taskbar
- Width: Gives the total width of the client’s screen, including the taskbar

---

78. What is the unshift() method?

This method is functional at the starting of the array, unlike the push(). It adds the desired number of elements to the top of an array. For example –

```
var name = [ "john" ];
name.unshift( "charlie" );
name.unshift( "joseph", "Jane" );
console.log(name);
```

The output is shown below:

```
[" joseph ","," Jane ,", " charlie ", " john "]
```

---

79. What is unescape() and escape() functions?

The escape () function is responsible for coding a string to transfer the information from one computer to the other across a network.

For Example:

```
<script>
document.write(escape("Hello? How are you!"));
</script>
```

Output: Hello%3F%20How%20are%20you%21

The unescape() function is very important as it decodes the coded string.

It works in the following way. For example:

```
<script>
    document.write(unescape("Hello%3F%20How%20are%20you%21"));
</script>
```

Output: Hello? How are you!

---

80. What are the decodeURI() and encodeURI()?

EncodeURl() is used to convert URL into their hex coding. And DecodeURI() is used to convert the encoded URL back to normal.

```
<script>
   var uri="my test.asp?name=ståle&car=saab";

   document.write(encodeURI(uri)+ "<br>");

    document.write(decodeURI(uri));
</script>
```

Output –

my%20test.asp?name=st%C3%A5le&car=saab

my test.asp?name=ståle&car=saab

---

81. Why you should not use innerHTML in JavaScript?

innerHTML content is refreshed every time and thus is slower. There is no scope for validation in innerHTML. Therefore, it is easier to insert rogue code in the document and make the web page unstable.

82. What does the following statement declare?

```
var myArray = [[[]]];
```

It declares a three-dimensional array.

83. How are JavaScript and ECMA Script related?

ECMA Script is like rules and guidelines, while Javascript is a scripting language used for web development.

84. What is namespacing in JavaScript, and how is it used?

Namespacing is used for grouping the desired functions, variables, etc., under a unique name. It is a name that has been attached to the desired functions, objects, and properties. This improves modularity in the coding and enables code reuse.

85. How can JavaScript codes be hidden from old browsers that do not support JavaScript?

For hiding JavaScript codes from old browsers:

Add “<!–” without the quotes in the code just after the <script> tag.

Add “//–>” without the quotes in the code just before the <script> tag.

Old browsers will now treat this JavaScript code as a long HTML comment. While a browser that supports JavaScript will take the “<!–” and “//–>” as one-line comments.

86. How to use Loop in JavaScript?

Loops are useful when you repeatedly execute the same lines of code a specific number of times or as long as a specific condition is true. Suppose you want to type a ‘Hello’ message 100 times on your webpage. Of course, you will have to copy and paste the same line 100 times. Instead, if you use loops, you can complete this task in just 3 or 4 lines.

87. How to use Loops in Javascript?

There are mainly four types of loops in JavaScript.

for loop

for/in a loop (explained later)

while loop

do…while loop

for loop

Syntax:

```
for(statement1; statement2; statment3)

{

lines of code to be executed

}
```

1. Statement1 is executed first, even before executing the looping code. So, this statement is normally used to assign values to variables used inside the loop.
2. The statement2 is the condition to execute the loop.
3. The statement3 is executed every time after the looping code is executed.

```
<html>
<head>
    <script type="text/javascript">
        var students = new Array("John", "Ann", "Aaron", "Edwin", "Elizabeth");
        document.write("<b>Using for loops </b><br />");
        for (i=0;i<students.length;i++)
        {
        document.write(students[i] + "<br />");
        }
    </script>
</head>
<body>
</body>
</html>
```

while loop

Syntax:

```
while(condition)

{

lines of code to be executed
```

```
}
```
The “while loop” is executed as long as the specified condition is true. Inside the while loop, you should include the statement that will end the loop at some point in time. Otherwise, your loop will never end, and your browser may crash.

do…while loop

Syntax:

```
<pre>
do

{

block of code to be executed

} while (condition)
```
The do…while loop is very similar to the while loop. The only difference is that in do…while loop, the block of code gets executed once even before checking the condition.

Example:

```
<html>
<head>
    <script type="text/javascript">
        document.write("<b>Using while loops </b><br />");
        var i = 0, j = 1, k;
        document.write("Fibonacci series less than 40<br />");
        while(i<40)
        {
            document.write(i + "<br />");
            k = i+j;
            i = j;
            j = k;
        }
    </script>
</head>
<body>
</body>
</html>
```

---

88. What are the important JavaScript Array Method explain with example?

JavaScript Array Methods

The Array object has many properties and methods which help developers to handle arrays easily and efficiently. You can get the value of a property by specifying arrayname.property and the output of a method by specifying arrayname.method().

- **length property** –> If you want to know the number of elements in an array, you can use the length property.
- **prototype property** –> If you want to add new properties and methods, you can use the prototype property.
- **reverse method** –> You can reverse the order of items in an array using a reverse method.
- **sort method –>** You can sort the items in an array using sort method.
- **pop method** –> You can remove the last item of an array using a pop method.
- **shift method** –> You can remove the first item of an array using shift method.
- **push method** –> You can add a value as the last item of the array.

```
<html>
<head>
    <title>Arrays!!!</title>
    <script type="text/javascript">
        var students = new Array("John", "Ann", "Aaron", "Edwin", "Elizabeth");
        Array.prototype.displayItems=function(){
            for (i=0;i<this.length;i++){
                document.write(this[i] + "<br />");
            }
        }
        document.write("students array<br />");
        students.displayItems();
        document.write("<br />The number of items in students array is " +
students.length + "<br />");
        document.write("<br />The SORTED students array<br />");
        students.sort();
        students.displayItems();
        document.write("<br />The REVERSED students array<br />");
        students.reverse();
        students.displayItems();
        document.write("<br />THE students array after REMOVING the LAST item<br />");
        students.pop();
        students.displayItems();
               document.write("<br />THE students array after PUSH<br />");
        students.push("New Stuff");
        students.displayItems();
    </script>
</head>
<body>
</body>
</html>
```

---

89. What is OOPS Concept in JavaScript?

Many times, variables or arrays are not sufficient to simulate real-life situations. JavaScript allows you to create objects that act like real-life objects. A student or a home can be an object that has many unique characteristics of its own. You can create properties and methods for your objects to make programming easier. If your object is a student, it will have properties like the first name, last name, id, etc., and methods like calculating rank, change address, etc. If your object is a

home, it will have properties like a number of rooms, paint color, location, etc. The methods like to calculate area, change owner, etc.

How to Create an Object

You can create an object like this:

```
var objName = new              Object();
objName.property1 = value1;
objName.property2 = value2;
objName.method1 = function()
{
line of code
}
```

OR

```
var objName= {property1:value1, property2:value2, method1: function()

{ lines of code} };
```

---

90. What is Loop Though the Properties of an Object?

The for/in a loop is usually used to loop through the properties of an object. You can give any name for the variable, but the object's name should be the same as an already existing object you need to loop through.

Syntax:

```
for (variablename in objectname)

{

lines of code to be executed

}
```

Example:

```
<html>
<head>
    <script type="text/javascript">
        var employee={first:"John", last:"Doe", department:"Accounts"};
               var             details = "";
               document.write("<b>Using for/in loops </b><br />");
        for (var x in employee)
        {
               details = x + ": " + employee[x];
               document.write(details + "<br />");
        }
    </script>
</head>
<body>
```

```
</body>
</html>
```

91. What is JavaScript Unit Testing, and what are the challenges in JavaScript Unit Testing?

JavaScript Unit Testing is a testing method in which JavaScript tests code written for a web page or web application module. It is combined with HTML as an inline event handler and executed in the browser to test if all functionalities work fine. These unit tests are then organized in the test suite.

Every suite contains several tests designed to be executed for a separate module. Most importantly, they don't conflict with any other module and run with fewer dependencies on each other (some critical situations may cause dependencies).

Challenges of JavaScript Unit Testing:

Here are important challenges of JavaScript Unit Testing:

- Many other languages support unit testing in browsers, in the stable as well as in runtime environment, but JavaScript can not
- You can understand some system actions with other languages, but this is not the case with JavaScript
- Some JavaScript are written for a web application that may have multiple dependencies.
- JavaScript is good to use in combination with HTML and CSS rather than on the web
- Difficulties with page rendering and DOM manipulation
- Sometimes you find an error message on your screen regarding 'Unable to load example.js' or any other JavaScript error regarding version control. These vulnerabilities come under Unit Testing JavaScript

Solutions of JavaScript Unit Testing:

To avoid such issues, what you can do is;

- Do not use global variables.
- Do not manipulate predefined objects.
- Design core functionalities based on the library.
- Try to create small pieces of functionalities with lesser dependencies.

92. What are some important JavaScript Unit Testing Frameworks?

Following is a curated list of popular JavaScript Unit Testing Frameworks and Tools that are widely used :

Unit.js: It is known as an open-source assertion library running on browser and Node.js. It is extremely compatible with other JavaScript Unit Testing frameworks like Mocha, Karma, Jasmine, QUnit, Protractor, etc. Provides the full documented API of assertion list.

QUnit: It is used for both client-side and server-side JavaScript Unit Testing. This Free JavaScript testing framework is used for jQuery projects. It follows Common JS unit testing Specification for unit testing in JavaScript. It supports the Node Long-term Support Schedule.

Jasmine: Jasmine is the behavior-driven development framework to unit test JavaScript. It is used for testing both synchronous and asynchronous JavaScript codes. It does not require DOM and comes with an easy syntax that can be written for any test.

Karma: Karma is an open-source productive testing environment. Easy workflow control running on the command line. Offers the freedom to write the tests with Jasmine, Mocha, and QUnit. You can run the test on real devices with easy debugging.

Mocha: Mocha runs on Node.js and in the browser. Mocha performs asynchronous testing more simply. Provides accuracy and flexibility in reporting. Provides tremendous support of rich features such as test-specific timeouts, JavaScript APIs.

Jest: Facebook uses jest so far to test all the JavaScript code. It provides the ‘zero-configuration testing experience. Supports independent and non-interrupting running tests without any conflict. Do not require any other setup configuration and libraries.

AVA: AVA is a simple JavaScript Unit Testing Framework. Tests are being run in parallel and serially. Parallel tests run without interrupting each other. This testing framework supports asynchronous testing as well. AVA uses subprocesses to run the unit test JavaScript.

---

93. What is QuickSort Algorithm in JavaScript?

Quick Sort algorithm follows Divide and Conquer approach. It divides elements into smaller parts based on some conditions and performing the sort of operations on those divided smaller parts.

Quick Sort algorithm is one of the most used and popular algorithms in any programming language. If you are a JavaScript developer, you might have heard of sort() which is already available in JavaScript. Then, you might have been thinking about what the need for this Quick Sort algorithm is. To understand this, first, we need what is sorting and what is the default sorting in JavaScript.

Quicksort follows the **Divide-and-Conquer** algorithm. It divides elements into smaller parts based on some conditions and performs the sort operations on those divided smaller parts. Hence, it works well for large datasets. So, here are the steps of how Quicksort works in simple words.

1. First, select an element that is to be called the **pivot** element.
2. Next, compare all array elements with the selected pivot element and arrange them so that elements less than the pivot element are left. Greater than pivot is to its right.
3. Finally, perform the same operations on the left and right side elements to the pivot element.

So, that is the basic outline of Quicksort. Here are the steps which need to be followed one by one to perform Quicksort.

94.How does QuickSort Work

Step 1) First, find the **"pivot"** element in the array.

Step 2) Start the left pointer at the first element of the array.

Step 3) Start the right pointer at the last element of the array.

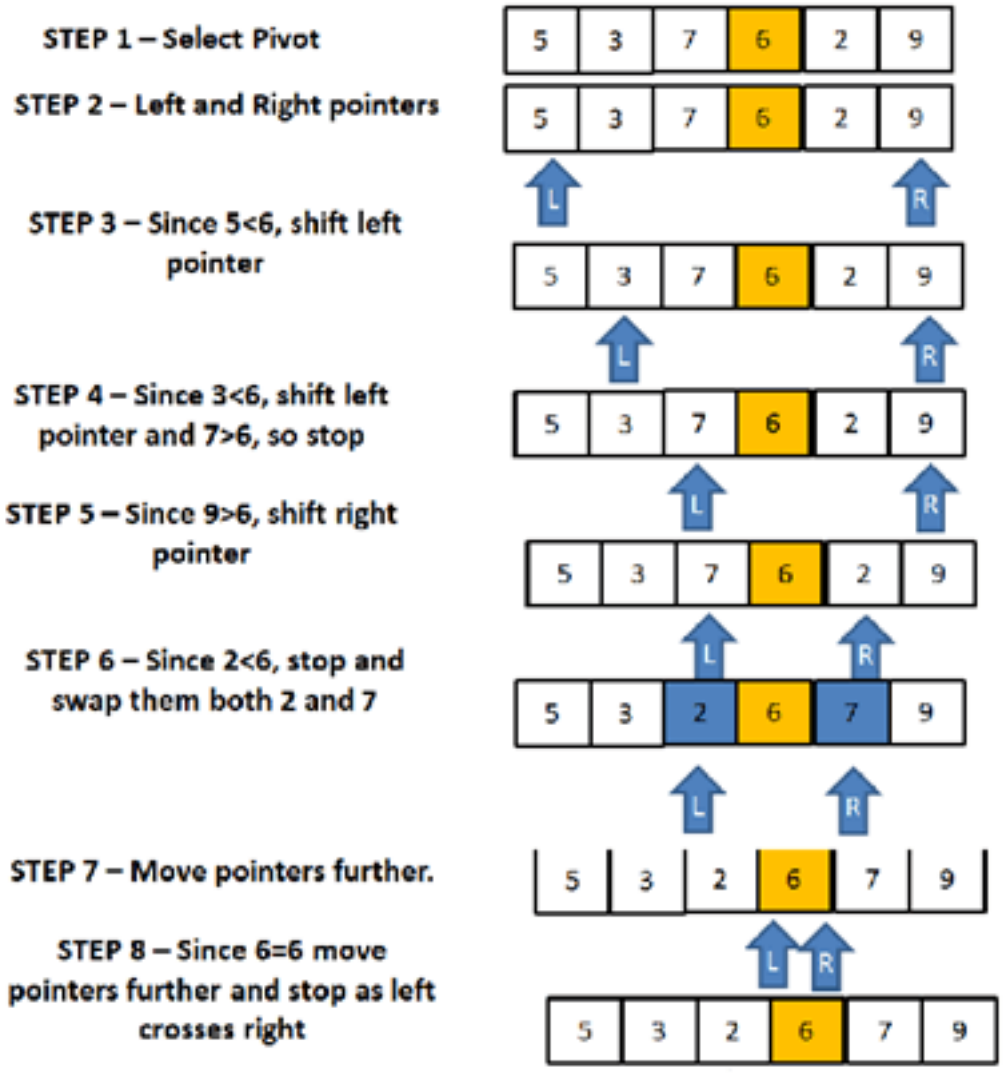
Step 4) Compare the element pointing with the left pointer, and if it is less than the pivot element, then move the left pointer to the right (add 1 to the left index). Continue this until the left side element is greater than or equal to the pivot element.

Step 5) Compare the element pointing with the right pointer. If it is greater than the pivot element, move the right pointer to the left (subtract 1 to the right index). Continue this until the right-side element is less than or equal to the pivot element.

Step 6) Check if the left pointer is less than or equal to a right pointer, then saw the elements in these pointers' locations.

Step 7) Increment the left pointer and decrement the right pointer.

Step 8) If the left pointer index is still less than the right pointer's index, repeat the process; else, return the left pointer's index.



So, let us see these steps with an example. Let us consider an array of elements which we need to sort is [5,3,7,6,2,9].

Here are the steps to perform Quick sort that is being shown with an example [5,3,7,6,2,9].

STEP 1) Determine pivot as a middle element. So, **7** is the pivot element.

STEP 2) Start left and right pointers as first and last elements of the array, respectively. The left pointer points to 5 at index 0, and the right pointer points to **9** at index 5.

STEP 3) Compare the left pointer element with the pivot element, since 5 < 6 shift left pointer to the right to index 1.

STEP 4) Now, still 3 <6, so shift the left pointer to one more index to the right. Now 7 > 6 stops incrementing the left pointer, and now the left pointer is index 2.

STEP 5) Now, compare the value at the right pointer with the pivot element. Since 9 > 6, move the right pointer to the left. Now, as 2 < 6, stop moving the right pointer.

STEP 6) Swap both values present at left and right pointers with each other.

STEP 7) Move both pointers one more step.

STEP 8) Since 6 = 6, move pointers to one more step and stop as the left pointer crosses the right pointer and returns the left pointer's index.

Here, based on the above approach, we need to write code for swapping elements and partitioning the array as mentioned in the above steps.

Example:

```
var items = [5,3,7,6,2,9];
function swap(items, leftIndex, rightIndex){
    var temp = items[leftIndex];
    items[leftIndex] = items[rightIndex];
    items[rightIndex] = temp;
}
function: partition(items, left, right) {
    var pivot   = items[Math.floor((right + left) / 2)], //middle element
        i       = left, //left pointer
        j       = right; //right pointer
    while (i <= j) {
        while (items[i] < pivot) {
            i++;
        }
        while (items[j] > pivot) {
            j--;
        }
        if (i <= j) {
            swap(items, i, j); //sawpping two elements
            i++;
            j--;
        }
    }
    return i;
}

function quickSort(items, left, right) {
    var index;
    if (items.length > 1) {
        index = partition(items, left, right); //index returned from partition
        if (left < index - 1) { //more elements on the left side of the pivot
            quickSort(items, left index - 1);
        }
```

```
        if (index < right) { //more elements on the right side of the pivot
            quickSort(items, index, right);
        }
    }
    return items;
}
// first call to quick sort
var sortedArray = quickSort(items, 0, items.length - 1);
console.log(sortedArray); //prints [2,3,5,6,7,9]
```

---

95. What is DOM in JavaScript?

JavaScript can access all the elements in a web page using the Document Object Model (DOM). The web browser creates a DOM of the webpage when the page is loaded.

---

96. How to use DOM and Events?

Using DOM, JavaScript can perform multiple tasks. It can create new elements and attributes, change the existing elements and attributes and even remove existing elements and attributes. JavaScript can also react to existing events and create new events in the page.

1. getElementById, innerHTML Example
2. getElementById: To access elements and attributes whose id is set.
3. innerHTML: To access the content of an element.

```
<html>
<head>
    <title>DOM!!!</title>
</head>
<body>
  <h3 id="one">Welcome</h3>
  <p>This is the welcome message.</p>
  <h3>Technology</h3>
  <p>This is the technology section.</p>
  <script type="text/javascript">
        var text = document.getElementById("one").innerHTML;
        alert("The first heading is " + text);
  </script>
</body>
</html>
```

2.getElementsByTagName Example

getElementsByTagName: To access elements and attributes using tag name. This method will return an array of all the items with the same tag name.

```
<html>

<head>
```

```
    <title>DOM!!!</title>
</head>
<body>
  <h3>Welcome</h3>
  <p>This is the welcome message.</p>
  <h3>Technology</h3>
  <p id="second">This is the technology section.</p>
  <script type="text/javascript">
    var paragraphs = document.getElementsByTagName("p");
    alert("Content in the second paragraph is " + paragraphs[1].innerHTML);
    document.getElementById("second").innerHTML = "The orginal message is changed.";
  </script>
</body>
</html>
```

Event handler Example

1. createElement: To create new element
2. removeChild: Remove an element
3. you can add an **event handler** to a particular element like this

```
document.getElementById(id).onclick=function()
    {
        lines of code to be executed
    }
```

OR

```
document.getElementById(id).addEventListener("click", functionname)
```

Example:

```
<html>
<head>
    <title>DOM!!!</title>
</head>
<body>
  <input type="button" id="btnClick" value="Click Me!!" />
  <script type="text/javascript">
    document.getElementById("btnClick").addEventListener("click", clicked);
    function clicked()
    {
```

```
            alert("You clicked me!!!");
    }
  </script>
</body>
</html>
```

---

97. What is External JavaScript?

You plan to display the current date and time on all your web pages. Suppose you wrote the code and copied it in to all your web pages (say 100). But later, you want to change the format in which the date or time is displayed. In this case, you will have to make changes to all the 100 web pages. This will be a very time-consuming and difficult task.

So, save the JavaScript code in a new file with the extension .js. Then, add a line of code in all your web pages to point to your .js file like this:

```
<script type="text/javascript," src="/currentdetails.js,">
```

Note: It is assumed that the .js file and all your web pages are in the same folder. If the external.js file is in a different folder, you need to specify your file's full path in the src attribute.

Example:

```
var currentDate = new Date();
var day = currentDate.getDate();
Var month = currentDate.getMonth() + 1;
var monthName;
var hours = currentDate.getHours();
var mins = currentDate.getMinutes();
var secs = currentDate.getSeconds();
var strToAppend;
It (hours >12 )
{
    hours1 = "0" + (hours - 12);
strToAppend = "PM";
}
else if (hours <12)
{
    hours1 = "0" + hours;
    strToAppend = "AM";
}
else
{
    hours1 = hours;
    strToAppend = "PM";
}
if(mins<10)
mins = "0" + mins;
if (secs<10)
    secs = "0" + secs;
switch (month)
{
    case 1:
```

```
        monthName = "January";
        break;
    case 2:
        monthName = "February";
        break;
    case 3:
        monthName = "March";
        break;
    case 4:
        monthName = "April";
        break;
    case 5:
        monthName = "May";
        break;
    case 6:
        monthName = "June";
        break;
    case 7:
        monthName = "July";
        break;
    case 8:
        monthName = "August";
        break;
    case 9:
        monthName = "September";
        break;
    case 10:
        monthName = "October";
        break;
    case 11:
        monthName = "November";
        break;
    case 12:
        monthName = "December";
        break;
}

var year = currentDate.getFullYear();
var myString;
myString = "Today is " + day +  " - " + monthName + " - " + year + ".<br />Current time
is " + hours1 + ":" + mins + ":" + secs + " " + strToAppend + ".";
document.write(myString);
```

---

98. When to Use Internal and External JavaScript Code?

Suppose you have only a few lines of code that is specific to a particular webpage. In that case, it is better to keep your JavaScript code internal within your HTML document.

On the other hand, if your JavaScript code is used in many web pages, you should consider keeping your code in a separate file. If you wish to make some changes to your code, you have to change only one file, making code maintenance easy. If your code is too long, it is better to keep it in a separate file. This helps in easy debugging.

99. What are Cookies in JavaScript?

A cookie is a piece of data stored on your computer to be accessed by your browser. You also might have enjoyed the benefits of cookies knowingly or unknowingly. Have you ever saved your Facebook password so that you do not have to type it every time you try to login? If yes, then you are using cookies. Cookies are saved as key/value pairs.

Javascript Set-Cookie:

You can create cookies using document. cookie property like this.

```
document.cookie = "cookiename=cookievalue"
```

You can even add an expiry date to your Cookie to remove the particular Cookie from the computer on the specified date. The expiry date should be set in the UTC/GMT format. If you do not set the expiry date, the cookie will be removed when the user closes the browser.

```
document.cookie = "cookiename=cookievalue; expires= Thu, 21 Aug 2014 20:00:00 UTC"
```

You can also set the domain and path to specify which domain and to which directories in the specific domain the Cookie belongs to. By default, a cookie belongs to the page that sets the Cookie.

```
document.cookie = "cookiename=cookievalue; expires= Thu, 21 Aug 2014 20:00:00 UTC;
path=/
```

//create a cookie with a domain to the current page and a path to the entire domain.

JavaScript get Cookie

You can access the Cookie like this, which will return all the cookies saved for the current domain.

```
var x =  document.cookie
```

JavaScript Delete Cookie

To delete a cookie, you just need to set the cookie's value to empty and set the value of expires to a passed date.

Example:

```
<html>
<head>
    <title>Cookie!!!</title>
    <script type="text/javascript">
        function createCookie(cookieName,cookieValue,daysToExpire)
        {
          var date = new Date();
```

```
          date.setTime(date.getTime()+(daysToExpire*24*60*60*1000));
          document.cookie = cookieName + "=" + cookieValue + "; expires=" +
date.toGMTString();
        }
        function accessCookie(cookieName)
        {
          var name = cookieName + "=";
          var allCookieArray = document.cookie.split(';');
          for(var i=0; i<allCookieArray.length; i++)
          {
            var temp = allCookieArray[i].trim();
            if (temp.indexOf(name)==0)
            return temp.substring(name.length,temp.length);
             }
            return "";
        }
        function checkCookie()
        {
          var user = accessCookie("testCookie");
          if (user!="")
            alert("Welcome Back " + user + "!!!");
          else
          {
            user = prompt("Please enter your name");
            num = prompt("How many days you want to store your name on your computer?");
            It (user!="" && user!=null)
            {
            createCookie("testCookie", user, num);
            }
          }
        }
    </script>
</head>
<body onload="checkCookie()"></body>
</html>
```

---

100. Give an example of JavaScript Multiplication Table

Here, are example of simple multiplication table asking the user the number of rows and columns he wants.

Example:

```
<html>
<head>
  <title>Multiplication Table</title>
  <script type="text/javascript">
    var rows = prompt("How many rows for your multiplication table?");
    var cols = prompt("How many columns for your multiplication table?");
    if(rows == "" || rows == null)
            rows = 10;
    if(cols== "" || cols== null)
            cols = 10;
    createTable(rows, cols);
    function createTable(rows, cols)
```

```
    {
      var j=1;
      var output = "<table border='1' width='500' cellspacing='0'cellpadding='5'>";
      for(i=1;i<=rows;i++)
      {
        output = output + "<tr>";
        while(j<=cols)
        {
            output = output + "<td>" + i*j + "</td>";
             j = j+1;
           }
            output = output + "</tr>";
            j = 1;
    }
    output = output + "</table>";
    document.write(output);
    }
  </script>
</head>
<body>
</body>
</html>
```

---

101. Explain Popup Message using event with example

Display a simple message “Welcome!!!” on your demo webpage and when the user hovers over the message, a popup should be displayed with a message “Welcome to my WebPage!!!”.

Example:

```
<html>
    <head>

    <title>Event!!!</title>

    <script type="text/javascript">

    function trigger()

    {

    document.getElementById("hover").addEventListener("mouseover", popup);

    function popup()

    {

    alert("Welcome to my WebPage!!!");

    }

    }

    </script>
```

```
<style>

p{
     font-size:50px;
     position: fixed;
     left: 550px;
     top: 300px;
}
</style>
</head>
<body  onload="trigger();">
<p id="hover">Welcome!!!</p>
</body>
</html>
```

These interview questions will also help in your viva(orals)

----------xxxxxxxx----------

REACT

1) What is Reactjs?

React is a JavaScript library that makes building user interfaces easy. It was developed by Facebook.

---

2) Does React use HTML?

No, It uses JSX, which is similar to HTML.

---

3) When was React first released?

React was first released on March 2013.

---

4) Give me two most significant drawbacks of React

- Integrating React with the MVC framework like Rails requires complex configuration.

- React require the users to have knowledge about the integration of user interface into MVC
framework.

---

5) State the difference between Real DOM and Virtual DOM

Real DOM	Virtual DOM
It is updated slowly.	It updates faster.
It allows a direct update from HTML.	It cannot be used to update HTML directly.
It wastes too much memory.	Memory consumption is less

---

6) What is Flux Concept In React?

Facebook widely uses flux architecture concept for developing client-side web applications. It is not a framework or a library. It is simply a new kind of architecture that complements React and the concept of Unidirectional Data Flow.

---

7) Define the term Redux in React

Redux is a library used for front end development. It is a state container for JavaScript applications which should be used for the applications state management. You can test and run an application developed with Redux in different environments.

---

8) What is the ‘Store’ feature in Redux?

Redux has a feature called ‘Store’ which allows you to save the application’s entire State at one place. Therefore all it’s component’s State are stored in the Store so that you will get regular updates directly from the Store. The single state tree helps you to keep track of changes over time and debug or inspect the application.

---

9) What is an action in Redux?

It is a function which returns an action object. The action-type and the action data are always stored in the action object. Actions can send data between the Store and the software application. All information retrieved by the Store is produced by the actions.

---

10) Name the important features of React

Here, are important features of React.

- Allows you to use 3rd party libraries
- Time-Saving
- Faster Development
- Simplicity and Composable
- Fully supported by Facebook.
- Code Stability with One-directional data binding
- React Components

---

11) Explain the term stateless components

Stateless components are pure functions that render DOM-based solely on the properties provided to them.

---

12) Explain React Router

React Router is a routing library which allows you to add new screen flows to your application, and it also keeps URL in sync with what's being shown on the page.

---

13) What are the popular animation package in React ecosystem?

Popular animation package in React ecosystem are

- React Motion
- React Transition Group

---

14) What is Jest?

Jest is a JavaScript unit testing framework created by Facebook based on Jasmine. It offers automated mock creation and a jsdom environment. It is also used as a testing component.

---

15) What is dispatcher?

A dispatcher is a central hub of app where you will receive actions and broadcast payload to registered callbacks.

---

16) What is meant by callback function? What is its purpose?

A callback function should be called when setState has finished, and the component is re-rendered.

As the setState is asynchronous, which is why it takes in a second callback function.

---

17) Explain the term high order component

A higher-order component also shortly known as HOC is an advanced technique for reusing component logic. It is not a part of the React API, but they are a pattern which emerges from React's compositional nature.

---

18) Explain the Presentational segment

A presentational part is a segment which allows you to renders HTML. The segment's capacity is presentational in markup.

19) What are Props in react js?

Props mean properties, which is a way of passing data from parent to child. We can say that props are just a communication channel between components. It is always moving from parent to child component.

---

20) What is the use of a super keyword in React?

The super keyword helps you to access and call functions on an object's parent.

---

21) Explain yield catchphrase in JavaScript

The yield catchphrase is utilized to delay and resume a generator work, which is known as yield catchphrase.

---

22) Name two types of React component

Two types of react Components are:

- Function component
- Class component

---

23) Explain synthetic event in React js

Synthetic event is a kind of object which acts as a cross-browser wrapper around the browser's native event. It also helps us to combine the behaviors of various browser into signal API.

---

24) What is React State?

It is an object which decides how a specific component renders and how it behaves. The state stores the information which can be changed over the lifetime of a React component.

---

25) How can you update state in react js?

A state can be updated on the component directly or indirectly.

---

26) Explain the use of the arrow function in React

The arrow function helps you to predict the behavior of bugs when passed as a callback. Therefore, it prevents bug caused by this all together.

---

27) What are the lifecycle steps of React?

Important lifecycle steps of React js are:

- Initialization
- State/Property updates
- Destruction are the lifecycle of React

---

28) State the main difference between Pros and State

The main difference the two is that the State is mutable and Pros are immutable.

---

29) Explain pure components in React js

Pure components are the fastest components which can replace any component with only a render(). It helps you to enhance the simplicity of the code and performance of the application.

30) What kind of information controls a segment in React?

There are mainly two sorts of information that control a segment: State and Props

- State: State information that will change, we need to utilize State.
- Props: Props are set by the parent and which are settled all through the lifetime of a part.

---

31) What is ‘create-react-app’?

‘create-react-app’ is a command-line tool which allows you to create one basic react application.

---

32) Explain the use of ‘key’ in react list

Keys allow you to provide each list element with a stable identity. The keys should be unique.

---

33) What are children prop?

Children props are used to pass component to other components as properties. You can access it by using

```
{props.children}
```

---

34) Explain error boundaries?

Error boundaries help you to catch Javascript error anywhere in the child components. They are most used to log the error and show a fallback UI.

---

35) What is the use of empty tags <> </>?

Empty tags are used in React for declaring fragments.

---

36) Explain strict mode

StrictMode allows you to run checks and warnings for react components. It runs only on development build. It helps you to highlight the issues without rendering any visible UI.

---

37) What are reacted portals?

Portal allows you to render children into a DOM node. **CreatePortalmethod** is used for it.

---

38) What is Context?

React context helps you to pass data using the tree of react components. It helps you to share data globally between various react components.

---

39) What is the use of Webpack?

Webpack in basically is a module builder. It is mainly runs during the development process.

---

40) What is Babel in React js?

Babel, is a JavaScript compiler that converts latest JavaScript like ES6, ES7 into plain old ES5 JavaScript that most browsers understand.

---

41) How can a browser read JSX file?

If you want the browser to read JSX, then that JSX file should be replaced using a JSX transformer like Babel and then send back to the browser.

---

42) What are the major issues of using MVC architecture in React?

Here are the major challenges you will face while handling MVC architecture:

- DOM handling is quite expensive
- Most of the time applications were slow and inefficient
- Because of circular functions, a complex model has been created around models and ideas

---

43) What can be done when there is more than one line of expression?

At that time a multi-line JSX expression is the only option left for you.

---

44) What is the reduction?

The reduction is an application method of handling State.

---

45) Explain the term synthetic events

It is actually a cross-browser wrapper around the browser's native event. These events have interface stopPropagation() and preventDefault().

---

46) When should you use the top-class elements for the function element?

If your element does a stage or lifetime cycle, we should use top-class elements.

---

47) How can you share an element in the parsing?

Using the State, we can share the data.

---

48) Explain the term reconciliation

When a component's state or props change then rest will compare the rendered element with previously rendered DOM and will update the actual DOM if it is needed. This process is known as reconciliation.

---

49) Ho can you re-render a component without using setState() function?

You can use forceUpdate() function for re-rending any component.

---

50) Can you update props in react?

You can't update props in react js because props are read-only. Moreover, you can not modify props received from parent to child.

---

51) Explain the term 'Restructuring.'

Restructuring is extraction process of array objects. Once the process is completed, you can separate each object in a separate variable.

---

52) Can you update the values of props?

It is not possible to update the value of props as it is immutable.

---

53) Explain the meaning of Mounting and Demounting

- The process of attaching the element to the DCOM is called mounting.
- The process of detaching the element from the DCOM is called the demounting process.

---

54) What is the use of 'prop-types' library?

'Prop-types' library allows you to perform runtime type checking for props and similar object in a recent application.

---

55) Explain react hooks

React hooks allows you to use State, and other React features without writing a class.

---

56) What are Fragments?

You can use fragment keyword to group a list of children components without using any extra nodes to the DOM.

For example :

```
render() {

return (
);
}
```

---

57) What is the main difference between createElement and cloneElment?

- createElement is used by react to create react elements.
- cloneElement is used to clone an element and pass it new props.

---

58) What are Controlled Components?

Controlled components are component which controls the input elements.

---

59) Why do you need to use props.children?

This props.children allow you to pass a component as data to other components.

---

60) List down some of the methods in a react-dom package

Important methods for react-dom packages are:

- render()
- hydrate()
- createPortal()
- unmountComponentAtNode()
- findDOMNode()

---

61) How can we do server-side rendering in React?

We can use reaction serve to do the server-side rendering.

---

62) State the difference between getIntialState() and constructor()?

If you want to create one component by extending 'React. Component', the constructor helps you to initialize the State. But, if you want to create by using 'Reat.createClass.' then you should use 'genInitiaState.'

---

63) What is refs?

Ref are an attribute of the DOM elements. The primary purpose of the refs is to find the DOM elements easily.

---

64) What is ComponentWillMount()

componentWillMount() is to make API calls once the component is initiated and configure the values into the state. To make an API call, use an HttpClient such as Axios, or we can use fetch() to trigger the AJAX call.

---

65) How to dispatch the data in-store?

We can dispatch the data to another component which should be based on the action which stores the parent component.

---

66) How will be you able to handle more action using redux?

In order to create the same component in more action flow, we are using the same functionality in various modules.

---

67) How can you spill the reducers?

We can spill the rescues based on the event actions. That action should be split in separate modules.

68) Name any five predefined prototypes used in React

Most important protoype used in React js are:

- number
- string
- array
- object
- element

---

69) What is the purpose of using bindActionsCreators?

BindActionCreator helps you to bind the event based on the action dispatcher to the HTML element.

---

70) What is REFS in React

Ref is a reference to the element. It should be avoided in most cases. However, sometimes it is used when you need to access DOM or instance of the component directly.

---

71) Can JSX element be attached to other JSX components?

Yes, you can use attach JSX element with other JSX components which is very much similar to nesting HTML elements.

---

72) What is the Current Stable Version of React?

The current stable version of React is version 17.5

---

73) Name out an important feature of Redux workflow features

Important features of Redux workflow are:

- Reset: Helps you to reset the State of the Store
- Revert: Allows you to roll back to the last committed State
- Sweep: All disable action that you might fire by mistake will be removed
- Commit: Helps you to make the current State the initial State.

---

74) State the difference between React JS and React Native

React JS is a front end open-source JavaScript library used for building UIs, While React Native, is an open-source, mobile framework which allows developers to user React on platforms like Android and iOS.

1. How do you create refs in React?

Can your candidates explain that they must use ***React.createRef()*** to create refs in React.js? They should also understand that the ref is often assigned to an instance property when constructing a component.

2. Name two advantages of using React.js.

Candidates may mention several advantages of using React.js when responding to this question. For example, they may explain that the library helps them build high-quality user interfaces or that it permits them to write custom components.

3. Name one advantage of using Flux.

Your candidates may start by explaining that Flux is a JavaScript architecture that operates on a unidirectional data flow. But do they know Flux offers many advantages for projects

with dynamic data? For example, can they explain that Flux helps ensure dynamic data is updated effectively?

4. Name three advantages of using React Hooks.

Three advantages of using React Hooks are that they enable developers to:

- Enhance component tree readability
- Share logic among different components
- Effectively handle the setup of side effects

5. How often does the React *useState* update? Why?

Since developers use ***useState*** to enhance performance by creating queues, React doesn't update changes immediately. Candidates should know that ***useState*** doesn't implement changes to the ***state*** object directly; instead, the updates occur asynchronously.

6. In which situation would you use refs in React?

Advanced candidates should understand that they can use React refs to access a DOM element. They may also explain that they would use refs to access an element they have created to change a child component's value.

7. In which situation would you use *useMemo()* in React?

Do your candidates know that developers can use ***useMemo()*** to cache a variable's value along with dependency lists? Can they explain that they would use ***useMemo()*** to help them avoid unnecessary re-renders? They may also say that ***useMemo()*** can be useful in situations where there are high processing amounts.

8. Why would you use super constructors with props arguments?

Candidates may explain that they pass props to super constructors to access and use ***this.props*** in the constructor. They may mention that when they implement

a ***constructor()*** function within a React component, they use ***super()*** to call the parent constructor.

9. How would you avoid binding in React?

Candidates who have advanced React skills should be aware that they can use arrow functions in class properties to avoid binding in React. They may mention that class properties are a new feature and, to use them, a developer must enable transform-class-properties.

10. Which method would you use to handle events in React?

Can your candidates explain that to handle events in React, they would name them using camelCase (instead of lowercase)? They should be able to also explain that when they use JSX, they pass a function as the event handler (instead of a string).

----------**xxxxx**----------

MONGODB

1) Explain what is MongoDB?

Mongo-DB is a document database which provides high performance, high availability and easy scalability.

---

2) What is "Namespace" in MongoDB?

MongoDB stores BSON (Binary Interchange and Structure Object Notation) objects in the collection. The concatenation of the collection name and database name is called a namespace.

---

3) What is sharding in MongoDB?

The procedure of storing data records across multiple machines is referred as sharding. It is a MongoDB approach to meet the demands of data growth. It is the horizontal partition of data in a database or search engine. Each partition is referred as shard or database shard.

---

4) How can you see the connection used by Mongos?

To see the connection used by Mongos use db_adminCommand ("connPoolStats");

---

5) Explain what is a replica set?

A replica set is a group of mongo instances that host the same data set. In replica set, one node is primary, and another is secondary. From primary to the secondary node all data replicates.

---

6) How replication works in MongoDB?

Across multiple servers, the process of synchronizing data is known as replication. It provides redundancy and increase data availability with multiple copies of data on different database server. Replication helps in protecting the database from the loss of a single server.

---

7) While creating Schema in MongoDB what are the points need to be taken in consideration?

Points need to be taken in consideration are

- Design your schema according to user requirements
- Combine objects into one document if you use them together. Otherwise, separate them
- Do joins while write, and not when it is on read
- For most frequent use cases optimize your schema
- Do complex aggregation in the schema

---

8) What is the syntax to create a collection and to drop a collection in MongoDB?

- Syntax to create collection in MongoDB is `db.createCollection(name,options)`
- Syntax to drop collection in MongoDB is `db.collection.drop()`

---

9) Explain what is the role of profiler in MongoDB?

MongoDB database profiler shows performance characteristics of each operation against the database. You can find queries using the profiler that are slower than they should be.

---

10) Explain can you move old files in the moveChunk directory?

Yes, it is possible to move old files in the moveChunk directory, during normal shard balancing operations these files are made as backups and can be deleted once the operations are done.

---

11) To do safe backups what is the feature in MongoDB that you can use?

Journaling is the feature in MongoDB that you can use to do safe backups.

---

12) Mention what is Objecld composed of?

Objectld is composed of

- Timestamp
- Client machine ID
- Client process ID
- 3 byte incremented counter

---

13) Mention what is the command syntax for inserting a document?

For inserting a document command syntax is `database.collection.insert (document)`.

---

14) Mention how you can inspect the source code of a function?

To inspect a source code of a function, without any parentheses, the function must be invoked.

---

15) What is the command syntax that tells you whether you are on the master server or not? And how many master does MongoDB allow?

Command syntax `Db.isMaster()` will tell you whether you are on the master server or not. MongoDB allows only one master server, while couchDB allows multiple masters.

---

16) Mention the command syntax that is used to view Mongo is using the link?

The command syntax that is used to view mongo is using the link is `db._adminCommand("connPoolStats.")`

---

17) Explain what are indexes in MongoDB?

Indexes are special structures in MongoDB, which stores a small portion of the data set in an easy to traverse form. Ordered by the value of the field specified in the index, the index stores the value of a specific field or set of fields.

---

18) Mention what is the basic syntax to use index in MongoDB?

The basic syntax to use in MongoDB is `>db.COLLECTION_NAME.ensureIndex ( {KEY:1} )`. In here the key is the the name of the COLUMN (or KEY:VALUE pair) which is present in the documents.

---

19) Explain what is GridFS in MongoDB?

For storing and retrieving large files such as images, video files and audio files GridFS is used. By default, it uses two files fs.files and fs.chunks to store the file's metadata and the chunks.

---

20) What are alternatives to MongoDB?

Cassandra, CouchDB, Redis, Riak, Hbase are a few good alternatives.

----------xxxxxxx----------

NODE.JS

1) What is node.js?

Node.js is a Server side scripting which is used to build scalable programs. Its multiple advantages over other server side languages, the prominent being non-blocking I/O.

---

2) How node.js works?

Node.js works on a v8 environment, it is a virtual machine that utilizes JavaScript as its scripting language and achieves high output via non-blocking I/O and single threaded event loop.

---

3) What do you mean by the term I/O?

I/O is the shorthand for input and output, and it will access anything outside of your application. It will be loaded into the machine memory to run the program, once the application is started.

---

4) What does event-driven programming mean?

In computer programming, event driven programming is a programming paradigm in which the flow of the program is determined by events like messages from other programs or threads. It is an application architecture technique divided into two sections 1) Event Selection 2) Event Handling.

---

5) Where can we use node.js?

Node.js can be used for the following purposes.

- Web applications (especially real-time web apps)
- Network applications
- Distributed systems
- General purpose applications

---

6) What is the advantage of using node.js?

- It provides an easy way to build scalable network programs
- Generally fast
- Great concurrency
- Asynchronous everything
- Almost never blocks

---

7) What are the two types of API functions in Node.js?

The two types of API functions in Node.js are

- Asynchronous, non-blocking functions
- Synchronous, blocking functions

8) What is control flow function?

A generic piece of code which runs in between several asynchronous function calls is known as control flow function.

9) Explain the steps how “Control Flow” controls the functions calls?

- Control the order of execution
- Collect data
- Limit concurrency
- Call the next step in program

10) Why Node.js is single threaded?

For async processing, Node.js was created explicitly as an experiment. It is believed that more performance and scalability can be achieved by doing async processing on a single thread under typical web loads than the typical thread based implementation.

11) Does node run on windows?

Yes – it does. Download the MSI installer from https://nodejs.org/download/

12) Can you access DOM in node?

No, you cannot access DOM in node.

13) Using the event loop what are the tasks that should be done asynchronously?

- I/O operations
- Heavy computation
- Anything requiring blocking

14) Why node.js is quickly gaining attention from JAVA programmers?

Node.js is quickly gaining attention as it is a loop based server for JavaScript. Node.js gives user the ability to write the JavaScript on the server, which has access to things like HTTP stack, file I/O, TCP and databases.

---

15) What are the two arguments that async.queue takes?

The two arguments that async.queue takes

- Task function
- Concurrency value

---

16) What is an event loop in Node.js?

To process and handle external events and to convert them into callback invocations an event loop is used. So, at I/O calls, node.js can switch from one request to another.

---

17) Mention the steps by which you can async in Node.js?

By following steps you can async Node.js

- First class functions
- Function composition
- Callback Counters
- Event loops

---

18) What are the pros and cons of Node.js?

Pros:

- If your application does not have any CPU intensive computation, you can build it in Javascript top to bottom, even down to the database level if you use JSON storage object DB like MongoDB.
- Crawlers receive a full-rendered HTML response, which is far more SEO friendly rather than a single page application or a websockets app run on top of Node.js.

Cons:

- Any intensive CPU computation will block node.js responsiveness, so a threaded platform is a better approach.
- Using relational database with Node.js is considered less favourable.

19) How Node.js overcomes the problem of blocking of I/O operations?

Node.js solves this problem by putting the event based model at its core, using an event loop instead of threads.

20) What is the difference between Node.js vs Ajax?

The difference between Node.js and Ajax is that, Ajax (short for Asynchronous Javascript and XML) is a client side technology, often used for updating the contents of the page without refreshing it. While,Node.js is Server Side Javascript, used for developing server software. Node.js does not execute in the browser but by the server.

21) What are the Challenges with Node.js?

Emphasizing on the technical side, it's a bit of challenge in Node.js to have one process with one thread to scale up on multi core server.

22) What does it mean "non-blocking" in node.js?

In node.js "non-blocking" means that its IO is non-blocking. Node uses "libuv" to handle its IO in a platform-agnostic way. On windows, it uses completion ports for unix it uses epoll or kqueue etc. So, it makes a non-blocking request and upon a request, it queues it within the event loop which call the JavaScript 'callback' on the main JavaScript thread.

23) What is the command that is used in node.js to import external libraries?

Command "require" is used for importing external libraries, for example, "var http=require ("http")". This will load the http library and the single exported object through the http variable.

24) Mention the framework most commonly used in node.js?

"Express" is the most common framework used in node.js.

25) What is 'Callback' in node.js?

Callback function is used in node.js to deal with multiple requests made to the server. Like if you have a large file which is going to take a long time for a server to read and if you don't want a

server to get engage in reading that large file while dealing with other requests, call back function is used. Call back function allows the server to deal with pending request first and call a function when it is finished.

----------xxxxxxx----------

EXPRESS JS

1. How do you install an express application generator for scaffolding?

Express application generator is used for quickly creating an application skeleton. The given command is used for installing the express application generator.

```
npm install expr
ess-generator –g
express myApp
```

It will create the project "myApp" with some files. Then we install all the dependencies stated in package.json using the given command:

```
cd myApp
npm install
```

2. How do you install a yeoman for scaffolding?

Generators are used by yeoman for scaffolding the applications. And we can use the following command to install yeoman.

```
npm install –g yeoman
```

3. Mention the arguments that are available in an Express JS route handler function.

The arguments that are available in the route handler function of Express JS are given below:

- **Res** - It is the response object.
- **Req** - It is the request object
- **Next (optional)** - This argument is used for passing the management to any of the above-given route handlers.

[Learn Complete *Express JS Tutorial*]

4. Mention the ways of debugging on Linux as well as Windows.

Debugging on Windows can be done as follows:

```
set DEBUG = express:*
node app.js
```

And debugging on Linux can be done as follows:

```
DEBUG = express:*
node app.js
```

5. List the built-in middleware functions provided by Express.

Express JS provides the following built-in middleware functions:

i. **Static**: We use it for serving static assets like images, HTML files, etc.

ii. **JSON**: This is available in Express 4.16.0+. And we use it for passing the incoming requests with JSON payloads.

iii. **URL encoded**: This is also available in Express 4.16.0+. And we use it for passing the incoming requests with URL-encoded payloads.

6. Mention some third-party middleware provided by Express JS.

Some of the many third-party middlewares that the Express JS provides are:

- Cookie-parser
- Body-parser
- Cors
- Mongoose
- Express-validator
- Sequelize.

7. When is application-level Middleware used?

We use the application-level Middleware for binding the app object with the help of the app.use() method. It can be applied on all routes. The syntax is given below:

```
// This Middleware executes for each route.
App.use(function (req, res, next) {
  console.log('Current Time:', Date.now())
  next()
})
```

8. Tell us about Router-level Middleware and Built-in Middleware.

Router-level Middleware - We use the router-level Middleware for binding with a particular instance of Express.Router().

Built-in Middleware - The version 4.x of Express introduced the built-in Middleware. The dependency on connecting gets removed by use of this Middleware.

9. Mention some of the databases with which Express JS is integrated.

A myriad of NoSQL and RDBMS databases are supported by Express JS, such as:

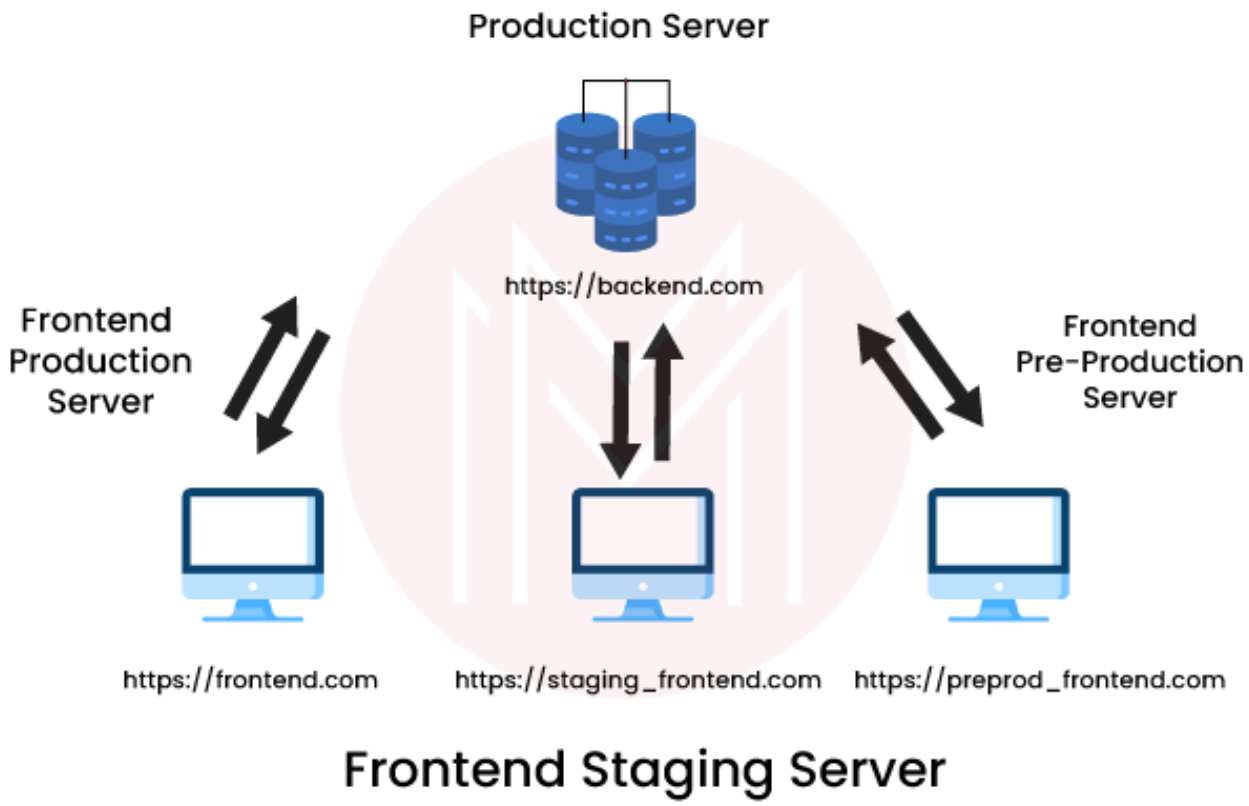
- MySQL
- MongoDB
- PostgreSQL
- SQLite
- Oracle
- SQLite.

Related Article MongoDB vs PostgreSQL

10. What is meant by CORS in Express JS? And what are the ways by which it can be achieved?

CORS is the acronym for Cross-origin resource sharing. We can request the restricted resources from another server or domain using this mechanism. And we can do this mainly in the following three ways:

i. Express cors module
ii. Res.header() (or res.set()): Multiple headers can be set using this way.
iii. Res.setHeader(): Only a single header can be set in this way.



11. What ways are provided by Express JS to configure the properties?

Express JS provides us with two ways for configuring the properties, which are given below:

- With process.ENV
- With require.JS.

12. How are the properties configured with process.ENV?

The properties are configured by the given steps:

- We create a file within the project folder and name it ".env."
- All other properties are let to be separate within the ".env" file.
- We can employ any of the properties in server.js.

13. How are the properties configured with require.JS?

The properties are configured by the given steps:

- We create a file within the config folder of the project folder and name it "config.json."
- All the config properties are present there within the config.json file.

14. How can the Express JS application be structured?

There is no specific answer to this question. The dimensions of our application and hence the concerned team define the solution in different situations. The express logic in the Routes and alternative applications can board as many files as we want in any directory structure. The given examples can be read for further inspiration:

- Route map
- Route listings
- MVC vogue controllers.

15. How is the plain HTML rendered?

We don't need to render HTML with the function- res.render(). Instead, we can use the res.sendFile() function if we have a specific file. And we can use the Express.static() middleware function if we serve several assets from a directory.

Express JS Interview Questions For Experienced

16. Write the code for "Hello world" using Express.

Create a new file by the name- index.js and type the following commands:

```
var express = require ('express');
var app = express ();

app.get ('/', function (req, res){
       res.send ("Hello world");
});

app.listen (3000);
```

Now go to the terminal after saving it and type:

```
nodemon index.js
```

17. What are the most used HTTP methods in Express JS?

The following HTTP methods are the most used ones:

- **GET** - A specified resource's representation is requested by the GET method. These requests can only retrieve data.
- **POST** - Posting of the data enclosed in the request as a new entity is done using the POST method. The entity is identified by the URI.
- **PUT** - Modification in the existing entity is done with the data enclosed in the request identified by the URI.
- **DELETE** - The request for deleting the specified source is made by the DELETE method.

18. How can the cookies be manipulated using 'Response.cookie()'?

We use the "res.cookie('username', 'Flavio')" command is used for manipulating. But it accepts a third parameter conaining various options as specified below:

```
res.cookie ('username', 'Flavio', { domain: 'flaviocopes.com', path: '/administrator', secure: true })
res.cookie ('username', 'Flavio' , { expires: new Date(Date.now() + 90000), httpOnly: true})
```

19. When does a Cross-Origin resource get failed in Express JS?

A cross-Origin can fail in the following scenarios-

- If it's to a different domain
- If it's to a different port
- If it's to a different subdomain
- If it's to a different protocol.

20. How can you use a Pug template engine inside Express?

We will first install it using the given command:

```
npm install pug
```

Then we will set it as following when initializing the Express app:

```
const express = require ('express')
const app = express()
app.set ('view engine', 'pug')
```

21. What do you mean by the sanitizing input process?

People can always enter weird things via the client-side code. They use the tools to POST things directly to our endpoints. For this, the Express provides various sanitizing methods to prevent these happenings.

22. Mention some methods for sanitizing.

Consider the following sanitizing methods:

- Trim() will trim the characters at the beginning as well as the ending of a string.
- Escape() will replace ‘, ", <, >, &, / with the corresponding HTML entities.
- NormalizeEmail() will canonicalize an email address.
- Blacklist() will remove the characters appearing on the blacklist.

23. Give an example of HTML form code allowing user to upload a file.

Consider the given example of HTML form code that allows a user to upload file.

```
<form method = “POST” action = “/submit-form”>
  <input type = “file” name = “document” />
  <input type = “submit” />
</form>
```

24. What are the methods that you can call when the Formidable.File objects arise giving the information about the uploaded file?

We can call the following methods in such cases-

- **File.name**- the name of the file
- **File.path**- the path to which the file is written
- **File.size**- the size of the file in bytes
- **File. type**- the file's MIME-type.

25. What steps will you follow to set up HTTP for Express with the help of Let's Encrypt and Certbot?

We will follow the given steps to set up HTTP:

1. Installing certbot
2. Generating the SSL certificate by Certbot
3. Allowing Express to serve the static files
4. Confirming the domain
5. Obtaining the certificate
6. Setting up the renewal.

26. What options are available in the terminal command for generating a skeleton Express JS app?

The options available are given below:

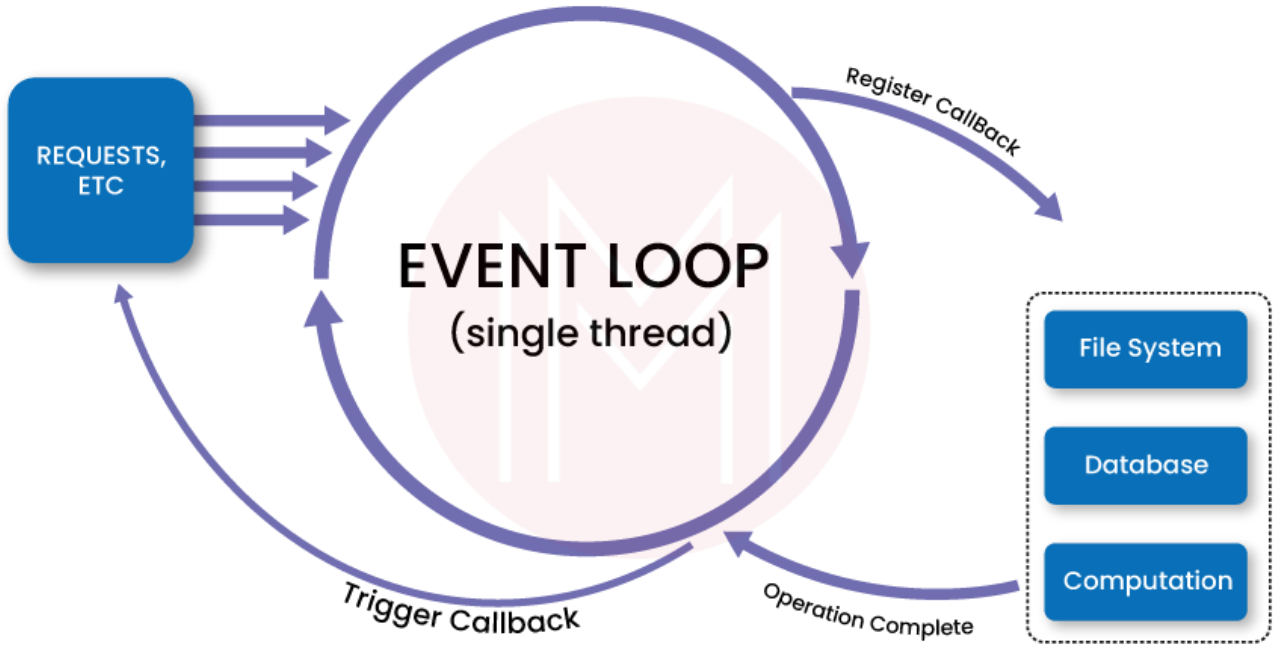
- --sessions or –s for adding session report
- --hogan or –H for adding Hogan.js engine support
- --ejs or –e for adding EJS engine support
- --css <engine> or –c <engine> for adding style sheet support
- -jshtml or –J for adding JSHTML engine support
- --force or –f for forcing app generation on the directory which is non-empty.

Related Article **Basic Examples of Node JS**

27. What is meant by an event-loop in Node JS?

The event-loop manages the async content using a listener and queue. The main thread sends the async function to a different thread whenever it requires to be executed. Alongside, v8 is allowed to

execute the main code. The event loop has different stages including pending callbacks, timers, check, poll, close callbacks, etc. with different FIFO queues.

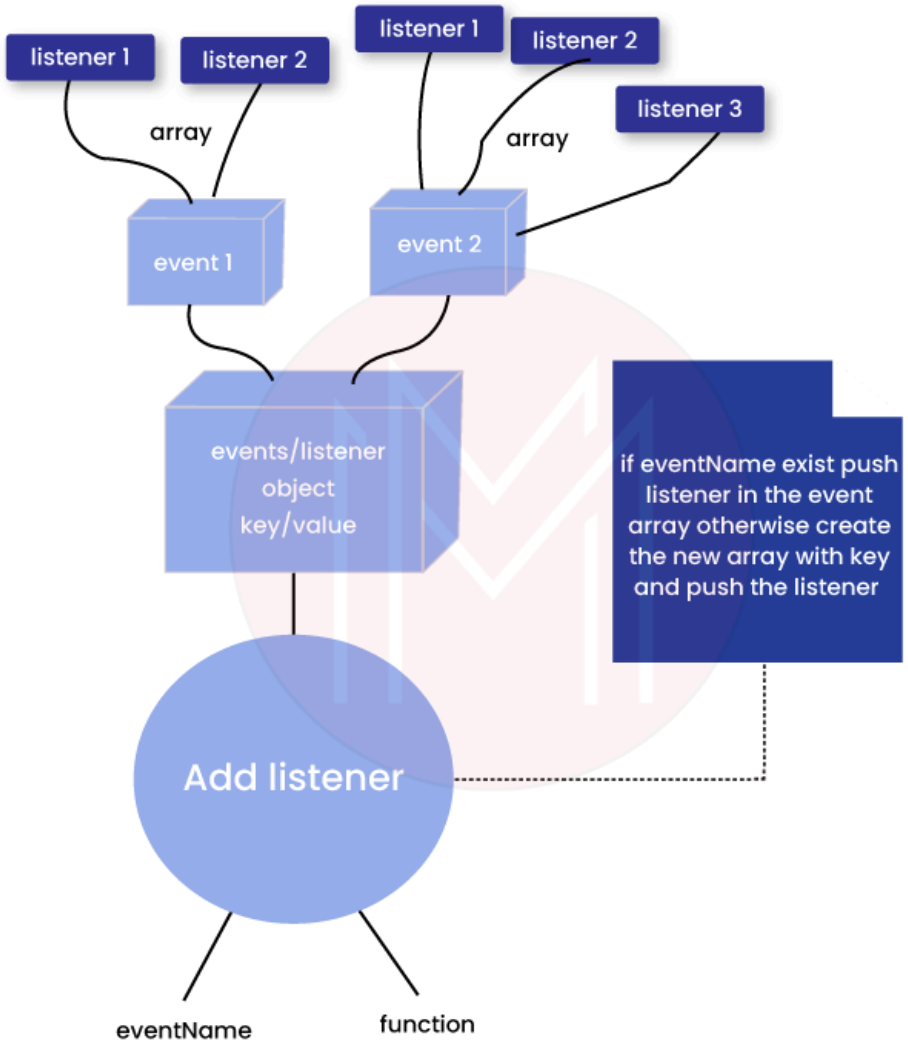


28. Why should the Express server and app be separated?

The server initializes the Middleware, routes, and other application logic. On the other hand, the app contains all the business logic that the server-initiated routes will serve. This enables the encapsulation of the business logic from the application logic for smooth functioning

29. What is meant by an Event emitter in Node JS?

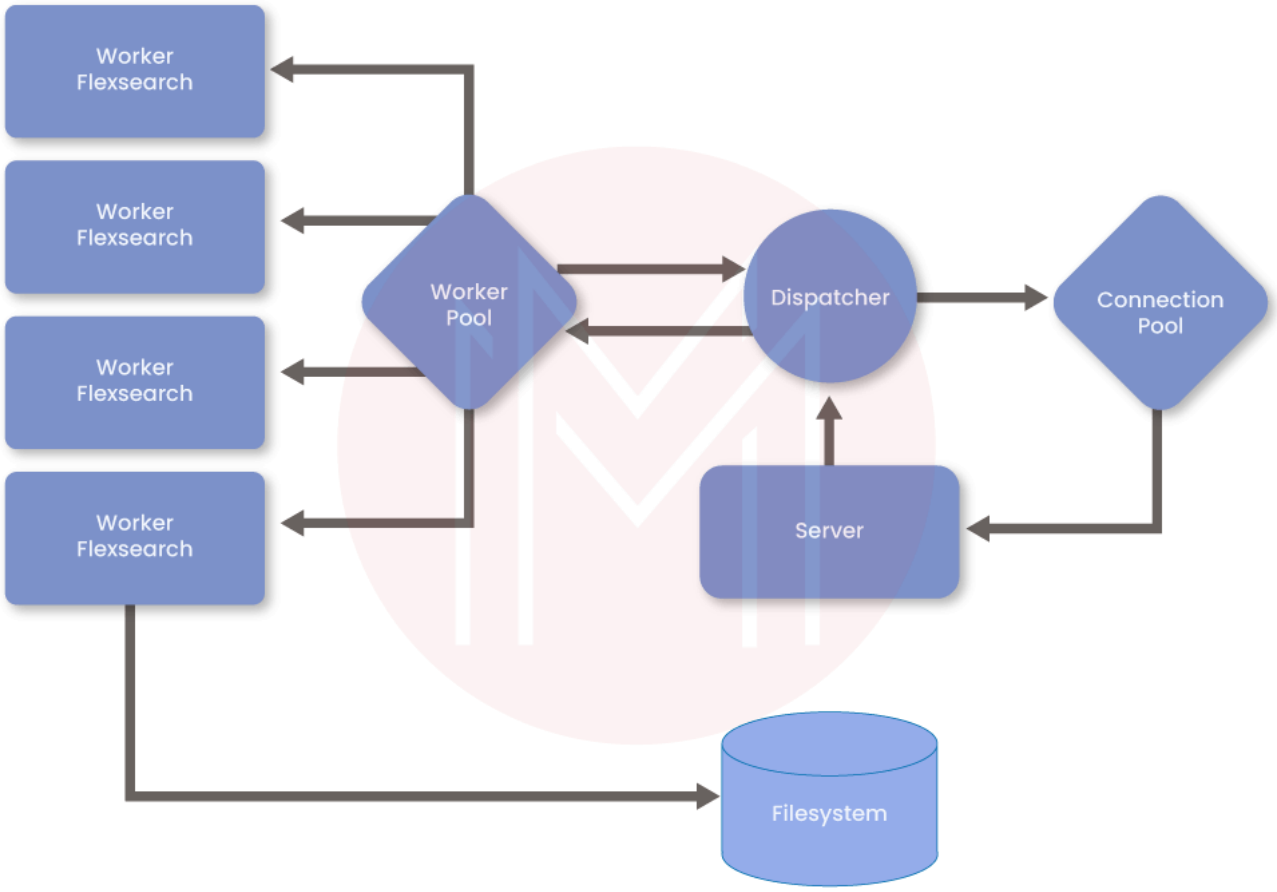
It's a class of Node JS capable of emitting events. We do this by attaching the named events emitted by the object by using the function- eventEmitter.on().



30. Differentiate between worker threads and clusters in Node JS.

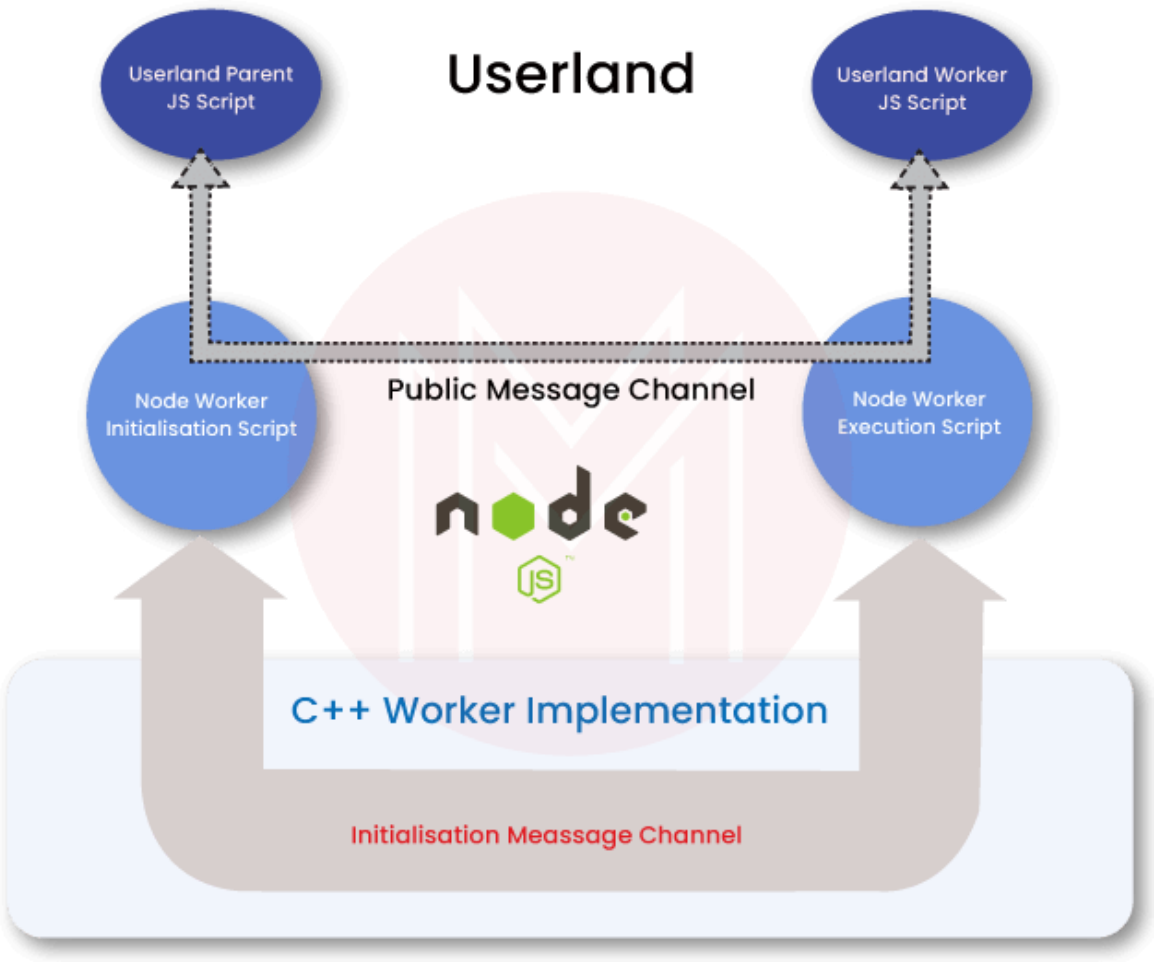
Cluster:

- Each CPU has one process with an IPC to communicate.
- Clusters help when multiple servers are required to accept HTTP requests through a single port.
- The processes have separate memory because of being spanned in different CPUs leading to memory issues.



Worker Threads:

- Only a single process is there with multiple threads.
- Each Node contains one Node having most APIs accessible.
- The memory is shared with other threads.
- We can use this for CPU-intensive tasks.



Let's go through the top 10 Frequently Asked Express JS Interview Questions and Answers (FAQs)

Most Common Express JS FAQs

1. List down major companies that use Express JS.

The following major companies use Express JS-

- Twitter
- Stack
- Accenture
- Intuit
- Bepro Company
- Trustpilot
- BlaBlaCar.

2. What are the popular alternatives to Express JS?

React JS, Meteor, Mean, Flask, Catalyst, Django, Apache Flex, and Laravel are some of the popular alternatives to Express JS.

3. Enlist some distinct features of Express JS.

Some distinct features of Express JS are given below-

- We can design single-page, multi-page, and hybrid web apps as well as APIs with Express JS.
- A routing table is defined for performing HTTP operations.
- Middleware can be set up for responding to RESTful/HTTP requests.
- The MVC-like structure enables organizing the web apps into MVC architecture.
- HTML pages can be dynamically rendered on the basis of passing arguments to templates.
- NoSQL, as well as RDBMS databases, are supported by it.
- High performance is delivered due to its super-fast I/O. the performance is adequate because of the thin layer prepared by it.
- Routing gets easy by its robust API.
- It is single-threaded as well as asynchronous.

4. Which major tools integrate with Express JS?

The following popular tools integrate with Express JS:

- Sentry
- Node JS
- Datadog
- Mean
- Nodemon
- Bugsnag
- LoopBack
- Sails JS.

5. Is Express JS a back-end framework or a front-end framework?

Express JS is a back-end framework built on JavaScript. It is the MEAN stack's back-end component. Here, 'M' refers to MongoDB, and it manages the database. 'E' refers to Express, and it manages the back-end. 'A' refers to AngularJS, and it handles the back-end. And 'N' refers to Node.

6. Differentiate between Node JS and Express JS.

Node JS is an open-source platform on which the JavaScript code is executed outside of a browser. It is used by several companies, including Uber, Walmart, Netflix, etc. It is a platform acting as a web server and not a programming language or framework. On the other hand, Express JS is a framework built on Node JS.

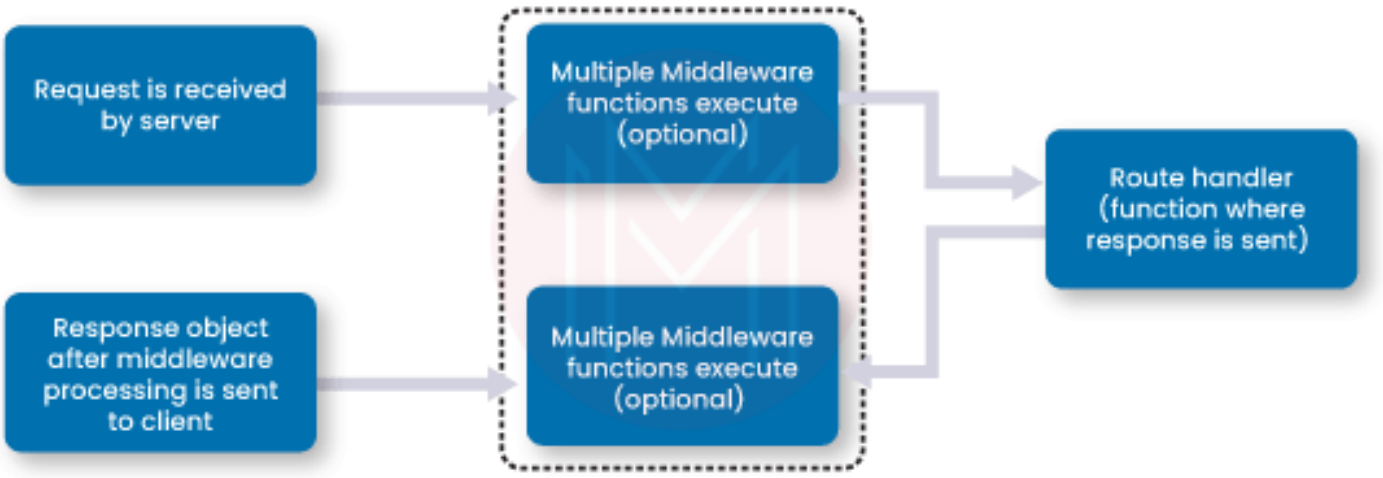
7. Which template engines are supported by Express?

All template engines following the locals, path, and callback signatures are supported by Express JS.

8. What is Middleware, and what are its functions?

Middleware is the function that we invoke before the final request process through the express routing layer. Its functions are given below:

1. Any code like setting headers, validation, etc., can be executed.
2. Changes can be made to the response (res) and request (req) objects.
3. The request-response cycle can also be ended by Middleware.
4. The next middleware function can be called in the stack for proceeding and processing the final request.



9. List the main types of Middleware.

The main types of Middleware are given below:

- Router-level Middleware
- Application-level Middleware
- Built-in Middleware
- Error-handling Middleware
- Third-party Middleware.

10. What is meant by Scaffolding in Express JS? What are the ways to achieve this?

The process of creating the structure of the application is referred to as scaffolding. The two ways of achieving it are given below:

- Express application generator
- Yeoman.

Conclusion

Node JS along with Express JS, allows building a complex and high-level web application from a single line of code easily. That's why it is one of the most sought-after skills in the industry today. And this was a comprehensive list of Express JS Interview Questions that are often asked in interviews. We hope that it will be beneficial for your scheduled interview in the near future.

1. What do you mean by Express JS and what is its use?

Answer:

Express JS is an application framework that is light-weighted node JS. Variety of versatile, helpful and vital options are provided by this **JavaScript** framework for the event of mobile additionally as internet applications with the assistance of node JS.

Express JS Use – Express.js could be a light-weight internet application that helps in organizing the net application into MVC design on the server aspect.

2. Write the steps for setting up an Express JS application?

Answer:

Following are the steps accustomed for An Express JS application:

- A folder with a constant name because the project name is made.
- A file named package.json is made within the folder created.
- "npm install" command is run on the electronic communication. It installs all the libraries gift in package.json.
- A file named server.js is made.
- "Router" file is made within the package that consists of a folder named index.js.
- "App" is made within the package that has the index.html file.

Let us move on to the next Express JS Interview Questions

3. What function are arguments available to Express JS route handlers?

Answer:

The arguments which are available to an Express JS route handler-function are-

- **Req:** the request object
- **Res:** the response object
- **Next (optional):** a function that is employed to pass management to 1 of the following route handlers.

The third argument is optional and should be omitted, however, in some cases, it's helpful wherever there's a series of handlers and management will be passed to 1 of the following route handlers skipping this one.

4. How to Config properties in Express JS?

Answer:

In Express JS, there are 2 ways that for configuring the properties:

With process.ENV:

- A file with the name ".env" is to be created within the project folder.

- All the properties are to be other within the ".env" file.
- Any of the properties will be employed in server.js.

With require.JS:

- A file with the name "config.json" is to be created within the config folder within the project folder.
- The config properties are to be present within the config.json file.
- Now, ought to be accustomed access the config.json file.

5. How Should I Structure my Express JS Application?

Answer:

This is the basic Express JS Interview Questions asked in an interview. There is no definitive answer to the current question. The solution depends on the dimensions of your application and therefore the team that's concerned.

Routes and alternative application-Express logic will board as several files as you would like, in any directory structure you favor. Read the subsequent examples for inspiration:

- Route listings
- Route map

- MVC vogue controllers

Also, there are third-party extensions for Express JS applications that modify a number of these patterns:

- Resourceful routing

6. How to allow CORS in Express JS? Explain with an example?

Answer:

In order to permit CORS in Express.js, add the subsequent code in server.js:

For Example:

```
app.all('*', function(req, res, next) {
res.set('Access-Control-Allow-Origin', '*');
res.set('Access-Control-Allow-Methods', 'GET, POST, DELETE,
PUT');
res.set('Access-Control-Allow-Headers', 'X-Requested-With,
Content-Type');
if ('OPTIONS' == req.method) return res.send(200);
```

```
next();

});
```

7. How to enable debugging in express app?

Answer:

In different operative Systems, we've got following commands:

On UNIX operating system the command would be as follows:

```
$ DEBUG=express:* node index.js
```

On Windows the command would be:

```
set DEBUG=express:* & node index.js
```

From Webstrome IDE

```
C:\Program Files (x86)\JetBrains\WebStorm

2016.2.4\bin\runnerw.exe" "C:\Program Files\nodejs\node.exe"

--debug-brk=61081 --expose_debug_as=v8debug

E:\Development\nodejd\librarey\bin\www
```

Let us move on to the next advanced Express JS Interview Questions.

8. Explain Error Handling In Express.js Using An Example?

Answer:

From Express 4.0 Error handling is easier. The steps are as following:

Create a middleware as following:

```
// error handler

app.use(function(err, req, res, next) solely providing error

in development

res.locals.message = err.message;

res.locals.error = req.app.get('env') === 'development' ?

err : ;

// render the error page

res.status(err.status || 500);

res.render('error');

});
```

Install Error Handler Middleware:

- Install errorhandler

```
npm install errorhandler --save
```

- Create a variable

```
var e errorhandler = require('errorhandler')
```

- Use the middleware as following:

```
if (process.env.NODE_ENV === 'development') solely use in
development
app.use(errorhandler())
}
function errorNotification(err, str, req) power unit title =
'Error in ' + req.method + ' ' + req.url
notifier.notify()
}
```

9. What is the use of next in Express JS?

Answer:

Next -It passes management to a consecutive matching route. OR a operate to pass management to 1 of the following route handlers.

The argument could also be omitted, however, is beneficial in cases wherever you have got a series of handlers and you'd wish to pass management to 1 of the following route handlers, and skip this one.

```
app.get('/user details/:id?', function(req, res, next));
```

- **Req and Res:** It represents the request and response objects
- **Next:** It passes management to a consecutive matching route.

10. How to Redirect 404 Errors to A Page In ExpressJS?

Answer:

In server.js add the subsequent code to send 404 errors back to a page in our ExpressJS App:

```
/* Define fallback route */

app.use(function(req, res, next) {
```

```
res.status(404).json({errorCode: 404, errorMsg: "route not found"});
});
```
