

Explore More

Subscription : Premium CDAC NOTES & MATERIAL @99



Contact to Join
Premium Group



Click to Join
Telegram Group

<CODEWITHARRAY'S/>

For More E-Notes

Join Our Community to stay Updated

TAP ON THE ICONS TO JOIN!

	codewitharrays.in freelance project available to buy contact on 8007592194	
SR.NO	Project NAME	Technology
1	Online E-Learning Platform Hub	React+Springboot+MySQL
2	PG Mates / RoomSharing / Flat Mates	React+Springboot+MySQL
3	Tour and Travel management System	React+Springboot+MySQL
4	Election commition of India (online Voting System)	React+Springboot+MySQL
5	HomeRental Booking System	React+Springboot+MySQL
6	Event Management System	React+Springboot+MySQL
7	Hotel Management System	React+Springboot+MySQL
8	Agriculture web Project	React+Springboot+MySQL
9	AirLine Reservation System / Flight booking System	React+Springboot+MySQL
10	E-commerce web Project	React+Springboot+MySQL
11	Hospital Management System	React+Springboot+MySQL
12	E-RTO Driving licence portal	React+Springboot+MySQL
13	Transpotation Services portal	React+Springboot+MySQL
14	Courier Services Portal / Courier Management System	React+Springboot+MySQL
15	Online Food Delivery Portal	React+Springboot+MySQL
16	Muncipal Corporation Management	React+Springboot+MySQL
17	Gym Management System	React+Springboot+MySQL
18	Bike/Car ental System Portal	React+Springboot+MySQL
19	CharityDonation web project	React+Springboot+MySQL
20	Movie Booking System	React+Springboot+MySQL

freelance_Project available to buy contact on 8007592194

21	Job Portal web project	React+Springboot+MySql
22	LIC Insurance Portal	React+Springboot+MySql
23	Employee Management System	React+Springboot+MySql
24	Payroll Management System	React+Springboot+MySql
25	RealEstate Property Project	React+Springboot+MySql
26	Marriage Hall Booking Project	React+Springboot+MySql
27	Online Student Management portal	React+Springboot+MySql
28	Resturant management System	React+Springboot+MySql
29	Solar Management Project	React+Springboot+MySql
30	OneStepService LinkLabourContractor	React+Springboot+MySql
31	Vehical Service Center Portal	React+Springboot+MySql
32	E-wallet Banking Project	React+Springboot+MySql
33	Blogg Application Project	React+Springboot+MySql
34	Car Parking booking Project	React+Springboot+MySql
35	OLA Cab Booking Portal	React+NextJs+Springboot+MySql
36	Society management Portal	React+Springboot+MySql
37	E-College Portal	React+Springboot+MySql
38	FoodWaste Management Donate System	React+Springboot+MySql
39	Sports Ground Booking	React+Springboot+MySql
40	BloodBank mangement System	React+Springboot+MySql

Project List

41	Bus Tickit Booking Project	React+Springboot+MySQL
42	Fruite Delivery Project	React+Springboot+MySQL
43	Woodworks Bed Shop	React+Springboot+MySQL
44	Online Dairy Product sell Project	React+Springboot+MySQL
45	Online E-Pharma medicine sell Project	React+Springboot+MySQL
46	FarmerMarketplace Web Project	React+Springboot+MySQL
47	Online Cloth Store Project	React+Springboot+MySQL
48	Train Ticket Booking Project	React+Springboot+MySQL
49	Quizz Application Project	JSP+Springboot+MySQL
50	Hotel Room Booking Project	React+Springboot+MySQL
51	Online Crime Reporting Portal Project	React+Springboot+MySQL
52	Online Child Adoption Portal Project	React+Springboot+MySQL
53	online Pizza Delivery System Project	React+Springboot+MySQL
54	Online Social Complaint Portal Project	React+Springboot+MySQL
55	Electric Vehical management system Project	React+Springboot+MySQL
56	Online mess / Tiffin management System Project	React+Springboot+MySQL
57		React+Springboot+MySQL
58		React+Springboot+MySQL
59		React+Springboot+MySQL
60		React+Springboot+MySQL

Spring Boot + React JS + MySQL Project List

Sr.No	Project Name	YouTube Link
1	Online E-Learning Hub Platform Project	https://youtu.be/KMjyBaWmgzg?si=YckHuNzs7eC84-IW
2	PG Mate / Room sharing/Flat sharing	https://youtu.be/4P9clHg3wvk?si=4uEsi0962CG6Xodp
3	Tour and Travel System Project Version 1.0	https://youtu.be/-UHOBywHaP8?si=KHHfE_A0uv725f12
4	Marriage Hall Booking	https://youtu.be/VXz0kZQi5to?si=IiOS-QG3TpAFP5k7
5	Ecommerce Shopping project	https://youtu.be/vJ_C6LkhrZ0?si=YhcBylSErvdn7paq
6	Bike Rental System Project	https://youtu.be/FIzsAmIBCbk?si=7ujQTJqEgkQ8ju2H
7	Multi-Restaurant management system	https://youtu.be/pvV-pM2Jf3s?si=PgvnT-yFc8ktrDxB
8	Hospital management system Project	https://youtu.be/lynLouBZvY4?si=CXzQs3BsRkjKhZCw
9	Municipal Corporation system Project	https://youtu.be/cVMx9NVyl4I?si=qX0oQt-GT-LR_5iF
10	Tour and Travel System Project version 2.0	https://youtu.be/_4u0mB9mHXE?si=gDiAhKBowi2gNUKZ

Sr.No	Project Name	YouTube Link
11	Tour and Travel System Project version 3.0	https://youtu.be/Dm7nOdpasWg?si=P_Lh2gcOFhlyudug
12	Gym Management system Project	https://youtu.be/J8_7Zrkg7ag?si=LcxV51ynfUB7OptX
13	Online Driving License system Project	https://youtu.be/3yRzsMs8TLE?si=JRI_z4FDx4Gmt7fn
14	Online Flight Booking system Project	https://youtu.be/m755rOwdk8U?si=HURvAY2VnizlyJlh
15	Employee management system project	https://youtu.be/ID1iE3W_GRw?si=Y_jv1xV_BljhrD0H
16	Online student school or college portal	https://youtu.be/4A25aEKfei0?si=RoVgZtxMk9TPdQvD
17	Online movie booking system project	https://youtu.be/Lfjv_U74SC4?si=fiDvrhhrjb4KSISm
18	Online Pizza Delivery system project	https://youtu.be/Tp3izreZ458?si=8eWA OzA8SVdNwlyM
19	Online Crime Reporting system Project	https://youtu.be/0UlzReSk9tQ?si=6vN0e70TVY1GOwPO
20	Online Children Adoption Project	https://youtu.be/3T5HC2HKyT4?si=bntP78niYH802i7N

1. What is HTML?

HTML stands for Hypertext Markup Language. It is a language of World Wide Web. The first web page was created at CERN by Tim Berners-Lee on August 6, 1991. You can visit and browse the first website and first web page at the <http://info.cern.ch/> address.

2 . What are the different versions of HTML?

HTML has many versions since its inception during 1990's. HTML versions are HTML, HTML+, HTML 2.0, HTML 3.2, HTML 4.0 and the latest version HTML5.

3 . What are Tags?

In HTML, Tag notify the browser that this can be a HTML document. Tag consider all other HTML elements.

<	= Open delimiter
>	= Closing delimiter
<html>	= Opening tag
</html>	= closing tag
<html>...</html>	= Paired tag
 	= Break tag
<hr>	= horizontal Line
	= Image Tag
<html>	= Opening tag
</html>	= Closing tag
<p></p>	= Paragraph Tag
<h1>...<h6>	= Heading Tags

4 . What is HTML Element?

An HTML element is everything from starting to the ending of HTML tags. For example, <p> This is an HTML Element</p>, in this example, from starting tag to ending tag everything is HTML element and "This is an HTML

Element is Element Content.

5 . What is HTML Attribute?

HTML attribute adds additional information to the HTML Elements. For example, ``, here size and color are html attributes.

6 . What is Hyperlinks?

Hyperlinks are used to navigate to new document with the help of text, links, image or with group of words. Ex : ` FTL `.

7 . What is WWW ?

WWW stands for World Wide Web. It is the inter-connection of all the links called as internet.

8 . What are some common lists ?

There are many common lists which are used to design a page.

Ordered list

Unordered list

Menu list

Directory list

Definition list

9 . What is image map ?

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

10 . Can you create a multi colored text on a web page ?

Yes. To create a multicolor text on a web page you can use `` `` for the specific texts you want to color.

11 . What is a marquee ?

Marquee is used to put the scrolling text on a web page. You should put the text which you want to scroll within the `<marquee>.....</marquee>` tag.

12 . 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 a site.

13 . What bullet types are available ?

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

14 . 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.

15 . What are empty elements ?

HTML elements with no content are called empty elements. For example:
, <hr> etc.

16 . What is the use of span tag? Give one example.

The span tag is used for following :

- . For adding color on text
- . For adding background on text
- . Highlight any color text etc.

Example:

```
<h3><span style="color:#0099da; font-weight:bold;">
```

Free Time Learning

```
</span></h3>
```

17 . How to open a link in new tab or window ?

To open a link in new tab or window, we have to use the following html code :

```
<a href="http://www.freetimelearning.com" target="_blank"> Welcome to freetimelearning.com ...! </a>.
```

18 . What is Domain Name, Domain Space and Browse. ?

Domain Name : Ex: www.google.com(is the doamin name here)

Domain Space : The space required to place our file or website on the world wide web is called Domain Space.

Browser : The Software are the application used to browse the html files is called browser. Ex : Internet Explorer, Mozilla filefox,Google Chrome, Opera, Safari, Netscape Naviagator,Tetra, Epic (First Indian Browser developed at banglore & powered by Mozilla firefox Ex: Epic Browser – Indian Web Browser Epic).

19 . What are empty elements ?

HTML elements with no content are called empty elements. For example:
, <hr> etc.

20 . How to change background color using HTML?

The back ground color of HTML can be change attribute bgcolor in the <body> tag.

Example :

```
<html>
<body bgcolor="#FF00FF">
<h1>Welcome to FreshersEmploy.com</h1>
<p>Hello Job Seekers</p>
</body>
</html>
```

21 . Explain HTML blocks ?

HTML blocks can be defined by HTML tags like and <div>.

22 . How to create frames ?

HTML frames can be created by using <frame> tag within a frameset(<frameset>). HTML <frame> tag defines a particular window within frameset.

23 . What is Semantic HTML ?

Semantic HTML is a coding style where the tags embody what the text is meant to convey. In Semantic HTML, tags like for bold, and for italic should not be used, reason being they just represent formatting, and provide no indication of meaning or structure. The semantically correct thing to do is use and . These tags will have the same bold and italic effects, while demonstrating meaning and structure (emphasis in this case).

24 . 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.

25 . What is a nest in HTML?

If you go and have a look at HTML markup for any webpage today, you will let to know that HTML elements are contained within other HTML elements. These elements which are inside of other are known as nested elements and they had become the essential part of building any web page nowadays. The most expertise way to know more about nesting is just to think about HTML tags in the form of boxes that hold your content which can be in form of text, images,etc.HTML tags are basically the boxes around the content.

26 . What is Cell Spacing and Cell Padding?

The **cellspacing** attribute defines the width of the border, while **cellpadding** represents the distance between cell borders and the content within a cell.

Example:

```
<table border cellspacing=3>
<table border cellpadding=3>
<table border cellspacing=3 cellpadding=3>
```

27 . what is multipart form data?

Multipart form-data is one of the values of the enctype attribute, which is used in the form element that has a file upload. Multi-part means form data divides into multiple parts and send to the server. It adds a few bytes of boundary overhead to the message, and must spend some time calculating it, but sends each byte in one byte.

28 . How many types of lists are available in HTML?

In HTML, we have two types of lists unordered lists and ordered lists. Unordered list starts with `` tag and ends with `` tag. Ordered tag starts with `` and ends with ``. Each list item is written as ``

29 . How can you make a bulleted list in HTML?

An unordered list (``) is used to create a bulleted list in HTML. Below is a simple example to create bulleted list in HTML.

```
<p>Fruits List</p>
<ul>
  <li>Apple</li>
  <li>Banana</li>
  <li>Mango</li>
</ul>
```

30 . What is the difference between DIV and SPAN in HTML?

The difference between span and `<div>` is that a `` element is in-line and usually used for a small chunk of HTML inside a line, such as inside a paragraph. Whereas, a div or division element is block-line which is equivalent to having a line-break before and after it and used to group larger chunks of code.

Example:

```
<div id="HTML">
  These are HTML<span class="blue_color">Interview Questions</span>
</div>
```

31 . Can you create a multi-colored text on a web page?

Yes, we can create a multi-colored text on a web page. To create a multicolor text, you can use `` `` for the specific texts that you want to color.

32 . What is the use of a span tag? Explain with example.

The span tag is used for following things:

For adding color on text

To add background on text

Highlight any color text

Example:

```
<span style="color:#ff00da;">  
    This is sample testing text  
</span>
```

33 . 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. Whereas, 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.

34 . What are the different tags to separate sections of text?

The `
` tag is one way to separate the lines of text. There are other tags like the `<p>` tag and `<blockquote>` tag that are also used to separate sections of text.

35 . What is the advantage of grouping several checkboxes together?

The checkboxes don't affect one another. But, grouping these 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.

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. Similarly, If the border attribute is not included, a default **1-pixel** border appears when the rules attribute is added to the `<table>` tag.

37 . List the media types and formats supported by HTML.

HTML supports a wide range of **media** formats for **sound**, **music**, **videos**, **movies**, and **animations**. Some of the extensions supported by each media format are:

Images : png, jpg, jpeg, gif, apng, svg, bmp, BMP ico, png ico

Audio : MIDI, RealAudio, WMA, AAC, WAV, Ogg, MP3, MP4

Video : MPEG, AVI, WMV, QuickTime, RealVideo, Flash, Ogg, WebM, MPEG-4 or MP4

38 . What are logical and physical tags in HTML?

Logical tags : Logical tags are used to tell the meaning of the enclosed text. The example of the logical tag is ` ` tag. When we enclose the text in the strong tag, it tells the browser that enclosed text is more important than other texts.

Physical tags : Physical tags are used to tell the browser how to display the text enclosed in the physical tag. Some of the examples of physical tags are ``, `<big>`, `<i>`.

39 . What is the primary function of these lists?

The primary function of these lists is to make use of different kinds of **tags** that could be set to compose.

40 . Why, in some cases, the character entities do not display correctly on all systems?

In some cases, the browser running does not support the character, and thus the character is displayed as boxes. This is the condition in which the character entities do not display correctly on all the systems.

41 . What is white space?

White space is the empty sequence of space characters. This white space is considered as a single space character in the HTML.

42 . What is the advantage of white space?

White space helps the browser to *collapse multiple spaces into one single space*, and thereby the indent lines of the text can be taken care of without caring for the multiple spaces that are left. Thereby using the HTML code, white space helps in better organizing the content and tags, making them readable and easy to understand.

43 . How can we create links to different sections within the same web page?

We can create several links to different sections within the same web page by using the `<a>` tag along with referencing through the use of the `#` symbol.

44 . Can HTML files work well on the new browser?

Of course, the HTML files could work very well on the new browser, just that the new browser is compliant to the HTML standards. It may be possible that some new browsers may not support the features of HTML and therefore, won't work well.

45 . Where are the numerical values taken in HTML?

The numerical values are taken from the **ASCII** values for the various characters in the HTML.

46 . How is overlapping affect the sets of tags?

An overlapping set of tags in the HTML results in the recognition of only the first tag. Such issues occur only when the text does not display on the browser screen.

47 . Can style sheets help in aligning images and wrap text?

Yes, the style sheet can apply tables to position text and images for aligning them accordingly.

48 . Is there any limit to a text field size?

Yes, there is a limit of **13 characters** in a text field size. By setting the size attribute, the size value can be set as low as **1**.

49 . Can the text appear outside the browser?

By default, the text cannot appear outside the browser, but, in case the text is part of a table cell with a predefined width, it could extend beyond the browser window.

50 . Write a program to create nested webpages in HTML.

Representing a webpage inside of another webpage is a nested webpage. It is done using the iframe tag which creates an inline frame.

```
<html>
  <body>
    <h2>HTML Iframes example</h2>
    <p>Use the height and width attributes to specify the size of the iframe:</p>
    <iframe src="https://freetimelearning.com/" height="400" width="600"></iframe>
  </body>
</html>
```

51 . Explain the button tag with the help of code.

Button tag lets you create a clickable button in the HTML form on the webpage.

```
<html>
  <body>
    <h2>HTML Button Tag Example</h2>
    <button name="button" type="button">CLICK HERE</button>
  </body>
</html>
```

52 . Explain the script tag.

Script tag can be inside the head or body tag of the HTML code. This tag is executed when the browser reaches that part of the document.

```
<html>
  <body>
    <h2>HTML Script Tag Example</h2>
    <script>
      var x = 5;
      var y = 6;
      var result = x + y;
      alert("X + Y is equal to " + result);
    </script>
  </body>
</html>
```

53 . Why we use the Embed tag?

Embed Tag is used for including a Video or an Audio within an HTML Document. The Embed Tag requires a Closing Tag. It requires the Source of the Video or the Audio file that needs to be displayed onto the Page.

Syntax: <EMBED> Source File </EMBED>

54 . Document Style Semantics And Specification Language (dsssl)?

Document Style Semantics and Specification Language is an international standard, an expression language, a styling language for associating processing (formatting and transformation) with SGML documents, for example XML.

55 . Do Url's Have Quotes Or Not?

Double or single quotes in URLs are optional. The tree following examples are equally valid:

```
body {background: url(imgs/ft1_logo.png) blue}
body {background: url("imgs/ft1_logo.png") blue}
body {background: url('imgs/ft1_logo.png') blue}
```

56 . What Is The Percentage Value In 'font-size' Relative To?

It is relative to the parent element's font-size. For example, if the style sheet says:

```
H1 {font-size: 20pt;}
SUP {font-size: 80%;}
```

57 . Can I Attach More Than One Declaration To A Selector?

Yes. If more than one declaration is attached to a selector they must appear in a semi colon separated list, e.g.;

```
Selector {declaration1; declaration2}  
P {background: white; color: black}
```

58 . How Can I Include Comments In Html?

Technically, since HTML is an **SGML** application, **HTML** uses **SGML** comment syntax. However, the full syntax is complex, and browsers don't support it in its entirety anyway. Therefore, use the following simplified rule to create HTML comments that both have valid syntax and work in browsers:

An HTML comment begins with "<!--", ends with "-->", and does not contain "--" or ">" anywhere in the comment. The following are examples of HTML comments:

```
* <!-- This is a comment. -->  
* <!-- This is another comment,  
and it continues onto a second line. -->  
* <!---->
```

59 . How Can I Eliminate The Extra Space After A Tag?

HTML has no mechanism to control this. However, with CSS, you can set the margin-bottom of the form to 0.

For example :

```
<form style="margin-bottom:0;" action=...>
```

You can also use a CSS style sheet to affect all the forms on a page:
form { margin-bottom: 0 ; }

60 . How Do I Link An Image To Something?

Just use the image as the link content, like this:

```
<a href=...><img src=... alt=...></a>
```

1. What is HTML5 ?

HTML5 (No space between "HTML" and "5") is a markup language used for structuring and presenting content on the World Wide Web. It was published in **28 October 2014**, W3C also published an HTML 5.1 Candidate Recommendation on **21 June 2016**.

2 . Which browsers support HTML5 ?

The latest versions of Apple Safari, Google Chrome, Mozilla Firefox, and Opera all support many HTML5 features and Internet Explorer 9.0 will also have support for some HTML5 functionality.

The mobile web browsers that come pre-installed on iPhones, iPads, and Android phones all have excellent support for HTML5.

codewitharrays.in 8007592194

3 . What are the audio tags provided by HTML5 ?

HTML5 provides following audio tags :

<audio> - Defines sound content
<video> - Defines video content
<source> - Defines multiple media resources for media elements, such as <video> and <audio>

4 . 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.

5 . What are the new FORM elements which is 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 used 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.

6 . 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 allows the user to select a month and year
5. **email** : These input fields used to contain an e-mail address.

7 . What are the Drawbacks of Cookies ?

Some drawbacks in **cookies** there are following :

- Cookies are included with every HTTP request, thereby slowing down your web application by transmitting the same data.
- Cookies are included with every HTTP request, thereby sending data unencrypted over the internet.
- Cookies are limited to about 4 KB of data . Not enough to store required data.

8 . What is WHATWG html5 ?

The Web Hypertext Application Technology Working Group (**WHATWG**) is a community of people interested in evolving HTML and related technologies. The **WHATWG** was founded by individuals from Apple, the Mozilla Foundation and Opera Software in 2004.

9 . What is DOM in html5 ?

The DOM is a W3C (World Wide Web Consortium) standard. The DOM defines a standard for accessing documents: "The W3C Document Object Model (DOM) is a platform and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure, and style of a document."

10 . What is Server Side Events in HTML5 ?

Along with HTML5, WHATWG Web Applications 1.0 introduces events which flow from web server to the web browsers and they are called Server-Sent Events (SSE). Using SSE you can push DOM (Document Object Model) events continuously from your web server to the visitor's browser.

The event streaming approach opens a persistent connection to the server, sending data to the client when new information is available, eliminating the need for continuous polling.

Server-sent events standardizes how we stream data from the server to the client.

11 . What are Waves in HTML ?

A sine wave is a mathematical function that repeats at a regular interval of time. The function is used in many fields including mathematics, physics, and engineering. We can also say that a sine wave is a smooth wave.

It has the following properties :

1. The sine wave is blue whenever the value is positive.
2. The sine wave is red whenever the value is red.
3. The thickness of the line is directly proportional to the absolute value of the magnitude of the wave. For example, where the sine value reaches 0, the wave is absent.

On the X-axis, we will map the angle Theta. Theta will vary from 0 degree to 1040 degrees.

On the Y-axis, we will map the sin (Theta). For this, we will use the Math function Math.sin. The Math.sin function takes angles in radians. So the angle is first multiplied by PI / 180.

12 . What is HTML5 Drag and Drops ?

At every stage of the drag and drop operation a different event is fired so that the browser knows what JavaScript code to execute; the events are:

- dragStart** :- fires when the user starts dragging the element.
dragEnter :- fires when the draggable element is first dragged over the target element.
dragOver :- fires when the mouse is moved over an element when the drag is occurring.
dragLeave :- fired if the user's cursor leaves an element when dragging.
drag :- fires every time we move the mouse during the dragging of our element.
drop :- fired when the actual drop is performed.
dragEnd :- fires when the user releases the mouse while dragging the object.

13 . What is Details Tag in HTML5 ?

The HTML <details> tag specifies additional details that the user can view or hide on demand.

The <details> tag can be used to create an interactive widget that the user can open and close. Any sort of content can be put inside the <details> tag.

14 . What is Dialog Tag in HTML5 ?

The HTML <dialog> tag indicates a part of an application that the user can interact with. Examples of dialog could include a dialog box, inspector, or window.

The <dialog> element accepts a boolean attribute called open that sets the element to "active" and allows users to interact with it.

15 . What is Progress Tag in HTML5 ?

The PROGRESS Element is used to show the completion or progress of a specific task.

We often see on the website, while downloading some text or video, music files and even on uploading some contents on the Internet, a progress bar comes before us and displays its current progress or completion of specific tasks.

16 . What are the various tags provided for better structuring in HTML5 ?

The various tags provided for better structuring in HTML 5 are:

- <article> - This tag defines an article.
- <aside> - It defines content other than the page content.
- <bdi> - This tag isolates a part of text for formatting in a different direction from other text.
- <command> - It defines a command button to be invoked by the user.
- <details> - It defines additional details that can be viewed or hidden by the user.
- <dialog> - It defines a dialog box.
- <figure> - This tag specifies content like illustrations, diagrams, photos, code listings, etc.
- <figcaption> - It is used to provide a caption for a <figure> element
- <footer> - This tag defines a footer for a document or section
- <header> - This tag is used to define a header for a document or section
- <hgroup> - When there are multiple levels in a heading, it groups a set of <h1> to <h6> elements.

17 . What is SVG ?

- a. SVG is the abbreviation for Scalable Vector Graphics and is recommended by W3C.
- b. It is used to define vector-based graphics for the Web. All the element and attributes of SVG files can be animated.
- c. The graphics are defined in XML format.
- d. An important quality of SVG graphics is that their quality is maintained even when they are zoomed or resized.

18 . What is Geolocation API in HTML5 ?

The geolocation API allows you to easily fetch the geographical position of a user. If the browser support Geolocation API, one can easily get the Longitude and Latitude just using Javascript. The Geolocation API uses multiple sources to determine the most accurate position of the user. The location accuray depnads upon the best location source available.

19 . What is image map in html5?

Image maps are a combination of URL and images, where clicking on these images (clickable area of the image) will open different new web pages.

Two types of image maps are available in HTML5, i.e. **client side image map** and **server side image map** :

The **client-side image map** is created by using two elements `<area>` and `<map>`, where the map holds the map information and the area element takes the attributes to define each section of the map.

Server-side image map created by using `<usemap>` attribute, the `usemap` attribute is the name of our map.

20 . How to optimize website assets?

We need to understand some basic optimization rules, in order to optimize website assets. Initially, we should decrease the download size and make fewer http requests.

To optimize website assets we can follow the below techniques :

- * File compression
- * File concatenation
- * CDN Hosting
- * Offloading assets
- * Re-organizing
- * Refining code

21 . What is the use of MathML Element in HTML5?

The word MathML (Mathematical Markup Language) is a markup language, that is used to show scientific and mathematical expression on the web. MathML is a form of XML (extensible markup language) to describe the Math notation.

We can use `$...$` tags inside the HTML5 documents for implementing MathML element.

Example: print $a^2+2b+5=0$ using HTML5 code.

22 . Why do we use HTML5?

HTML5 supports animation, drawing, audio, video, etc and it easily embeds a video on the web page. It does not require any additional software like Flash for watching videos.

Some of the important reasons to use HTML5 are given below :

- * Legacy and cross-browser support
- * Better interactions
- * Smarter storage
- * Cleaner code

23 . Explain HTML5 Graphics.

HTML5 supports two types of graphics i.e. Canvas and SVG.

a) **Canvas** : <canvas> element is used to design graphics on the web page, and it has several methods available for drawing circles, boxes, adding images and text. 300 px X 150 px (width X height) is the default pixel size of canvas.

Example to Draw square Box using canvas element is shown below.

```
<!DOCTYPE HTML>
<html>
  <head>
    <style>
      #mycanvas{border:2px solid pink;}
    </style>
  </head>
  <body>
    <canvas id = "mycanvas" width = "150" height = "150"></canvas>
  </body>
</html>
```

b) **SVG** : <svg> Scalable Vector Graphics, are mostly used for graphical application and scalable vector type diagrams, such as X, Y coordinate system, two-dimensional and pie charts. This makes it faster and lightweight. SVG follows XML format.

Example to draw a Rectangle using SVG element is shown below.

```
<!DOCTYPE html>
<html>
  <body>
    <svg width = "400" height = "150">
      <rect width = "400" height = "150" style = "fill:rgb(200,220,255);stroke-width:10;stroke:rgb(0,0,0)">
    </svg>
  </body>
</html>
```

24 . How many web browsers does HTML5 support?

Most of the latest versions of Apple Safari, Google Chrome, Opera, Edge and Mozilla Firefox are supported by HTML5.

25 . Which element provides the autocomplete feature in a textbox?

In HTML5 <Datalist> element, provides the autocomplete feature in a textbox.

26 . What is Microdata in HTML5?

Microdata is a new simple semantic syntax, that is used to add the nested groups of name and value pair of data to documents, that are commonly based on the page content. Microdata is used for new global attributes.

27 . What is an API and list the API's available in HTML5?

API (Application Programming Interfaces) in HTML5 is a way to build different applications using pre-built components. Developers can integrate the relevant features of existing APIs into their new websites.

The APIs in HTML5 are :

- * HTML Web Workers
- * HTML Drag and Drop
- * HTML Application Cache
- * HTML Local Storage
- * HTML SSE
- * HTML Geolocation

28 . How do you optimize a website's assets & reduce page load time?

Follow these steps to optimize assets and also reduce the page load time of any website :

- * Disabling etags.
- * Properly naming all assets.
- * Using CSS Sprites for Images.
- * Minimizing the use of JS in code.
- * Using a content delivery network (CDN) for media files.
- * Minimizing inline CSS and using internal and external style sheets.
- * Hosting assets on different domains while also reducing DNS lookups
- * Using a cookie-free domain to place assets and splitting them among domains

29 . What is the difference between the SVG and Canvas?

Canvas	SVG
Resolution dependent	Resolution independent
No support for event handlers	Provides support for event handlers
Suitable for graphic-intensive	Ideal for creating graphics similar to CAD software
Works best for small rendering areas	Works even for large rendering areas

30 . What are the different types of storage in html5? Explain

In HTML5, data can be stored in 2 ways – *session storage* and *local storage*.

Session Storage : The data or details from the current browsing session are stored. Once the browser gets closer by the user, the storage data gets removed.

Local Storage : The data does not get cleared automatically when the user closes the browser.

31 . What is a Web Worker in Html5?

A web worker in HTML5 is a script that runs on a separate thread in the background without the need for the main web page to wait for it to complete. Web Workers enable long tasks to be achieved without keeping the web page responsive.

32 . What does WHATWG stand for in the world of HTML5?

The WHATWG stands for **Web Hypertext Application Technology Working Group** in HTML5. This is an organization which both develops and maintains the HTML and API for web applications.

33 . What is Meta tags? Why it is used?

HTML5 has introduced a way to let designers take charge of the web page area that is visible to the users through tags. The meta tag tells information related to an HTML document. Some of the examples are **Link**, **Style** and **Title**. With meta tag, we can add **meta keywords**, **meta description** and **meta title** that helps for **SEO Purposes**.

34 . Can we use multiple <header> and <footer> elements in html5?

Yes. Both <header> and <footer> tags are made to fulfill their respective purposes in relation to their parent "section". So not only can you use both header and footer elements on a page in HTML5, but you can also use both in every <article> or <section>. A <header> tag must be present for all articles, sections, and pages, although a <footer> tag is not necessary.

35 . What is a manifest file in HTML?

In HTML 5, the manifest file is a normal text file used for listing resources in the browser which are to be cached for offline access. It also helps in loading resources faster and reducing server load time.

36 . Please explain how to indicate the character set being used by a document in HTML5?

The character set being used by an HTML5 document is indicated using the charset attribute of a <meta> tag inside the <head> element of the HTML5 document :

```
<!DOCTYPE html>
<html>
  <head>
    ...
    ...
    <meta charset="UTF-8">
    ...
    ...
  </head>
  <body>
    -----
    -----
  </body>
</html>
```

37 . What are the useful API in HTML5?

Some of the useful APIs in HTML5 are :

Fetch : Fetch has made Http requests easier than it was with XMLHttpRequest.

Battery status : checks the battery status of the device.

Geolocation : tells the device location.

Clipboard : copy the contents to the clipboard.

Forms : new types have been added for validation and rendering.

Drag and drop : easily drag and drop items in the app.

Screen orientation : Checks the device's screen orientation.

Web audio : Process audio on the client-side.

Internalization : International formatting and string comparison.

Web sockets : real-time communication between server and client.

38 . What is the use of URL input control in Web form 2.0?

This control accepts only URL value and is used for input fields that need a URL address. It will not accept any other values like text input. The accepted formats are : http://www.<webaddr> or http://<webaddr>

39 . What is the server-sent events in HTML5?

The events that flow from the webserver to the browsers are called server-sent events. DOM events can be continuously pushed from the webserver to the browser of a user. To use a server-sent event (**SSE**), you should add an `<eventsource>` element to the document. The `src` attribute of this element describes the **HTTP** connection location, which sends a data stream having the events.

Example : `<eventsource src = "/cgi-bin/myfile.cgi" />`

40 . What is Vibration API in HTML5?

Vibration is a simple, nice way of alert when you get a new message or a phone call. It is especially useful when you are in a noisy environment or the place where you feel the ringing would be a distraction to others.

It is interesting to know that HTML5 is now providing us to play with the vibration on the devices but the HTML5 Vibrate API supports only the recent version of Firefox & Chrome.

To check the vibration, API support in browsers as shown below,

```
navigator.vibrate = navigator.vibrate || navigator.mozVibrate ||
navigator.webkitVibrate || navigator.msVibrate;
if (navigator.vibrate) {
    // supports vibration API.
}
```

Vibration Syntax : Vibration basic syntax is,

```
navigator.vibrate(long | [long]);
```

The vibrate function accepts milliseconds or an array of milliseconds.

Example

```
// vibrate for 1000 ms
navigator.vibrate(1000);
// same like above but in array of ms
navigator.vibrate([1000]);
```

In the above examples, we are setting the device to vibrate 1000 milliseconds.

Codewitharrays.in 8007592194

1. What is CSS ?

CSS is the acronym for "Cascading Style Sheet". CSS is the language for describing the presentation of Web pages, including colors, layout, and fonts. Css are three types **Class base css**, **Tag base css**, **ID base css**.

2 . What are the different variations of CSS ?

Following are the different variations of CSS :

CSS1
CSS2
CSS2.1
CSS3
CSS4

3 . Explain the advantages of CSS ?

Below are some of the advantages of CSS :

1. Accessibility
2. Multiple Device Compatibility
3. Page will load fast
4. Maintenance is Easy
5. Offline Browsing

4 . What is CSS selectors ?

In CSS selectors are patterns used to select the element's you want to style. Use our CSS Selector Tester to demonstrate the different selectors.

There are several different types of selectors in CSS.

1. CSS Class Selector
2. CSS Id Selector

- 3. CSS Universal Selector
- 4. CSS Element Selector
- 5. CSS Group Selectors and more selectors etc...

5 . How can you integrate CSS on a web page ?

There are three methods to integrate CSS on web pages.

- 1. Inline method (or) CSS
- 2. Internal method (or) CSS
- 3. External method (or) CSS

Inline css : The css which is written in the tags in the body of html documents is called Inline css.

Internal css : The css which is written internally within the html document in the head tag is called Internal css.

External css : The css which is written externally and it is linked within head tag is called External css.

6 . What is the difference between Class base CSS, Tag base CSS, ID base CSS ?

Class Base : The css used to redefine the elements of html is called as class base css. It is denoted by dot(.) before the name. The class based css file is using to multiple times of single document.
ex: (.sample{color:#000;})

Tag Base : The css used to apply for tag of html is called as tag base css. Tag base css is denoted with tags(body,img, ul, li,p,table,h1 to h6..etc)

ID Base : The css used to create block elements is called as ID base css. Alternative for creating table(cell) without html. It is denoted by (#)before the name. The ID based css file is using to single time of single document.
ex: #header,#banner etc.

7 . Explain CSS Padding and Margin.

Padding : Padding is used to define the space between the content and the border. (Inside Border)

Margin : Margin is the space outside the border. (Outside Border)

You can easily specify the different paddings and margins for the different sides of an element such as top, right, bottom or left side using the CSS individual padding and margin properties.

8 . What is CSS Box Model ?

The CSS box model is essentially a box that wraps around every HTML element. It consists of: margins, borders, padding, and the actual content.

9 . Explain child selector in CSS ?

Child selectors can be used for applying the style for parent element and this will descend to the child elements. Below is the example -

```
body > input{  
color: #FFFFFF;  
}
```

10 . How to combine multiple style sheets into single page ?

Multiple style sheets can be combined by using the `<link>` tag and the `title` attribute. The title value allows one or more `<link>` tags to link with each other. After combination that theme will be applied as combined and will be shown to the user.

The syntax of it will be as follows :

```
<link rel= "stylesheet" href="bootstrap.css" title="combined">  
<link rel= "stylesheet" href="style.css" title="combined">  
<link rel= "stylesheet" href="slider.css" title="combined">
```

Another way to combine the style sheets is the use of import which can be used in the `<style>` tag and the syntax can be given as follows:

```
@import url(site_url);
```

11 . What is the purpose of pseudo-elements?

Pseudo elements allow the use of the part of element and not the element itself. They are applied to block level element, which is used for the complete block for which the CSS is being written. This allow the subpart of an element to be styled like paragraphs and headings.

For example:

```
selector:pseudo-element {property:value;}  
p: first-line {text-transform: lowercase;}
```

12 . What is the difference between CSS2 and CSS3 ?

The main difference between CSS2 and CSS3 is that CSS3 is divided into different sections which is also known as modules.

Unlike CSS2, CSS3 modules are supported by many browsers.

13 . What is Pseudo-elements ?

Pseudo-elements are used to add special effects to some selectors. CSS is 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.

14 . 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

15 . What is 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.

16 . How comments can be added in CSS ?

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

17 . What are quotes mandatory in URLâ€™s ?

Quotes are optional in URLâ€™s, and it can be single(' ') or double.(" ").

18 . Explain the concept of Tweening.

Tweening is the process in which we create intermediate frames between two images to get the appearance of the first image which develops into the second image.

It is mainly used for creating animation.

19 . What are CSS counters?

CSS counters are variables that can be incremented by rules of CSS that inspector track how many times the variable has been used.

20 . What are gradients in CSS?

It is a property of CSS which allows you to display a smooth transformation between two or more than two specified colors.

There are two types of gradients that are present in CSS. They are:

- * Linear Gradient
- * Radial Gradient

21 . Write all the properties of the flexbox.

There are several properties of the flexbox that are used in the HTML webpage.

They are :

- * flex-direction
- * flex-wrap
- * flex-flow
- * justify-content
- * align-items
- * align-content

22 . What is the use of the Box Model in CSS?

In CSS, the box model is a box that binds all the HTML elements and it includes features like margins, border, padding, and the actual content.

By using a box model we will get the authority to add the borders all around the elements and we can also define the space between the elements.

23 . How can we add icons to the web page?

We can add icons to the HTML webpage by using an icon library like font-awesome.

We have to add the name of the given icon class to any inline HTML element. (*<i>* or **) . Icons in the icon libraries are scalable vectors that can be customized with CSS.

24 . What are navigation bars in CSS?

By using navigation bars we can make an ordinary HTML page into a user-specific and more dynamic web page. Basically, it is a list of links, hence use of ** and ** elements makes the perfect sense.

```
ul {  
    list-style-type: none;  
    margin: 0;  
    padding: 0;  
}
```

25 . Define 'important' declarations used in CSS.

Important declarations are defined as that declaration which is having more importance than the normal declaration.

While executing, these declarations override the declaration which is having less importance.

For example, if there are two users having an important declaration then one of the declarations will override the declaration of another user.

For Example :

```
Body {background: #FF00FF !important; color: blue}
```

In this body, background has more weight than the color.

26 . Differentiate between inline and block element.

Inline element does not have an element to set width and height and also it does not have the line break.

Example : em, strong, etc.

Block element specification:

They do have the line break.

They define the width by setting a container and also allow setting height.

It can also contain an element that occurs in the inline element.

Example :

```
width and height  
max-width and max-height  
min-width and min-height  
hi (i=1-6)- heading element  
p- Paragraph element.
```

27 . What are the benefits of CSS sprites?

Benefits of using CSS sprites are : It is a technique where one has a large image containing a set of small images. Those images can be broken down with the help of CSS to disintegrate into multiple images.

It helps large images or pages to load faster hence, saving a lot of time. It cuts back HTTP requests. It is also flexible over website's layout and design.

The concept of CSS sprites is used to reduce the loading time for a web page because it combines the various small images into one image. It reduces the number of http requests and hence the loading time.

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 . 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 #.

Ex : g {color: #00cjfi}

- * A mixture of red, green and blue represents a color. The value of a color can also be specified.

Example : 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.

30 . Enlist the various fonts' attributes?

They are :

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

31 . Comment on the Case-sensitivity of CSS ?

Basically it is not case sensitive but the class names are considered as case sensitive in HTML 4.01 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.

32 . Describe 'rule set'?

It is an instruction that tells browser on how to render a specific element on the HTML page. It consists of a selector with a declaration block that follows.

Rule set : Rule set Selectors can be attached to other selectors to be identified by rule set.

33 . What are the disadvantages of CSS?

The disadvantages of using CSS are :

There are no expressions.

Vertical control is limited.

The column declaration is not there.

Target specific texts, styling, and rules are not allowed.

There is no dynamic behavior to control pseudo-class.

It is not possible to ascend by selectors.

34 . What is meant by CSS frameworks?

CSS frameworks are pre-planned libraries, allowing a more convenient web page styling which is also compliant to standards.

35 . How to center block elements using CSS1?

In order to center the *block-level elements*, we need to set the margin-right and margin-left properties to explicit values.

36 . What are the ways to integrate CSS as a web page?

There are three methods to integrate CSS in the form of the web page:

Embedded : There can be a style element inside the head element inside which we can place the code.

Inline : CSS applied to HTML elements can be found using style attributes.

Imported or Linked : In case of linked or imported CSS, the CSS is placed in an external file and link element is used to link it.

37 . What is a declaration block in CSS?

It is basically a catalog of directions comprising the property, followed by a colon, and finally the value enclosed within braces.

38 . What is RWD?

RWD is the abbreviated form of **Responsive Web Design**. It is used to display the designed page suitably on any kind of screen size depending on the device under consideration.

39 . List out any 5 properties of cursor in CSS?

Below are the list of properties of cursor :

- * Pointer
- * Help
- * Wait
- * Hand
- * Crosshair

40 . What is the difference between “display:none” and “visibility:hidden” in CSS?

Display:none : This will just hide the element and does not take any space of the element.

Visibility:hidden : This also hides the element and will take space for the element and this will affect the entire layout of the document.

41 . What do you know about graceful degradation?

Graceful degradation is about creating a website design that will be compatible with the latest browsers and, at the same time, will remain functional to an extent on smaller screens and older browsers. I use the principle of graceful degradation in my web development work, as many of our viewers still use older browsers and it is important that they should be able to access and view our website.

42 . What is file splitting, and what is its benefit?

File splitting is about splitting the CSS code into many separate files and organizing them according to the component type or site section to facilitate better management of CSS. You can create and assign a separate style sheet for each of these files for further convenience. So, when someone visits a section of the website, only the style sheets for that section will load. The style sheets for the other website sections will not load until specifically needed. I've found that this reduces the time taken for the page to load on the browser. That then leads to quicker access to websites and enhances user experience.

43 . What is the use of opacity in CSS?

The opacity is mainly used in defining the transparency of the element. In other words, it specifies the clarity of an image by allowing some fixed size of light to pass through it.

Example :

```
<style>
  img{
    height:250px;
    width:250px;
    opacity:50px;
  }
</style>
```

44 . Which property is used for controlling image repetition in the background?

The background-repeat property is responsible for controlling the image repetition. Using this property, we can repeat an image horizontally, vertically, or both. It has the following different values:

repeat: it repeats the image horizontally and vertically

repeat-x: it repeats the image horizontally

repeat-y: it repeats the image vertically

no-repeat: it does not repeat the image.

Example :

```
<style>
  div {
    background-image:url(freetimelearn_logo.jpeg);
    background-repeat:no-repeat;
  }
</style>
```

45 . How we can override CSS?

With the help of !important we can override CSS properties.

Example :

```
.left_side{
  width:30%;
}

.left_side{
  width:40% !important;
}
```

46 . List some CSS frameworks?

- * Bootstrap
- * Semantic
- * Materialize
- * Bulma
- * Foundation
- * Pure CSS etc

47 . What is float in CSS?

The float property in CSS places an element on the right or left side of its container, allowing text and inline elements to wrap around it.

Example :

```
.right_side{
  float: right;
  width:30%;
}
```

48 . How to make scroll bars with the help of CSS?

The CSS overflow property specifies whether to clip content or to add scrollbars when the content of an element is too big to fit in a specified area.

Property of overflow is given below :

```
* visible  
* hidden  
* auto  
* scroll
```

49 . What is CSS unicode-bidi Property?

The unicode-bidi property is used together with the direction property to set or return whether the text should be overridden to support multiple languages in the same document.

CSS unicode-bidi property example :

```
div {  
    direction: rtl;  
    unicode-bidi: bidi-override;  
}
```

50 . What is CSS text-indent Property?

The text-indent property specifies the indentation of the first line in a text-block.

Example :

```
p {  
    text-indent: 50px;  
}
```

51 . What is CSS transform-origin Property?

The transform-origin property allows you to change the position of transformed elements. 2D transformations can change the x- and y-axis of an element. 3D transformations can also change the z-axis of an element.

Example :

```
div {  
    transform: rotate(45deg);  
    transform-origin: 20% 40%;  
}
```

52 . What is CSS hanging-punctuation Property?

The **hanging-punctuation** property specifies whether a punctuation mark may be placed outside the line box at the start or at the end of a full line of text.

Example :

```
p {  
    hanging-punctuation: first;  
}
```

53 . What is CSS counter-increment Property?

The **counter-increment** property increases or decreases the value of one or more CSS counters. The counter-increment property is usually used together with the counter-reset property and the content property.

Example :

```
body {  
    /* Set "my-sec-counter" to 0 */  
    counter-reset: my-sec-counter;  
}  
  
h2:before {  
    /* Increment "my-sec-counter" by 1 */  
    counter-increment: my-sec-counter;  
    content: "Section " counter(my-sec-counter) ". "  
}
```

54 . What is CSS backface-visibility Property?

The **backface-visibility** property defines whether or not the back face of an element should be visible when facing the user. The back face of an element is a mirror image of the front face being displayed. This property is useful when an element is rotated.

Example :

```
#div1 {  
    backface-visibility: hidden;  
}  
#div2 {  
    backface-visibility: visible;  
}
```

55 . What is CSS calc() function?

The **calc()** function performs a calculation to be used as the property value.

Example:

```
#div1 {  
    position: absolute;  
    left: 50px;  
    width: calc(100% - 100px);  
    border: 1px solid black;  
    background-color: yellow;  
    padding: 5px;  
    text-align: center;  
}
```

56 . What is the cubic-bezier() function?

The **cubic-bezier()** function defines a Cubic Bezier curve.

Example :

```
div {  
    width: 150px;  
    height: 150px;  
    background: blue;  
    transition: width 2s;  
    transition-timing-function: cubic-bezier(0.1, 0.7, 1.0, 0.1);  
}
```

1. What is CSS3 ?

The first working draft of CSS3 came in 19-01-2001. but it was initially declared early in the June 1999. The main difference between css2 and css3 is Media Queries, Namespaces, Selectors Level 3, Color.

2 . Explain some CSS3 Modules.

CSS3 has been split into "modules". It contains the "old CSS specification". In addition, new modules are added.

Ex : Selectors, Box Model, Backgrounds, Borders, Gradients, shadows, Image Values and Replaced Content, Text Effects, 2D Transformations, 3D Transformations, Transitions, Animations, Multiple Column Layout, User Interface, etc,..

3 . What new futures added in CSS3 for Borders and how Browser Support it ?

3 following border futures added :

border-radius
box-shadow
border-image

4 . What is the word wrap / word wrapping in CSS3 ?

To Allow long words to be able to break and wrap onto the next line in css3 we used word-wrap property like below class

```
.wordwrap{word-wrap:break-word;}
```

5 . What is the CSS3 animation ?

When the animation is created in the **@keyframe**, bind it to a selector, otherwise the animation will have no effect. Bind the animation to a selector by specifying at least these two CSS3 animation properties :
Specify the name of the animation
Specify the duration of the animation

6 . What do we mean by Opacity in CSS3 ?

Opacity is a style property used to show or hide the html element.

For example 0 for hide and 1 for show.

```
<p style="opacity: 0">Show Me</p>
<p style="opacity: 0.5">Show Me</p>
<p style="opacity: 1">Show Me</p>
```

The Opacity declaration sets how opaque an element is. An opacity value of 1 means the element is fully opaque. An opacity value of 0 means an element is not at all opaque, i.e. fully transparent. This element's opacity is 0.5 .

Note : That both the text and the background-color are affected by the opacity level.

7 . How is multiple background images handled in CSS3 ?

If you want to use the multiple background ,we can insert as follows...

background-image : url(image.png), url(image-2.png), url(image-3.jpg); also we can mention the position of the image either percentage value or top left, left center like this.

8 . How CSS3 is related to responsive web designing?

Media queries are introduced in CSS3, which are directly used for making website responsive i.e. **RWD** (Responsive Web Designing). Get more guidance on Responsive Web Design Tutorials with step by step guidance!

9 . Describe about new CSS3 property â€˜Flexâ€™ in few words.

Flexbox is the new layout model being used in CSS3. Used along with display, its primary function is to control the elements behaviour in multiple screen sizes and devices.

10 . What is CSS3 multiple column property ?

CSS3 multiple Column property is easiest way of dividing section of particular element just like in newspaper and there are various rule applied to it regarding count, gap, rule, style, width, color and span.

11 . What are CSS3 transitions ?

CSS3 transitions are the properties which lets change effect using CSS smoothly over specified duration. Multiple changes simultaneously are allowed within one section.

12 . Explain the use of media queries in CSS3.!

Media queries are the latest web standard for crafting adaptable style sheets for tablets, Desktop, mobiles and iOS enabled devices. They are applied using height, width of the device, viewport, orientation and resolution. So any structure design can be altered for making design responsive.

13 . How many CSS3 User Interfaces.!

CSS3 has introduced mainly Two types of user interface that has been described as follows:

1. resize : The resize is a such property of User Interface, by which you can resize your div layout on your browser.
Three features of resize you can use

- i. resize:both
- ii. resize:vertical
- iii. resize:horizontal.

2. outline-offset : The outline-offset is such property of the CSS3 User Interface, which can be used to define the space between the element's border and its outline.

14 . What is CSS3 3D Transforms ?

CSS3 3D transform feature you can perform basic transform manipulations such as move, rotate, scale and skew on elements in a three-dimensional space.

The following 3D transformation methods:

- 1. rotateX()
- 2. rotateY()
- 3. rotateZ()

15 . Explain all the measurement units in CSS3?

% - Defines a measurement as a percentage relative to another value, typically an enclosing element.
cm - Defines a measurement in centimeters. `div {margin-bottom: 2cm;}`

em - A relative measurement for the height of a font in em spaces. Because an em unit is equivalent to the size of a given font, if you assign a font to 12pt, each "em" unit would be 12pt; thus, 2em would be 24pt.

ex - This value defines a measurement relative to a font's x-height. The x-height is determined by the height of the font's lowercase letter x.

in - Defines a measurement in inches.

mm - Defines a measurement in millimeters.

pc - Defines a measurement in picas. A pica is equivalent to 12 points; thus, there are 6 picas per inch.

pt - Defines a measurement in points. A point is defined as 1/72nd of an inch.

px - Defines a measurement in screen pixels.

16 . What is CSS3 Flexbox?

The Flexible Box Module, usually referred to as flexbox, was designed as a one-dimensional layout model, and as a method that could offer space distribution between items in an interface and powerful alignment capabilities.

An area of a document laid out using flexbox is called a flex container. To create a flex container, we set the value of the area's container's display property to flex or inline-flex. As soon as we do this the direct children of that

container become flex items. As with all properties in CSS, some initial values are defined, so when creating a flex container all of the contained flex items will behave in the following way.

- * Items display in a row (the flex-direction property's default is row).
- * The items start from the start edge of the main axis.
- * The items do not stretch on the main dimension but can shrink.
- * The items will stretch to fill the size of the cross axis.
- * The flex-basis property is set to auto.
- * The flex-wrap property is set to nowrap.

17 . How do you add shadow to elements in CSS3?

```
box-shadow: 10px 10px 5px grey;  
shadow-right: 10px;  
shadow-bottom: 10px;  
shadow-color: grey;  
alpha-effect[shadow]: 10px 10px 5px grey;
```

18 . How to resize a background image using CSS3?

```
background-size: 80px 60px;  
bg-dimensions: 80px 60px;  
background-proportion: 80px 60px;  
alpha-effect: bg-resize 80px 60px;
```

19 . How to rotate objects using CSS3?

```
object-rotation: 30deg;  
transform: rotate(30deg);  
rotate-object: 30deg;  
transform: rotate-30deg-clockwise;
```

20 . How to re-size/scale objects using CSS3?

```
transform: scale(2,4);  
scale-object: 2,4;  
scale: (2,4);
```

21 . Is it possible to create border as a Image in CSS3?

Yes it is possible, by using CSS3 border image property we can use image as a border.

```
border: url(logo.png);  
border: image url(logo.png);  
border-image: url(logo.png) 30 30 round;
```

22 . How can you change the first line of paragraph color using css3?

The::first-line selector is used to add a style to the first line of the specified selector.

```
p::first-line {  
    background-color: yellow;  
}
```

23 . What are the most commonly used Multi Backgroud properties in CSS3?

There are four most commonly used multi background properties, they are:

background-clip : Used to declare the painting area of the background

background-image : Used to specify the background image

background-origin : Used to specify position of the background images

background-size : Used to specify size of the background images

24 . Can you write CSS3 code for creating Multi Background Images?

Here is the CSS3 code for creating multi background images.

```
<style>  
    #multibackgroundimg {  
        background-image: url(imgs/logo.png), url(imgs/sample_logo.png);  
        background-position: left top, left top;  
        background-repeat: no-repeat, repeat;  
        padding: 75px;  
    }  
</style>
```

25 . What HSLA stands for in CSS3?

HSLA stands for hue, saturation, lightness and alpha.

26 . What are the newly introduced Text related features in CSS3?

There are mainly three Text related features are introduced, they are :

* text-overflow

- * text-emphasis
- * text-align-last
- * word-wrap
- * word-break

27 . What are the different web fonts formats in CSS3?

Web fonts allows users to use the fonts in CSS3, which are not installed on local system.

There are five types of web fonts formats are there, they are:

- 1) TTF – TrueType Fonts
- 2) OTF – OpenType Fonts
- 3) WOFF – The Web Open Font Format
- 4) SVG Fonts
- 5) EOT – Embedded OpenType Fonts

28 . What are the common values used in 2D Transforms?

Here are the some commonly used values in 2D Transforms,

matrix(n,n,n,n,n,n) : Used to defines matrix transforms with six values
translate(x,y) : Used to transforms the element along with x-axis and y-axis
translateX(n) : Used to transforms the element along with x-axis
translateY(n) : Used to transforms the element along with y-axis
scale(x,y) : Used to change the width and height of element
scaleX(n) : Used to change the width of element
scaleY(n) : Used to change the height of element
rotate(angle) : Used to rotate the element based on an angle
skewX(angle) : Used to defines skew transforms along with x axis
skewY(angle) : Used to defines skew transforms along with y axis

29 . What are the common values used in 3D Transforms?

Here are the some commonly used values in 3D Transforms,

matrix3d(n,n,n,n,n,n,n,n,n,n,n,n,n,n,n,n) : Used to transforms the element by using 16 values of matrix
translate3d(x,y,z) : Used to transforms the element by using x-axis,y-axis and z-axis
translateX(x) : Used to transforms the element by using x-axis
translateY(y) : Used to transforms the element by using y-axis
translateZ(z) : Used to transforms the element by using z-axis
scaleX(x) : Used to scale transforms the element by using x-axis
scaleY(y) : Used to scale transforms the element by using y-axis
scaleZ(z) : Used to transforms the element by using z-axis
rotateX(angle) : Used to rotate transforms the element by using x-axis
rotateY(angle) : Used to rotate transforms the element by using y-axis
rotateZ(angle) : Used to rotate transforms the element by using z-axis

30 . What are the values associated with multi columns?

Here is the list of most commonly used multi column values, they are:

column-count : Used to count the number of columns that element should be divided
column-fill : Used to decide, how to fill the columns
column-gap : Used to decide the gap between the columns
column-rule : Used to specifies the number of rules
rule-color : Used to specifies the column rule color
rule-style : Used to specifies the style rule for column
rule-width : Used to specifies the width
column-span : Used to specifies the span between columns

31 . Can you write CSS3 code to arrange text in multi columns?

Here is the code for arranging text in multi columns,

```
<style>
.multi_col {
    /* Column count property */
    -webkit-column-count: 4;
    -moz-column-count: 4;
    column-count: 4;

    /* Column gap property */
    -webkit-column-gap: 40px;
    -moz-column-gap: 40px;
    column-gap: 40px;

    /* Column style property */
    -webkit-column-rule-style: solid;
    -moz-column-rule-style: solid;
    column-rule-style: solid;
}
</style>
```

32 . What is CSS3 Outline offset?

CSS3 outline, draws a line around the element at outside boarder.

Sample Code for creating Outline :

```
<style>
div {
    margin: 20px;
    padding: 10px;
    width: 300px;
    height: 100px;
    border: 5px solid pink;
    outline: 5px solid green;
    outline-offset: 15px;
}
</style>
```

```
}
```

```
</style>
```

33 . What is the CSS3 hsl() function?

The **hsl()** function define colors using the **Hue-saturation-lightness model (HSL)**. HSL stands for hue, saturation, and lightness – and represents a cylindrical-coordinate representation of colors.

Example :

```
#p1 {background-color:hsl(120,100%,50%);/* green */  
#p2 {background-color:hsl(120,100%,75%);/* light green */  
#p3 {background-color:hsl(120,100%,25%);/* dark green */  
#p4 {background-color:hsl(120,60%,70%);/* pastel green */
```

34 . What is CSS3 hsla() Function?

The **hsla()** function define colors using the **Hue-Saturation-Lightness-Alpha** model (HSLA). HSLA color values are an extension of HSL color values with an alpha channel – which specifies the opacity of the color.

Example :

```
#p1 {background-color:hsla(120,100%,50%,0.3);/* green */  
#p2 {background-color:hsla(120,100%,75%,0.3);/* light green */  
#p3 {background-color:hsla(120,100%,25%,0.3);/* dark green */  
#p4 {background-color:hsla(120,60%,70%,0.3);/* pastel green */
```

Codewitharray.com 8007532194

1. What is JavaScript ?

JavaScript is the programming language of the Web. Javascript Most commonly used websites this is client-side script to interact with the user and make dynamic pages. All modern web browsers on desktops, tablets, and smart phones are using JavaScript.

2 . Which company developed JavaScript ?

The first JavaScript engine was created by Brendan Eich at Netscape, for the Netscape Navigator Web browser. The engine, code-named SpiderMonkey, is implemented in C. It has since been updated (in JavaScript 1.5) to conform to ECMAScript 3.

3 . How to create function in JavaScript ?

To create function in JavaScript, follow the following syntax :

```
<script type="text/javascript">
  function function_name(){
    //function body
  }
</script>
```

4 . What is Javascript Variables ?

JavaScript uses reserved keyword var to declare a variable. A variable must have a unique name. You can assign a value to a variable using equal to (=) operator when you declare it or before using it.

There are two types of variables in JavaScript :

- 1. local variables** : A variable that is declared inside of a function definition is called local variable and has scope to that function only.

2.global variables : A variable that is declared outside of a function definition is called a global variable and its scope is throughout your program means its value is accessible and modifiable throughout your program.

Basic Syntax :

```
var <variable-name>;  
var <variable-name> = <value>;
```

5 . How many Operators in Javascript ?

JavaScript operators are used to assign values, compare values, arithmetic operations, and more. For example $1 + 2$, where + sign is an operator and 1 is left operand and 2 is right operand. + operator adds two numeric values and produces a result which is 3 in this case.

1. Arithmetic Operators
2. Comparison Operators
3. Logical Operators
4. Bitwise Operators
5. Assignment Operators

6 . What are JavaScript Data Types ?

Following are the JavaScript Data types :

1. Primitive data type
2. Non-primitive (reference) data type

JavaScript has dynamic types. This means that the same variable can be used to hold different data types :

```
var x; // Now x is undefined  
var x = 10; // Now x is a Number  
var x = "Name"; // Now x is a String
```

Primitive data types : There are six types of primitive data types in JavaScript. They are as follows :

Number
String
Boolean
Null
Symbol
Undefined

Non-Primitive data types : The non-primitive data types are as follows :

Object
Array
RegExp

7 . What is a prompt box ?

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

8 . What is `this` keyword in JavaScript ?

â€˜thisâ€™ keyword refers to the object from where it was called.

9 . What is the difference between == and === ?

The == operator checks equality only whereas === checks equality and data type i.e. value must be of same type.

10 . What is the function of delete operator ?

The functionality of delete operator is used to delete all variables and objects in a program but it cannot delete variables declared with VAR keyword.

11 . 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.

12 . How to create objects in JavaScript ?

There are 3 ways to create object in JavaScript.

By object literal

By creating instance of Object

By Object Constructor

Let's see a simple code to create object using object literal.

Ex : emp={id:100,name:"Free Time Learn",salary:45000}

13 . Difference between Client side JavaScript and Server side JavaScript ?

Client side JavaScript comprises the basic language and predefined objects which are relevant to running java script in a browser. The client side JavaScript is embedded directly by in the HTML pages. This script is interpreted by the browser at run time.

Server side JavaScript also resembles like client side java script. It has relevant java script which is to run in a server. The server side JavaScript are deployed only after compilation.

14 . How to set the cursor to wait in JavaScript ?

The cursor can be set to wait in JavaScript by using the property "cursor". The following example illustrates the usage :

```
<script type="text/javascript">
  window.document.body.style.cursor = "wait";
</script>
```

15 . What is callback ?

A callback is a plain JavaScript function passed to some method as an argument or option. Some callbacks are just events, called to give the user a chance to react when a certain state is triggered.

16 . What is closure ?

Closures are created whenever a variable that is defined outside the current scope is accessed from within some inner scope.

17 . Can you access Cookie using javascript ?

JavaScript can also manipulate cookies using the cookie property of the Document object. JavaScript can read, create, modify, and delete the cookie or cookies that apply to the current web page.

18 . How to create a Cookie using JavaScript ?

The simplest way to create a cookie is to assign a string value to the document.cookie object, which looks like this :

Syntax : document.cookie = "key1 = value1; key2 = value2; expires = date";

19 . How to redirect a url using JavaScript ?

His is very simple to do a page redirect using JavaScript at client side. To redirect your site visitors to a new page, you just need to add a line in your head

Section as follows :

```
<head>
<script type="text/javascript">
  window.location="http://www.newlocation.com";
</script>
</head>
```

20 . What is purpose of onError event handler in JavaScript ?

The onerror event handler provides three pieces of information to identify the exact nature of the error :

Error message : The same message that the browser would display for the given error.

URL : The file in which the error occurred.

Line number : The line number in the given URL that caused the error.

21 . What are the common Errors in JavaScript ?

The common errors in JavaScript programming are the following :

Spelling and typing errors.

Missing brackets or quotation marks.

Mismatching quotation marks.

Using single equal sign instead of double equal sign in comparison.

Referencing objects that does not exist.

Using reserved keywords for the variable naming.

Using the wrong type of brackets.

These are the main causes of these errors.

In JavaScript, there are the following three types of errors :

Syntax Error

Runtime Error

Logic Error

22 . What Is EncodeURI() Function ?

The encodeURI() function is used to encode a URI. This function encodes all special characters, except these < , / ? : @ & = + \$ #>.

Example :

```
var uri="http://www.freetimelearning.com/this is a educational website";
var encodedURI = encodeURI(uri);
console.log(encodedURI);
```

Output :

http://www.freetimelearning.com/this%20is%20a%20educational%20website

23 . What is NaN property in JavaScript?

Nan property represents “Not-a-Number” value. It indicates a value which is not a legal number.

typeof of a NaN will return a Number .

To check if a value is NaN, we use the isNaN() function,

Note : isNaN() function converts the given value to a Number type, and then equates to NaN.

```
isNaN("Hello") // Returns true
isNaN(345)    // Returns false
isNaN('1')    // Returns false, since '1' is converted to Number type which results in 0 ( a number)
isNaN(true)   // Returns false, since true converted to Number type results in 1 ( a number)
isNaN(false)  // Returns false
isNaN(undefined) // Returns true
```

24 . Explain Higher Order Functions in javascript.

Functions that operate on other functions, either by taking them as arguments or by returning them, are called higher-order functions.

Higher order functions are a result of functions being first-class citizens in javascript.

Examples of higher order functions :

```
function higherOrder(fn) {  
    fn();  
}  
  
higherOrder(function() { console.log("Hello world") });
```

```
function higherOrder2() {  
    return function() {  
        return "Do something";  
    }  
}  
  
var x = higherOrder2();  
x() // Returns "Do something"
```

25 . What is negative infinity?

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

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

Breaking within a string statement can be done by the use of 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 break in line.

Example:

```
var x=1, y=2,  
z=  
x+y;
```

The above code is perfectly fine, though not advisable as it hampers debugging.

27 . Write the code for adding new elements dynamically?

```
<html>  
    <head>  
        <title>Adding New Elements</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>  
        <div id="firstP"></div>  
    </body>  
</html>
```

```
</script>
</head>
<body> <p id="firstP">firstP</p> </body>
</html>
```

28 . What are global variables? How are these variable declared and what are the problems associated with using them?

Global variables are those that are available throughout the length of the code, that is, these have 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 that are 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.

29 . In JavaScript, what is 'This' keyword?

In JavaScript, the keyword 'this' refers to the object it belongs to and gives different values depending upon its usage.

For Example :

- * In method – refers own object
- * Alone – refers to the global object
- * Function – undefined (in strict mode)

30 . Which symbol is used for comments in Javascript?

```
// for Single line comments and
```

```
/* Multi Line Comment */
```

31 . 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 pages in the web application.

32 . What do 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.

33 . What is the function of 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;
```

34 . 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 doesn't exist

35 . 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://freetimelearning.com/javascript/index.php"; //--&gt;
&lt;/script&gt;</pre>
```

36 . 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.

37 . Explain what is pop()method in JavaScript?

The pop() method is similar as the shift() method but the difference is that the Shift method works at the start of the array. Also the pop() method take the last element off of the given array and returns it. The array on which is called is then altered.

38 . what is the disadvantage of using innerHTML in JavaScript?

If you use innerHTML in JavaScript the disadvantage is

- * Content is replaced everywhere
- * We cannot use 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 build into elements, therefore its much slower

* The innerHTML does not provide validation and therefore we can potentially insert valid and broken HTML in the document and break it

39 . What are the two basic groups of datatypes in JavaScript?

There are two types of data types in JavaScript. JavaScript, also known as ECMAScript specifies six primitive data types and object (or) non-primitive data types

1. Primitive data type
2. Non-primitive (reference) data type

JavaScript has dynamic types. This means that the same variable can be used to hold different data types:

```
var x;      // Now x is undefined
var x = 10;    // Now x is a Number
var x = "Name"; // Now x is a String
```

40 . How generic objects can be created?

Generic objects can be created as:

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

41 . 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
}
```

42 . What is the use of blur function?

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

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

The 'Navigator.appversion' is used to find the name of the operating system in the client machine.

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

There are three types of errors :

Load time errors : Errors which come up when loading a web page like improper syntax errors are known as Load time errors and it generates the errors dynamically.

Run time 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 which is having different operation.

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

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

46 . What is the difference between JavaScript and Jscript?

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

47 . 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 codes, 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 to work 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";  
}
```

48 . What is the significance of, and reason for, wrapping the entire content of a JavaScript source file in a function block?

This is an increasingly common practice, employed by many popular JavaScript libraries (jQuery, Node.js, etc.). This technique creates a closure around the entire contents of the file which, perhaps most importantly, creates a private namespace and thereby helps avoid potential name clashes between different JavaScript modules and libraries.

Another feature of this technique is to allow for an easily referenceable (presumably shorter) alias for a global variable. This is often used, for example, in jQuery plugins. jQuery allows you to disable the \$ reference to the jQuery namespace, using jQuery.noConflict(). If this has been done, your code can still use \$ employing this closure technique, as follows:

```
(function($) { /* jQuery plugin code referencing $ */ })(jQuery);
```

49 . What is the DOM?

DOM stands for **Document Object Model** is an interface (**API**) for HTML and XML documents. When the browser first reads (parses) our HTML document it creates a big object, a really big object based on the HTML document this is the DOM. It is a tree-like structure that is modeled from the HTML document. The DOM is used for interacting and modifying the DOM structure or specific Elements or Nodes.

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document Object Model</title>
</head>

<body>
  <div>
    <p>
      <span></span>
    </p>
    <label></label>
    <input>
  </div>
</body>

</html>
```

The document object in JavaScript represents the DOM. It provides us many methods that we can use to selecting elements to update element contents and many more.

50 . 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 will be checked, this alert will return TRUE.

51 . Explain 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.

52 . Explain the for-in loop?

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 to the variable name, and the loop is continued till all the properties of the object are depleted.

53 . Describe the 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();
```

54 . Define 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 parent will also work as if it were clicked too.

55 . 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.

56 . Write the point of difference between 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 while web-farm is a larger setup that uses more than one server.

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

Break statement is used to come out of the current loop while the continue statement continues the current loop with a new recurrence.

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

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

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

59 . 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 proper execution of all these events. Event handlers are an extra attribute of the object. This attribute includes event's name and the action taken if the event takes place.

60 . 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 client's screen

AvailWidth : Gives the width of 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

61 . Explain 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 = [ "free" ];
name.unshift( "time" );
name.unshift( "learning", "FTL" );
console.log(name);
```

The output is shown below:

```
[ " free ", " time ", " learning ", " FTL " ]
```

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

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

```
<script>
    var url="my test.php?name=volkswagen&car=polo";

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

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

Output:

```
my%20test.php?name=volkswagen&car=polo
my test.php?name=volkswagen&car=polo
```

63 . What's the difference between event.preventDefault() and event.stopPropagation() methods?

The event.preventDefault() method **prevents** the default behavior of an element. If used in a form element it **prevents** it from submitting. If used in an anchor element it **prevents** it from navigating. If used in a contextmenu it **prevents** it from showing or displaying. While the event.stopPropagation() method stops the propagation of an event or it stops the event from occurring in the bubbling or capturing phase.

64 . How to know if the event.preventDefault() method was used in an element?

We can use the event.defaultPrevented property in the event object. It returns a boolean indicating if the event.preventDefault() was called in a particular element.

65 . What are the falsy values in JavaScript?

```
const falsyValues = [ "", 0, null, undefined, NaN, false];
```

falsy values are values that when converted to boolean becomes false.

66 . What is the prototype of an object?

A prototype in simplest terms is a blueprint of an object. It is used as a fallback for properties and methods if it does exist in the current object. It's the way to share properties and functionality between objects. It's the core concept around JavaScript's **Prototypal Inheritance**.

```
const o = {};
console.log(o.toString()); // logs [object Object]
```

Even though the `o.toString` method does not exist in the `o` object it does not throw an error instead returns a string [object Object]. When a property does not exist in the object it looks into its prototype and if it still does not exist it looks into the prototype's prototype and so on until it finds a property with the same in the Prototype Chain. The end of the Prototype Chain is the `Object.prototype`.

```
console.log(o.toString === Object.prototype.toString); // logs true
// which means we're looking up the Prototype Chain and it reached
// the Object.prototype and used the "toString" method.
```

67 . What is the use `Function.prototype.apply` method?

The `apply` invokes a function specifying the `this` or the "owner" object of that function on that time of invocation.

```
const details = {
  message: 'Hello World!'
};

function getMessage(){
  return this.message;
}
getMessage.apply(details); // returns 'Hello World!'
```

This method works like `Function.prototype.call` the only difference is how we pass arguments. In `apply` we pass arguments as an array.

```
const person = {
  name: "Marko Polo"
};

function greeting(greetingMessage) {
  return `${greetingMessage} ${this.name}`;
}

greeting.apply(person, ['Hello']); // returns "Hello Marko Polo!"
```

68 . What is the use `Function.prototype.call` method?

The `call` invokes a function specifying the `this` or the "owner" object of that function on that time of invocation.

```
const details = {
  message: 'Hello World!'
};

function getMessage(){
  return this.message;
}
```

```
}

getMessage.call(details); // returns 'Hello World!'
```

This method works like Function.prototype.apply the only difference is how we pass arguments. In call we pass directly the arguments separating them with a comma , for every argument.

```
const person = {
  name: "Marko Polo"
};

function greeting(greetingMessage) {
  return `${greetingMessage} ${this.name}`;
}

greeting.call(person, 'Hello'); // returns "Hello Marko Polo!"
```

69 . What is console.log() and why it is used?

It is not a jQuery feature but a feature for debugging purposes used by developers. It is used to writes a message to the console.

console.log() accepts a parameter which can be an object, an array or any message.

Syntax : console.log(name); // here name is object, an array or variable.

Example :

```
$('#form').submit(function() {

  console.log('Your form is submitted successfully!');

  // do something

});
```

70 . What is the importance of JavaScript?

It is a client side technology, it is mainly used for gives client side validation, but it have lot of features which are given below;

- * It is an object-based scripting language.
- * It helps the user more control over the browser.
- * JavaScript Handling dates and time also.
- * It also helps to Detect the client's browser and Operating System.
- * JavaScript is light weighted & case sensitive.
- * All most JavaScript control statements syntax is same as syntax of control statements in C programming language.

71 . Explain export & import in JavaScript?

The **export** statement is used when creating modules for JavaScript to export functions, primitive values or objects from the blades so they can be used by the programs with import statement which works opposite to it.

The **import** statement is used when we import buildings which are exported by another module. All imported modules are in strict mode. This statement can't be used in embedded scripts.

72 . How can you do a character count in Javascript?

To perform a character count in JavaScript, use the following syntax:

```
var str = "Hello World!";  
  
var n = str.length;
```

73 . What is the method to change the title of a page using JavaScript?

Generally, the page title varies based on the HTML document and the element structure. We can give id to an element and use code:

```
document.getElementById('page-title-id').innerHTML=NewTitle;
```

74 . Can you redirect a page to another page using JavaScript? How?

Yes, it is possible to redirect a page to another page or URL using JavaScript by using location, replace, and location.assign.

75 . What you mean by Hoisting in JS?

The concept of hoisting stands for uplifting the variable and functions on the top of their scope before the code execution takes place. In the JavaScript mechanism, no matter wherever the functions are declared, they are taken on the topmost position, immaterial of their scope (global or local).

76 . How can you submit a form using JS?

To submit a simple form, the below code will work.

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

77 . What is BOM in JS?

BOM stands for **Browser Object Model**, a more extensive representation of elements provided by the browser like document, location, history, and frames exposed to JavaScript. **DOM** is a part of **BOM**, and to access the document, one can use code document or window.document.

78 . Please list some of the Design Patterns in JS.

Whenever somebody wants to reuse solutions for regularly occurring problems in software designing, Design Patterns will solve the purpose.

Latest Design Patterns in JS are :

- * Module Design pattern
- * Revealing Module pattern
- * Prototype Design pattern
- * Revealing Prototype pattern
- * Observer Design pattern
- * Singleton

79 . Deep vs. shallow object copying in JavaScript.

Deep Copying : This means that all the values of the existing/original variables are copied to a new variable, and thus, disconnected from the existing variables.

Shallow Copying : When commanded for shallow copying, not all the existing variables are disconnected, and some of the values or sub-values are still connected to the original.

80 . How to formulate a cookie using JS?

A cookie is a set of data saved on the computer and accessed by the browser.

Step to create a JavaScript cookie :

```
document.cookie = "cookiename=Ftl"; expires = date";
```

81 . If you need to calculate the Fibonacci series in JS, what will you do?

Fibonacci series is a pattern in which each given value is the sum of the previous two, and it starts with 0,1.

Method :

- * use function fib(n),
- * Declare var a=0 and b=1
- * Use this condition (var i=0; i<n; i++)
- * Use var temp = a+b;
- * Next make a=b
- * And b=temp;
- * }

* Return a;

The series will come like 0,1,1,2,3,5...

82 . In JavaScript, append a new element at the end of the array.

To append an element in a JavaScript array, we use push(), and to remove an array, pop() is used.

Syntax :

```
array.push(item1, item2, ..., itemX)
```

83 . How do we add/remove properties to objects dynamically in JS?

We can add a property to an object by using :

```
object.property_name =value
```

To delete a property, we can use :

```
object.property_name
```

Example :

```
let user = new Object();

// adding a property

user.name='ftl';

user.age =22;

console.log(user);

delete user.age;
```

84 . What is the procedure to merge two arrays?

Syntax :

```
var arr1 = [1,2,3]
var arr2 = [4,5,6,7]
var mergedArrays = [...arr1, ...arr2]
document.write('Merged arrays', mergedArrays)
```

85 . What is Means by Currying in Javascript?

Currying is a method of evaluating the function with many arguments, into a sequence of function with a single argument. Currying helps you to avoid spending the same variable again and again. It helps to create a larger order function

86 . How does inheritance work in JavaScript?

When it comes to inheritance, JavaScript simply has one construct: objects. Each object has a separate property which operates a link to another object called its model. That prototype object has a prototype of its individual, and so on until an object is terminated with null as its prototype.

87 . Why do we use onabort event in JavaScript?

The onabort event is executed when the loading of an image is aborted. Use it in an image tag like this:

```

```

Here, you can display an alert when the loading of image aborts :

```
function abortFunc()
{
    alert('Image isn't loading!');
}
```

88 . What are logical errors in JavaScript?

Logic errors occur when there is a mistake in the logic. This can lead your script to work inappropriately to give unexpected results.

89 . Which event triggers in JavaScript when an object loses focus?

The blur event triggers when object loses focus. An example here displays that we write a name in the textbox and then when the object loses focus, the text color is changed. This happens since the blur event triggers when the focus ends:

```
<!DOCTYPE html>
<html>
<body>
<h2>Testing</h2>
<input type="text" onblur="display(this)">
<script>
    function display(z) {
        z.style.color = "green";
    }

```

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

90 . What is the usage of onpagehide event in JavaScript?

When a visitor leaves the web page, then the onpagehide event triggers. The visitor can move to another page or click a link after leaving the current web page. An example here displays an alert box when the user tries to leave the page. This happens since onpagehide event fires when the page is left :

```
<!DOCTYPE html>
<html>
  <body onpagehide="display()">
    <p>Close the page</p>
    <script>
      function display() {
        alert("It was nice having you here!");
      }
    </script>
  </body>
</html>
```

Codewitharrays.in 8007592194

1. What is jQuery ?

jQuery is a set of (a rich bundle of) JavaScript libraries that helps users apply dynamic functionality to web pages with great ease. It is designed to simplify the client-side scripting of HTML.

2 . Why do we use jQuery ?

It is very easy to learn and use.
It is used to develop browser compatible web applications.
It improves the performance of an application.
It is very fast and extensible.
It facilitates you to write minimal lines of codes for UI related functions.
It provides a cross-browser support.

3 . What are the effects methods used in jQuery ?

These are some effects methods used in jQuery :

show()
hide()
toggle()
fadeIn()
fadeOut()

4 . What is advantages of jQuery ?

Easy to learn : jQuery is easy to learn because it supports same JavaScript style coding.

Write less do more : jQuery provides a rich set of features that increase developers' productivity by writing less and readable code.

Excellent API Documentation : jQuery provides excellent online API documentation.

Cross-browser support : jQuery provides excellent cross-browser support without writing extra code.

Unobtrusive : jQuery is unobtrusive which allows separation of concerns by separating html and jQuery code.

5 . How JavaScript and jQuery are different ?

JavaScript is a language While jQuery is a library built in the JavaScript language that helps to use the JavaScript language.

6 . What does dollar sign (\$) means in jQuery ?

Dollar Sign is nothing but it's an alias for JQuery. Take a look at below jQuery code.

```
$(document).ready(function(){  
    //script-content  
});
```

Over here \$ sign can be replaced with "jQuery" keyword.

```
jQuery(document).ready(function(){  
    //script-content  
});
```

7 . What is a CDN ?

A Content Delivery Network or Content Distribution Network (CDN) is a large distributed system of servers deployed in multiple data centers across the Internet. The goal of a CDN is to serve content to end-users with high availability and high performance.

8 . How to load jQuery from CDN ?

Code to load jQuery from two types of CDNs :

Google CDN : <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>

Microsoft CDN : <script src="https://ajax.aspnetcdn.com/ajax/jQuery/jquery-3.2.1.min.js"></script>

9 . jQuery library is server scripting or client scripting ?

The jQuery library is a client side Scripting.

10 . Is jQuery a W3C standard ?

No, jQuery is not a W3C standard.

11 . What are selectors in jQuery? How many types of selectors in jQuery ?

If you want to work with an element on the web page, first you need to find it. Selectors are used to find the HTML elements in jQuery. There are many types of selectors. Some basic selectors are:

1. Name Selectors - It is used to selects all elements which match with the given element Name. Ex : \$("p") , \$("h1"), \$("h2"), etc.

2. ID(#) Selectors - It is used to selects a single element which matches with the given ID. Ex : \$("#test")

3. class(.) Selectors - It is used to selects all elements which match with the given Class. Ex : \$(".test")

and more selectors please [click here!](#)

12 . How to read, write and delete cookies in jQuery ?

To deal with cookies in jQuery we have to use the ftl cookie plugin.

1. Create cookie : \$.ftl("cookie_name", "cookie_value");
2. Read Cookie : \$.ftl("cookie_name");
3. Delete cookie : \$.ftl("cookie_name", "remove");

13 . What is difference between jQuery.get() and jQuery.ajax() method ?

ajax() method is more powerful and configurable, allows you to specify how long to wait and how to handle error, get() is a specialization to over ajax just to retrieve some data.

14 . What is the difference between .empty(), .remove() and .detach() methods in jQuery ?

All these methods .empty(), .remove() and .detach() are used for removing elements from DOM but they all are different.

.empty() : This method removes all the child element of the matched element where remove() method removes set of matched elements from DOM.

.remove() : Use .remove() when you want to remove the element itself, as well as everything inside it. In addition to the elements themselves, all bound events and jQuery data associated with the elements are removed.

.detach() : This method is the same as .remove(), except that .detach() keeps all jQuery data associated with the removed elements. This method is useful when removed elements are to be reinserted into the DOM at a later time.

15 . What are the four parameters used for jQuery Ajax method ?

The four parameters are

URL : Need to specify the URL to send the request

type : Specifies type of request(Get or Post)

data : Specifies data to be sent to server

Cache : Whether the browser should cache the requested page

16 . What is the use of jQuery load method ?

jQuery load method is a powerful AJAX method which is used to load the data from a server and assign the data into the element without loading the page.

17 . What are various methods to make ajax request in jQuery ?

Using below jQuery methods, you can make ajax calls.

load() : Load a piece of html into a container DOM

\$.getJSON() : Load JSON with GET method.

\$.getScript() : Load a JavaScript file.

\$.get() : Use to make a GET call and play extensively with the response.

\$.post() : Use to make a POST call and don't want to load the response to some container DOM.

\$.ajax() : Use this to do something on XHR failures, or to specify ajax options (e.g. cache: true) on the fly.

18 . What is jQuery UI ?

jQuery UI is a curated set of user interface interactions, effects, widgets, and themes built on top of the jQuery JavaScript Library that can be used to build interactive web applications.

19 . Mention some advantages of jQuery.

There are many advantages of using jQuery. Some of them include :

It is like an **enhanced version of JavaScript** so there is no overhead in learning a new syntax.

jQuery has the ability to keep the code **simple, readable, clear** and **reusable**.

It has **Cross-browser support**.

This would remove the requirement for writing complex loops and **DOM scripting** library calls.

jQuery helps in **event detection and handling**.

It provides tons of **plug-ins** for all kind of needs.

20 . What is the purpose of fadeToggle() method in JQuery?

The jQuery fadeToggle() method is used to toggle between the fadeIn() and fadeOut() methods. If the elements are faded in, it makes them faded out, and if they are faded out, it makes them faded in.

Syntax :

```
$(selector).fadeToggle();
$(selector).fadeToggle(speed,callback);
$(selector).fadeToggle(speed, easing, callback);
```

speed : It is an optional parameter. It specifies the speed of the delay. Its possible values are slow, fast and milliseconds.

easing: It specifies the easing function to be used for transition.

callback: It is also an optional parameter. It specifies the function to be called after completion of fadeToggle() effect.

21 . What is the use of delay() method in JQuery?

The jQuery delay() method is used to delay the execution of functions in the queue. It is the best method to make a delay between the queued jQuery effects. The jQuery delay () method sets a timer to delay the execution of the next item in the queue.

Syntax :

```
$(<selector>).delay (<speed>, <queueName>) â€¢
```

speed : It is an optional parameter. It specifies the speed of the delay. Its possible values are slow, fast and milliseconds.

queueName : It is also an optional parameter. It specifies the name of the queue. Its default value is "fx" the standard queue effect.

22 . What is the use of html() method in JQuery?

The jQuery html() method is used to change the entire content of the selected elements. It replaces the selected element content with new contents.

Syntax :

```
$(<document>).ready(function(){
    $("button").click(function(){
        $("p").html("Hello <b>freetimelearning.com</b>");
    });
});â€¢
```

23 . What is the use of css() method in JQuery?

The jQuery CSS() method is used to get (return)or set style properties or values for selected elements. It facilitates you to get one or more style properties. The jQuery CSS() provides two ways:

Return a CSS property

It is used to get the value of a specified CSS property.

```
$(<document>).ready(function(){
    $("button").click(function(){
        alert("Background color = " + $("p").css("background-color"));
    });
}); â€¢
```

Set a CSS property

This property is used to set a specific value for all matched element.

```
$document.ready(function(){
    $("button").click(function(){
        $("p").css("background-color", "violet");
    });
});
```

24 . Can you use multiple document.ready() function on the same page?

Yes. You can use any number of document.ready() function on the same page.

Example :

```
$document.ready(function() {
    $("h1").css("background-color", "red");
});
$document.ready(function() {
    $("p").css("background-color", "blue");
});
```

25 . What is a use of jQuery filter?

jQuery filter is used to filter the specific values from the object. It filters the result of your original query into specific elements.

26 . What is the use of serialize() method in JQuery?

The jQuery serialize() method is used to create a text string in standard URL-encoded notation. It serializes the form values so that its serialized values can be used in the URL query string while making an AJAX request.

Syntax :

```
$document.ready(function(){
    $("button").click(function(){
        $("div").text($("#form").serialize());
    });
});
```

27 . What is the use of the animate() method in jQuery?

The animate function is used to apply the custom animation effect to elements. **Syntax :**

```
$(selector).animate({params}, [duration], [easing], [callback])
```

Here,

- * "param" defines the CSS properties on which you want to apply the animation.
- * "duration" specify how long the animation run. It can be one of the following values : "slow," "fast," "normal" or milliseconds
- * "easing" is the string which specifies the function for the transition.
- * "callback" is the function which we want to run once the animation effect is complete.

28 . What does the jQuery Ajax event method ajaxComplete() do?

Whenever an Ajax request completes, jQuery triggers the ajaxComplete event. Any and all handlers that have been registered with the .ajaxComplete() method are executed at this time.

29 . What does the jQuery ajax event method ajaxStart() do?

Whenever an Ajax request is about to be sent, jQuery checks whether there are any other outstanding Ajax requests. If none are in progress, jQuery triggers the ajaxStart event. Any and all handlers that have been registered with the .ajaxStart() method are executed at this time.

30 . What is the difference between onload() and document.ready() methods?

body.Onload() event will be called only after the DOM and associated resources like images get loaded, but jQuery's document.ready() event will be called once the DOM is loaded and it does not wait for the resources such as images to be loaded.

31 . What is Qunit?

QUnit is a powerful, easy-to-use JavaScript unit testing framework. It's used by the *jQuery, jQuery UI, and jQuery Mobile* projects and is capable of testing any generic JavaScript code.

32 . What is jQuery.noConflict?

Usually, JS functions and variables use \$ as a name. In jQuery, \$ is just an alias for jQuery, so we don't need to use \$. If we have to use a JS library along with jQuery, the control of \$ is given to the JS library. To give this control, we use jQuery.noConflict(). It is also used to assign a new name to a variable.

```
var newname = jQuery.noConflict();
```

33 . Explain the various Ajax functions available in jQuery?

There are many methods like :

.ajaxStart() : register the handler to be called when the first Ajax request begins.

.ajaxStop() : register the handler to be called when all requests are complete.

.ajaxSuccess() : register the handler to be called when an Ajax request is successfully completed..

34 . What is the difference between width() vs css('width') in jQuery?

CSS('width') returns the width value in pixels, whereas width() returns the integer (without the unit values). For example:

```
div{  
width: 20cm;  
}
```

If you print the values:

```
$(this).width();  
$(this).css('width');
```

you will get the values like 756 and 756px respectively. Note that though we specified width in cm, it is converted to pixel (px) for output purposes.

35 . What is the difference between bind() vs live() vs delegate() methods in jQuery?

bind() : this method registers the event handler directly to the required DOM element.

```
$("#members a").bind("click", function(f){...});
```

This means any matching anchors will have this event handler attached!

live() : this method attaches the event handler to the root of the document. This means one handler can be used for all events that propagated to the root. The handler is thus attached only once.

delegate() : in this method, you can choose where to attach the handler. This is the most efficient and robust method for delegation.

Eg:

```
$("#members").delegate("ul li a", "click", function(f){...});
```

36 . Explain the difference between jquery.size() and jquery.length?

In jquery both works and gives the same result but now the time in current version of jquery we use the `jquery.length` because `jquery.size` is depreciated.

Below is the some points about this properties :

`jquery.size()` and `jquery.length()` both returns the number of element in an object.

`jquery.length()` property is faster as compared to `jquery.size()` method.

`jquery.length()` property prefered compared to `size()` because it does not have overhead of a function call.

Very significant difference is that `jquery.size()` is depreciated so we always use `jquery.length()` instead of `jquery.size()`.

For Examples: Counts the number of unordered list li on the page to demonstrate the same according to the jQuery documentation.

```
$('li').size();  
$('li').length;// faster
```

37 . What is jQuery Datepicker in jQuery?

It is a plugin/widget that adds datepicker functionality in HTML pages. It is highly configurable and can be customized for date format, language, restricting date selection, etc. Refer to this jQuery documentation for datepicker options.

38 . Difference between Map and Grep function in jQuery?

Map function translates a set of elements into another set of values in a jQuery array that may or may not have the elements. The map is called as :

```
$("<element>").map(<function to execute for elements in the object>)
```

Grep, on the other hand, finds an element in an array.

```
jQuery.grep(myArr, function(){})
```

39 . Difference between prop and attr?

Both `attr()` and `prop()` can be used to set or get an element's value, however `attr()` returns the original (default) value whereas `prop()` returns the most recent (current) value. For example, if a text input had an initial value of 'Male,' and later it was changed by the user to 'female,' `attr()` will return the value as 'Male' whereas `prop()` will return the value as 'female.'

40 . What are all the ways to include jQuery in a page?

Following are the ways to include jQuery in a page :

- * Local copy inside script tag
- * Remote copy of jQuery.com
- * Remote copy of Ajax API
- * Local copy of script manager control
- * Embedded script using client script object

41 . Which is the fastest selector in jQuery?

ID and Element are the fastest selectors in jQuery.

42 . What is the slowest selector in jQuery?

Class selectors are the slowest selectors in jQuery.

43 . What is chaining in jQuery?

Chaining is a powerful feature of jQuery. Chaining means specifying multiple functions and/or selectors to an element.

Chaining reduces the code segment and keeps it very clean and easy to understand. Generally chaining uses the jQuery built in functions that makes compilation a bit faster.

By using chaining we can write the above code as follows :

```
$document.ready(function() {  
    $("#div2").html($("#txtBox").prop("readonly")) + '<br>';  
    $("#div3").html($("#txtBox").attr("readonly"));  
});
```

44 . How to work with parent(), children() and siblings() methods in jQuery?

The parent() function returns the parent of the selected element by calling the jQuery parent() function.
The siblings() function returns all the siblings of given HTML elements.

45 . What is jQuery.holdReady() function?

jQuery.holdReady() function is what we can hold or release the execution of jQuery's ready event. This method should be called before we run the ready event. To delay the ready event, we need to call jQuery.holdReady(true);

When we want to release the ready event then we need to call jQuery.holdReady(false);

This function is helpful when we want to load any jQuery plugin before the execution of the ready event or want to perform certain events/functions before document.ready() loads .

46 . What is resize() function in jQuery?

The resize event occurs when the size of the browser window is changed. jQuery resize() Method attaches window element to an event handler function and this event handler function executes when the resize event occurs.

jQuery resize() Method Syntax :

```
$window.resize(function(){
  //code that executes when the resize event occurs.
});
```

47 . Differentiate the concepts of .js and .min.js?

The jQuery library has two different versions Development and Production. The other name for the deployment version is minified version.

Considering the functionality, both the files are much similar to each other. Being smaller in size, the .min.js gets loaded quickly saving the bandwidth.

48 . Elaborate jQuery Ajax Events.

Ajax methods trigger an event handler that results in jQuery Ajax Events. Some of the examples of jQuery Ajax Events are as listed below.

These events are categorized into local events and global events.

1. ajaxStart() : It is a Global Event, This event triggers as a result of starting of an Ajax request, provided no other Ajax request currently running.
2. beforeSend() : It is a Local Event, as the name indicates, this event gets invoked before Ajax request starts, thereby allowing to modify XMLHttpRequest objects.
3. ajaxSend() : It is a Global Event, and this event gets called before the Ajax request is run.
4. success() : It is a Local Event. This event triggers only if the Ajax request was successfully sent (i.e. while sending Ajax request, the event does not display any error from the server or from data).
5. ajaxSuccess() : It is a Global Event, triggers only if the request sent was successful.
6. error() : It is a Local Event, that gets triggered if an error occurs while executing the request. (You can have either error or a success callback while sending Ajax request)
7. ajaxError() : It is a Global Event, which behaves the same as its local counterpart error() event.
8. complete() : It is a Local Event. This event gets called regardless of the request being successful or result in an error, and complete callbacks are received, even for synchronous requests.
9. ajaxComplete() : It is a Global Event, which behaves the same as its local counterpart complete() event, even for synchronous requests.

10. ajaxStop() : It is a Global Event, that gets triggered when no Ajax requests are still being processed/ pending for processing.

49 . Explain the different ways in which we can debug jQuery.

There are few ways in which we can debug jQuery code.

Modern-day browsers such as Google Chrome, Mozilla Firefox, Opera and Safari have built-in Javascript debugger. To debug the jQuery code, Press F12 from your keyboard and the browser with the built-in debugger will open a UI, wherein select the 'Console' menu. The error will be displayed in the console menu if any.

You can write console.log() into the code to get the error text, you can also write debugger; in between the code line, due to debugger the script will start in **debug mode**, pressing F12 into the browser will open console which will **debug the code**, pressing F10 will read values of the jQuery objects, and this is how we can debug jQuery code.

50 . How jQuery stores data related to an element?

jQuery.data() method aids in attaching any type of data to DOM elements, free from memory leaks. jQuery makes sure that data is removed along with the DOM elements removed via jQuery methods.

Code for Storing and retrieving data related to an element.

```
$('#myDiv').data('keyName', { foo : 'bar'});
$('#myDiv').data('keyName'); // { foo : 'bar'}
```

51 . How we can get the value of a radio button using Jquery?

When a radio button is selected addition check attribute is not added to it. You have to enter "checked".

Example :

```
$(document).ready(function(){
    $("input[type='button']").on('click', function(){
        var radioValue = $("input[name='gender']:checked").val();
        if(radioValue){
            alert(radioValue);
        }
    });
});
```

52 . How to validate phone number using jQuery?

We can use regex ([0-9]{10})|([(0-9){3}\s+[0-9]{3}\-[0-9]{4})

53 . What are the element's relative commonly used in jQuery API?

1. **parent()** and **parents()** are used to select an element's parent or ancestors.

2. ***children()*** is used to select an element's immediate children.
 3. ***siblings()*** is used to select all of an element's surrounding sibling elements.
 4. ***prev()*** is used to select an element's immediate preceding sibling.
 5. ***next()*** is used to select an element's immediate following sibling.
 6. ***prevAll()*** is used to select all siblings coming before an element.
 7. ***nextAll()*** is used to select all siblings coming after an element.
 8. ***not()*** is used to remove elements from a selection using a selector.
 9. ***eq()*** is used to zero in on a single element in a selection by providing its offset position within the selection offset from zero.

54 . What do you mean by Tag name, Tag ID, Tag class?

Tag Name : Represents a tag name available in the DOM. For example `$('p')` selects all paragraphs `<p>` in the document.

Tag ID : Represents a tag available with the given ID in the DOM. For example `$('#some id')` selects the single element in the document that has an ID of some-id.

Tag Class : Represents a tag available with the given class in the DOM. For example `$('.some-class')` selects all elements in the document that have a class of some class.

55 . How to Valid an email using JQuery?

To validate email id using JQuery create a separate function `ValidEmail()`. Pass the email id as a parameter. Using Regex we can validate the email id. Example shown below.

```
function ValidEmail>EmailID {
var filter = /^[^\w\-.]+\@([^\[\]]{1,3}\.[^\-\.\-]{1,3}\.[^\-\.\-]{1,3})|(([^\w-]+\.)+)([a-zA-Z]{2,4}|[0-9]{1,4})$/
if (filter.test>EmailID) {
return true;
}
else {
return false;
}
}
```

56 . What is the use of JQuery \$.extend method?

Using JQuery \$.extend method we can merge the contents of two or more objects into the first object. The syntax is as follow.

```
var obj-0 = $.extend({}, obj-1, obj-2);
```

Let's we have two array's arr-i & arr-j. During development what I need is I want to merge arr-j values to arr-i. In this case jquery extend method helps.

```
$.extend( arr-i, arr-j );
```

But this is not recursive. To merge two objects recursively we need to pass the first parameter value true to extend method. As shown in below.

```
$.extend( true, arr-i, arr-j );
```

57 . What is JQuery \$.map method?

Jquery map method is used to translate items of an Array or object to new array of items. Let's take an example.

```
var tempArray = { "name":"RamanaReddy", "age":30, "designation":"UI Developer" };
var realArray = $.makeArray( tempArray )
$.map( realArray, function( val, i ) {
// Do something
});
```

1. What is React?

React is an open-source frontend JavaScript library developed by Facebook in 2011. React was created by Jordan Walke, a software engineer working for Facebook.

- * React is used for building user interfaces especially for single page applications.
- * It follows the component based approach which helps in building reusable UI components.
- * It is used for developing complex and interactive web and mobile UI. Even though it was open-sourced only in 2015, it has one of the largest communities supporting it.

2 . What are the major features of React?

The major features of React are:

- * It uses VirtualDOM instead of RealDOM considering that RealDOM manipulations are expensive.
- * It uses server-side rendering.
- * It follows uni-directional data flow or data binding.

Uses reusable/composable UI components to develop the view.

3 . What is JSX?

JSX is a XML-like syntax extension to ECMAScript (the acronym stands for JavaScript XML). Basically it just provides syntactic sugar for the `React.createElement()` function, giving us expressiveness of JavaScript along with HTML like template syntax.

In the example below text inside `<h1>` tag is returned as JavaScript function to the render function.

```
class App extends React.Component {  
  render() {  
    return(  
      <h1>Hello world</h1>  
    );  
  }  
}
```

```
<div>
  <h1>'Welcome to React world!'</h1>
</div>
)
}
}
```

4 . List some of the major advantages of React.

Some of the major advantages of React are:

- * It increases the application's performance
- * It can be conveniently used on the client as well as server side
- * Because of JSX, code's readability increases
- * React is easy to integrate with other frameworks like Meteor, Angular, etc
- * Using React, writing UI test cases become extremely easy

5 . How to create components in React?

There are two possible ways to create a component.

Function Components : This is the simplest way to create a component. Those are pure JavaScript functions that accept props object as first parameter and return React elements:

```
function Greeting({ message }) {
  return <h1>`Hello, ${message}`</h1>

}
```

Class Components : You can also use ES6 class to define a component. The above function component can be written as:

```
class Greeting extends React.Component {
  render() {
    return <h1>`Hello, ${this.props.message}`</h1>
  }
}
```

6 . When to use a Class Component over a Function Component?

If the component needs state or lifecycle methods then use class component otherwise use function component. However, from React 16.8 with the addition of Hooks, you could use state , lifecycle methods and other features that were only available in class component right in your function component.

7 . What are Pure Components?

React.PureComponent is exactly the same as React.Component except that it handles the shouldComponentUpdate() method for you. When props or state changes, PureComponent will do a shallow

comparison on both props and state. Component on the other hand won't compare current props and state to next out of the box. Thus, the component will *re-render* by default whenever shouldComponentUpdate is called.

8 . What is state in React?

State of a component is an object that holds some information that may change over the lifetime of the component. We should always try to make our state as simple as possible and minimize the number of stateful components.

Let's create an user component with message state,

```
class User extends React.Component {
  constructor(props) {
    super(props)

    this.state = {
      message: 'Welcome to React world'
    }
  }

  render() {
    return (
      <div>
        <h1>{this.state.message}</h1>
      </div>
    )
  }
}
```

9 . Why can't browsers read JSX?

Browsers can only read JavaScript objects but JSX is not a regular JavaScript object. Thus to enable a browser to read JSX, first, we need to transform JSX file into a JavaScript object using JSX transformers like Babel and then pass it to the browser.

10 . Explain the purpose of render() in React.

Each React component must have a render() mandatorily. It returns a single React element which is the representation of the native DOM component. If more than one HTML element needs to be rendered, then they must be grouped together inside one enclosing tag such as <form>, <group>, <div> etc. This function must be kept pure i.e., it must return the same result each time it is invoked.

11 . What are props in React?

Props are inputs to components. They are single values or objects containing a set of values that are passed to components on creation using a naming convention similar to HTML-tag attributes. They are data passed down from a parent component to a child component.

The primary purpose of props in React is to provide following component functionality:

- * Pass custom data to your component.

- * Trigger state changes.
- * Use via this.props.reactProp inside component's render() method.

For example, let us create an element with reactProp property:

```
<Element reactProp={'1'} />
```

This reactProp (or whatever you came up with) name then becomes a property attached to React's native props object which originally already exists on all components created using React library.

props.reactProp

12 . What is the difference between state and props?

Both props and state are plain JavaScript objects. While both of them hold information that influences the output of render, they are different in their functionality with respect to component. Props get passed to the component similar to function parameters whereas state is managed within the component similar to variables declared within a function.

13 . What is the purpose of callback function as an argument of setState()?

The callback function is invoked when setState finished and the component gets rendered. Since setState() is asynchronous the callback function is used for any post action.

Note: It is recommended to use lifecycle method rather than this callback function.

```
setState({ name: 'John' }, () => console.log('The name has updated and component re-rendered'))
```

14 . What is the use of refs?

The ref is used to return a reference to the element. They should be avoided in most cases, however, they can be useful when you need a direct access to the DOM element or an instance of a component.

15 . What are the lifecycle methods of ReactJS? What are the lifecycle methods of ReactJS?

componentWillMount : Executed before rendering and is used for App level configuration in your root component.

componentDidMount : Executed after first rendering and here all AJAX requests, DOM or state updates, and set up eventListeners should occur.

componentWillReceiveProps : Executed when particular prop updates to trigger state transitions.

shouldComponentUpdate : Determines if the component will be updated or not. By default it returns true. If you are sure that the component doesn't need to render after state or props are updated, you can return false value. It is a great place to improve performance as it allows you to prevent a rerender if component receives new prop.

componentWillUpdate : Executed before re-rendering the component when there are pros & state changes confirmed by shouldComponentUpdate which returns true.

componentDidUpdate : Mostly it is used to update the DOM in response to prop or state changes.

componentWillUnmount : It will be used to cancel any outgoing network requests, or remove all event listeners associated with the component.

16 . What is Redux?

It is one of the most in-demand libraries for front-end development in today's growing world. It is defined as the predictable state container mainly designed for JavaScript apps and also it is used for managing the entire state of an application. Redux is very small in size and has no dependencies. It builds applications that are easy to deploy in different environments and easy to test. Redux is very small in size and has no dependencies.

17 . What is Use of Redux thunk?

Redux thunk acts as middleware which allows an individual to write action creators that return functions instead of actions. This is also used as a delay function in order to delay dispatch of an action if a certain condition is met. The two store methods getState() and dispatch() are provided as parameters to the inner function.

In order to activate Redux thunk, we must first use applyMiddleware() method as shown below:

```
import{ createStore, applyMiddleware } from 'redux';
import thunk from 'redux-thunk';
import rootReducer from './reducers/index';

//Note: this API requires redux@>=3.1.0

const store= createStore(
  rootReducer,
  applyMiddleware(thunk)
);
```

18 . What do you know about Flux?

Basically, Flux is a basic illustration that is helpful in maintaining the unidirectional data stream. It is meant to control construed data unique fragments to make them interface with that data without creating issues. Flux configuration is insipid; it's not specific to React applications, nor is it required to collect a React application. Flux is basically a straightforward idea, however in you have to exhibit a profound comprehension of its usage.

19 . What are Synthetic events in React?

React implements Synthetic events to improve the consistency and performance of applications and interfaces. The synthetic event is a cross-browser wrapper around the browser's native event. It combines the behaviour of multiple browsers into a single API to make sure events have the same properties across different browsers and platforms.

20 . What are forms in React?

React employs forms to enable users to interact with web applications.

Using forms, users can interact with the application and enter the required information whenever needed. Form contain certain elements, such as text fields, buttons, checkboxes, radio buttons, etc.,

Forms are used for many different tasks such as user authentication, searching, filtering, indexing, etc.,

21 . What is a higher-order component in React?

A Higher-Order Component (HOC) is a function that takes a component and returns a new component, simple and enables re-usability. They are generally used when multiple components have to use a common logic.

We call them pure components because they can accept any dynamically provided child component but they won't modify or copy any behavior from their input components.

```
const EnhancedComponent = higherOrderComponent(WrappedComponent)
```

HOC can be used for many use cases:

- * Code reuse, logic and bootstrap abstraction.
- * Render hijacking.
- * State abstraction and manipulation.
- * Props manipulation.

22 . What is React Router?

React Router is a routing library built on top of React, which is used to create routes in a React application.

23 . Why do we need to React Router?

It maintains consistent structure and behavior and is used to develop single-page web applications. Enables multiple views in a single application by defining multiple routes in the React application.

24 . What is React Fiber?

Fiber is the new *reconciliation* engine or reimplementation of core algorithm in React v16. The goal of React Fiber is to increase its suitability for areas like animation, layout, gestures, ability to pause, abort, or reuse work and assign priority to different types of updates; and new concurrency primitives.

25 . What is the main goal of React Fiber?

The goal of React Fiber is to increase its suitability for areas like animation, layout, and gestures. Its headline feature is incremental rendering: the ability to split rendering work into chunks and spread it out over multiple frames.

26 . What is the difference between createElement and cloneElement?

JSX elements will be transpiled to React.createElement() functions to create React elements which are going to be used for the object representation of UI. Whereas cloneElement is used to clone an element and pass it new props.

27 . How to create props proxy for HOC component?

You can add/edit props passed to the component using props proxy pattern like this:

```
function HOC(WrappedComponent) {  
  return class Test extends Component {
```

```
render() {
  const newProps = {
    title: 'New Header',
    footer: false,
    showFeatureX: false,
    showFeatureY: true
  }

  return <WrappedComponent {...this.props} {...newProps} />
}
}
}
```

28 . What is context?

Context provides a way to pass data through the component tree without having to pass props down manually at every level.

For example, authenticated user, locale preference, UI theme need to be accessed in the application by many components.

```
const {Provider, Consumer} = React.createContext(defaultValue)
```

29 . What is reconciliation?

When a component's props or state change, React decides whether an actual DOM update is necessary by comparing the newly returned element with the previously rendered one. When they are not equal, React will update the DOM. This process is called reconciliation.

30 . Why React uses className over class attribute?

class is a keyword in JavaScript, and JSX is an extension of JavaScript. That's the principal reason why React uses className instead of class. Pass a string as the className prop.

```
render() {
  return <span className={'menu navigation-menu'}>{'Menu'}</span>
}
```

31 . List some of the cases when you should use Refs.

Following are the cases when refs should be used:

- * When you need to manage focus, select text or media playback
- * To trigger imperative animations
- * Integrate with third-party DOM libraries

32 . What were the major problems with MVC framework?

- * DOM manipulation was very expensive
- * Applications were slow and inefficient
- * There was huge memory wastage
- * Because of circular dependencies, a complicated model was created around models and views

33 . How are Actions defined in Redux?

Actions in React must have a type property that indicates the type of ACTION being performed. They must be defined as a String constant and you can add more properties to it as well. In Redux, actions are created using the functions called Action Creators. Below is an example of Action and Action Creator:

```
function addTodo(text) {
  return {
    type: ADD_TODO,
    text
  }
}
```

34 . Explain the role of Reducer.

Reducers are pure functions which specify how the application's state changes in response to an ACTION. Reducers work by taking in the previous state and action, and then it returns a new state. It determines what sort of update needs to be done based on the type of the action, and then returns new values. It returns the previous state as it is, if no work needs to be done.

35 . Why is switch keyword used in React Router v4?

Although a <div> is used to encapsulate multiple routes inside the Router. The 'switch' keyword is used when you want to display only a single route to be rendered amongst the several defined routes. The <switch> tag when in use matches the typed URL with the defined routes in sequential order. When the first match is found, it renders the specified route. Thereby bypassing the remaining routes.

36 . How do you tell React to build in Production mode and what will that do?

Ordinarily, you'd utilize Webpack's DefinePlugin strategy to set NODE_ENV to production. This will strip out things like prototype approval and additional notices. Over that, it's likewise a smart thought to minify your code in light of the fact that React utilizes Uglify's dead-code end to strip out advancement just code and remarks, which will radically diminish the measure of your package.

37 . What do you understand with the term polling?

The server needs to be monitored to for updates with respect to time. The primary aim in most of the cases is to check whether novel comments are there or not. This process is basically considered as pooling. It checks for updates approximately every 5 seconds. It is possible to change this time period easily. Pooling help keeping an eye on the users and always make sure that no negative information is present on the servers. Actually, it can create issues related to several things and thus pooling is considered.

38 . What do you mean by virtual DOM?

For all the available DOM objects in ReactJS, there is a parallel virtual DOM object. It is nothing but can be considered as the lighter version of the true copy and is powerful in eliminating the complex code. It is also used as a Blue Print for performing several basic experiments. Many developers also use it while practicing this technology.

39 . Compare MVC with Flux?

MVC approaches are presently considered as outdated. Although they are capable to handle data concerns, controllers as well as UI, many developers found that it doesn't properly work when applications size increases. However, they are capable to handle some of the key issues such as eliminating the lack of data integrity as well as managing the data flow which is not properly defined. On the other side, Flux works perfectly with all the sizes irrespective of their size.

40 . How error boundaries handled in React v15?

React v15 provided very basic support for error boundaries using `unstable_handleError` method. It has been renamed to `componentDidCatch` in React v16.

41 . What are the recommended ways for static type checking?

Normally we use PropTypes library (`React.PropTypes` moved to a `prop-types` package since React v15.5) for type checking in the React applications. For large code bases, it is recommended to use static type checkers such as Flow or TypeScript, that perform type checking at compile time and provide auto-completion features.

42 . What is the use of react-dom package?

The react-dom package provides DOM-specific methods that can be used at the top level of your app. Most of the components are not required to use this module. Some of the methods of this package are:

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

43 . What is the purpose of render method of react-dom?

This method is used to render a React element into the DOM in the supplied container and return a reference to the component. If the React element was previously rendered into container, it will perform an update on it and only mutate the DOM as necessary to reflect the latest changes.

```
ReactDOM.render(element, container[, callback])
```

If the optional callback is provided, it will be executed after the component is rendered or updated.

44 . How to use innerHTML in React?

The `dangerouslySetInnerHTML` attribute is React's replacement for using `innerHTML` in the browser DOM. Just like `innerHTML`, it is risky to use this attribute considering cross-site scripting (XSS) attacks. You just need to pass a `_html` object as key and HTML text as value.

In this example `MyComponent` uses `dangerouslySetInnerHTML` attribute for setting HTML markup:

```
function createMarkup() {
  return { __html: 'First &middot; Second' }
}

function MyComponent() {
  return <div dangerouslySetInnerHTML={createMarkup()} />
}
```

45 . Is it good to use `setState()` in `componentWillMount()` method?

It is recommended to avoid async initialization in `componentWillMount()` lifecycle method. `componentWillMount()` is invoked immediately before mounting occurs. It is called before `render()`, therefore setting state in this method will not trigger a re-render. Avoid introducing any side-effects or subscriptions in this method. We need to make sure async calls for component initialization happened in `componentDidMount()` instead of `componentWillMount()`.

```
componentDidMount() {
  axios.get(`api/todos`)
    .then((result) => {
      this.setState({
        messages: [...result.data]
      })
    })
}
```

46 . How do you memoize a component?

There are memoize libraries available which can be used on function components.

For example moize library can memoize the component in another component.

```
import moize from 'moize'
import Component from './components/Component' // this module exports a non-memoized component

const MemoizedFoo = moize.react(Component)

const Consumer = () => {
  <div>
    {'I will memoize the following entry:'}
    <MemoizedFoo/>
  </div>
}
```

Update : Since React v16.6.0, we have a `React.memo`. It provides a higher order component which memoizes component unless the props change. To use it, simply wrap the component using `React.memo` before you use it.

```
const MemoComponent = React.memo(function MemoComponent(props) {  
  /* render using props */  
});  
OR  
export default React.memo(MyFunctionComponent);
```

47 . How you implement Server Side Rendering or SSR?

React is already equipped to handle rendering on Node servers. A special version of the DOM renderer is available, which follows the same pattern as on the client side.

```
import ReactDOMServer from 'react-dom/server'  
import App from './App'  
  
ReactDOMServer.renderToString(<App />)
```

This method will output the regular HTML as a string, which can be then placed inside a page body as part of the server response. On the client side, React detects the pre-rendered content and seamlessly picks up where it left off.

48 . What is the purpose of getDerivedStateFromProps() lifecycle method?

The new static getDerivedStateFromProps() lifecycle method is invoked after a component is instantiated as well as before it is re-rendered. It can return an object to update state, or null to indicate that the new props do not require any state updates.

```
class MyComponent extends React.Component {  
  static getDerivedStateFromProps(props, state) {  
    // ...  
  }  
}
```

This lifecycle method along with componentDidUpdate() covers all the use cases of componentWillReceiveProps()

49 . What are React Mixins?

Mixins are a way to totally separate components to have a common functionality. Mixins should not be used and can be replaced with higher-order components or decorators.

One of the most commonly used mixins is PureRenderMixin. You might be using it in some components to prevent unnecessary re-renders when the props and state are shallowly equal to the previous props and state:

```
const PureRenderMixin = require('react-addons-pure-render-mixin')  
  
const Button = React.createClass({  
  mixins: [PureRenderMixin],  
  // ...  
})
```

50 . Can you force a component to re-render without calling setState?

By default, when your component's state or props change, your component will re-render. If your render() method depends on some other data, you can tell React that the component needs re-rendering by calling forceUpdate().

```
component.forceUpdate(callback)
```

It is recommended to avoid all uses of forceUpdate() and only read from this.props and this.state in render().

51 . What is the difference between React and ReactDOM?

The **react** package contains React.createElement(), React.Component, React.Children, and other helpers related to elements and component classes. You can think of these as the isomorphic or universal helpers that you need to build components.

The **react-dom** package contains ReactDOM.render(), and in react-dom/server we have server-side rendering support with ReactDOMServer.renderToString() and ReactDOMServer.renderToStaticMarkup().

52 . How to use React label element?

If you try to render a <label> element bound to a text input using the standard for attribute, then it produces HTML missing that attribute and prints a warning to the console.

```
<label for={'user'}>'User'</label>
<input type={'text'} id={'user'} />
```

Since for is a reserved keyword in JavaScript, use htmlFor instead.

```
<label htmlFor={'user'}>'User'</label>
<input type={'text'} id={'user'} />
```

53 . How to combine multiple inline style objects?

You can use spread operator in regular React:

```
<button style={{...styles.panel.button, ...styles.panel.submitButton}}>'Submit'</button>
```

If you're using React Native then you can use the array notation:

```
<button style={[styles.panel.button, styles.panel.submitButton]}>'Submit'</button>
```

54 . How can we find the version of React at runtime in the browser?

You can use React.version to get the version.

```
const REACT_VERSION = React.version
```

```
ReactDOM.render(  
  <div>`React version: ${REACT_VERSION}`</div>,  
  document.getElementById('app')  
)
```

55 . How to use https instead of http in create-react-app?

You just need to use HTTPS=true configuration. You can edit your package.json scripts section:

```
"scripts": {  
  "start": "set HTTPS=true && react-scripts start"  
}
```

or just run set HTTPS=true && npm start

56 . How to add Google Analytics for React Router?

Add a listener on the history object to record each page view:

```
history.listen(function (location) {  
  window.ga('set', 'page', location.pathname + location.search)  
  window.ga('send', 'pageview', location.pathname + location.search)  
})
```

57 . What are the popular React-specific linters?

ESLint is a popular JavaScript linter. There are plugins available that analyse specific code styles. One of the most common for React is an npm package called eslint-plugin-react. By default, it will check a number of best practices, with rules checking things from keys in iterators to a complete set of prop types.

Another popular plugin is eslint-plugin-jsx-a11y, which will help fix common issues with accessibility. As JSX offers slightly different syntax to regular HTML, issues with alt text and tabindex, for example, will not be picked up by regular plugins.

58 . What is the purpose of push() and replace() methods of history?

A history instance has two methods for navigation purpose.

- * **push()**
- * **replace()**

If you think of the history as an array of visited locations, push() will add a new location to the array and replace() will replace the current location in the array with the new one.

59 . How to implement default or NotFound page?

A <Switch> renders the first child <Route> that matches. A <Route> with no path always matches. So you just need to simply drop path attribute as below

```
<Switch>
  <Route exact path="/" component={Home}/>
  <Route path="/user" component={User}/>
  <Route component={NotFound} />
</Switch>
```

60 . How to perform automatic redirect after login?

The react-router package provides `<Redirect>` component in React Router. Rendering a `<Redirect>` will navigate to a new location. Like server-side redirects, the new location will override the current location in the history stack.

```
import React, { Component } from 'react'
import { Redirect } from 'react-router'

export default class LoginComponent extends Component {
  render() {
    if (this.state.isLoggedIn === true) {
      return <Redirect to="/your/redirect/page" />
    } else {
      return <div>{'Login Please'}</div>
    }
  }
}
```

61 . What is React Intl?

The React Intl library makes internalization in React straightforward, with off-the-shelf components and an API that can handle everything from formatting strings, dates, and numbers, to pluralization. React Intl is part of FormatJS which provides bindings to React via its components and API.

62 . How many outermost elements can be there in a JSX expression?

It must have one JSX element present so that the task can be accomplished easily. Having more than one expression is not an issue but probably it will slow down the process. There are also chances of confusion with more than one expression if you are new to this technology.

63 . What are advantages of using React Hooks? What are advantages of using React Hooks?

Primarily, hooks in general enable the extraction and reuse of stateful logic that is common across multiple components without the burden of higher order components or render props. Hooks allow to easily manipulate the state of our functional component without needing to convert them into class components.

Hooks don't work inside classes (because they let you use React without classes). By using them, we can totally avoid using lifecycle methods, such as `componentDidMount`, `componentDidUpdate`, `componentWillUnmount`. Instead, we will use built-in hooks like `useEffect` .

64 . What is useState() in React?

Explain what is the use of useState(0) there:

```
...
const [count, setCounter] = useState(0);
const [moreStuff, setMoreStuff] = useState(...);
...

const setCount = () => {
  setCounter(count + 1);
  setMoreStuff(...);
  ...
};

};
```

useState is one of build-in react hooks. useState(0) returns a tuple where the first parameter count is the current state of the counter and setCounter is the method that will allow us to update the counter's state.

We can use the setCounter method to update the state of count anywhere - In this case we are using it inside of the setCount function where we can do more things; the idea with hooks is that we are able to keep our code more functional and avoid class based components if not desired/needed.

65 . What is the purpose of ReactTestUtils package?

ReactTestUtils are provided in the with-addons package and allow you to perform actions against a simulated DOM for the purpose of unit testing.

66 . What is Jest?

Jest is a JavaScript unit testing framework created by Facebook based on Jasmine and provides automated mock creation and a jsdom environment. It's often used for testing components.

67 . What is the difference between React Native and React?

React is a JavaScript library, supporting both front end web and being run on the server, for building user interfaces and web applications.

React Native is a mobile framework that compiles to native app components, allowing you to build native mobile applications (iOS, Android, and Windows) in JavaScript that allows you to use React to build your components, and implements React under the hood.

68 . What is Relay?

Relay is a JavaScript framework for providing a data layer and client-server communication to web applications using the React view layer.

69 . What are the controlled components and uncontrolled components in ReactJS ?

Controlled component is more advisable to use as it is easier to implement forms in it. In this, form data are handled by React components. A controlled input accepts values as props and callbacks to change that value.

uncontrolled component is a substitute for controlled components. Here form data is handled by DOM itself. In uncomfortable components, the ref can be used to get the form values from DOM.

70 . How to make AJAX ask for in Redux?

You can utilize redux-thud middleware which enables you to characterize async activities.

How about we take a case of bringing explicit record as an AJAX call utilizing get API:

```
trade work fetchAccount(id) {
  return dispatch => {
    dispatch(setLoadingAccountState())/Show a stacking spinner
    get('/account/${id}', (reaction) => {
      dispatch(doneFetchingAccount())/Hide stacking spinner
      on the off chance that (response.status === 200) {
        dispatch(setAccount(response.json))/Use an ordinary capacity to set the got state
      } else {
        dispatch(someError)
      }
    })
  }
}

work setAccount(data) {
  return { type: 'SET_Account', information: information }
}â€¢
```

Codewitharrays.in 8007592194



<https://www.youtube.com/@codewitharrays>



<https://www.instagram.com/codewitharrays/>



<https://t.me/codewitharrays> Group Link: <https://t.me/ccee2025notes>



[+91 8007592194 +91 9284926333](#)



codewitharrays@gmail.com



<https://codewitharrays.in/project>