1. Difference between html5 and other html versions.

Ans :

Html	Html5
A hypertext markup language (HTML) is the primary language for developing web pages.	HTML5 is a new version of HTML with new functionalities with markup language with Internet technologies.
Language in HTML does not have support for video and audio.	HTML5 supports both video and audio.
The HTML browser uses cache memory as temporary storage.	HTML5 has the storage options like:application cache, SQL database, and web storage.
HTML is compatible with almost all browsers because it has been present for a long time, and the browser made modifications to support all the features.	In HTML5, we have many new tags, elements, and some tags that have been removed/modified, so only some browsers are fully compatible with HTML5.
In HTML, vector graphics are possible with tools LikeSilver light, Adobe Flash, VML, etc.	In HTML5, vector graphics are supported by default.
In HTML, the browser interface and JavaScript running in the same thread.	The HTML5 has the JavaScript Web Worker API, which allows the browser interface to run in multiple threads.
Uses cookies to store data.	Uses local storage instead of cookies
Html5 uses cookies.	It supplies local storage in place of cookies.
Works with all older browsers	A new browser supports this.

2. Inline css or external file .which one is good.

Ans : External css.

3. Where to use # in css

Ans : The #id selector allows you to target an element by referencing the id HTML attribute. Similar to how class attributes are denoted in CSS with a “period” (.) before the class name, ID attributes are prefixed with an “octothorpe” (#), more commonly known as a “hash” or “pound sign”.

4. How can you make your page responsive

Ans : The Viewport setting allows you to insert a meta tag on all of your web pages to make your site responsive.

- Responsive Images.
- Responsive Text Size.
- Media Queries.
- A responsive web page with an example of navigation.
  It is important to provide access to responsive web design and framework services.
  Bootstrap.

5. Html5 elements

Ans :

<abbr>	Defines an abbreviated form of a longer word or phrase.
<acronym>	Defines an acronym. Use <abbr> instead.
<address>	Specifies the author's contact information.
<applet>	Embeds a Java applet (mini Java applications) on the page. Use <object> instead.
<area>	Defines a specific area within an image map.
<article>	Defines an article.
<aside>	Defines some content loosely related to the page content.
<audio>	Embeds a sound, or an audio stream in an HTML document.
<b>	Displays text in a bold style.
<base>	Defines the base URL for all relative URLs in a document.
<basefont>	Specifies the base font for a page. Use CSS instead.
<bdi>	Represents text that is isolated from its surrounding for the purposes of bidirectional text formatting.
<bdo>	Overrides the current text direction.
<big>	Displays text in a large size. Use CSS instead.
<blockquote>	Represents a section that is quoted from another source.
<body>	Defines the document's body.
<br>	Produces a single line break.
<button>	Creates a clickable button.
<canvas>	Defines a region in the document, which can be used to draw graphics on the fly via scripting (usually JavaScript).
<caption>	Defines the caption or title of the table.
<center>	Align contents in the center. Use CSS instead.
<cite>	Indicates a citation or reference to another source.
<code>	Specifies text as computer code.
<col>	Defines attribute values for one or more columns in a table.
<colgroup>	Specifies attributes for multiple columns in a table.
<data>	Links a piece of content with a machine-readable translation.

<datalist>	Represents a set of pre-defined options for an <input> element.
<dd>	Specifies a description, or value for the term (<dt>) in a description list (<dl>).
<del>	Represents text that has been deleted from the document.
<details>	Represents a widget from which the user can obtain additional information or controls on-demand.
<dfn>	Specifies a definition.
<dialog>	Defines a dialog box or subwindow.
<dir>	Defines a directory list. Use <ul> instead.
<div>	Specifies a division or a section in a document.
<dl>	Defines a description list.
<dt>	Defines a term (an item) in a description list.
<em>	Defines emphasized text.
<embed>	Embeds external application, typically multimedia content like audio or video into an HTML document.
<fieldset>	Specifies a set of related form fields.
<figcaption>	Defines a caption or legend for a figure.
<figure>	Represents a figure illustrated as part of the document.
<font>	Defines font, color, and size for text. Use CSS instead.
<footer>	Represents the footer of a document or a section.
<form>	Defines an HTML form for user input.
<frame>	Defines a single frame within a frameset.
<frameset>	Defines a collection of frames or other frameset.
<head>	Defines the head portion of the document that contains information about the document such as title.
<header>	Represents the header of a document or a section.
<hgroup>	Defines a group of headings.
<h1> to <h6>	Defines HTML headings.
<hr>	Produce a horizontal line.
<html>	Defines the root of an HTML document.
<i>	Displays text in an italic style.
<iframe>	Displays a URL in an inline frame.
<img>	Represents an image.
<input>	Defines an input control.
<ins>	Defines a block of text that has been inserted into a document.
<kbd>	Specifies text as keyboard input.
<keygen>	Represents a control for generating a public-private key pair.
<label>	Defines a label for an <input> control.
<legend>	Defines a caption for a <fieldset> element.
<li>	Defines a list item.
<link>	Defines the relationship between the current document and an external resource.
<main>	Represents the main or dominant content of the document.
<map>	Defines a client-side image-map.
<mark>	Represents text highlighted for reference purposes.
<menu>	Represents a list of commands.
<menuitem>	Defines a list (or menuitem) of commands that a user can perform.
<meta>	Provides structured metadata about the document content.
<meter>	Represents a scalar measurement within a known range.
<nav>	Defines a section of navigation links.

<noframes>	Defines an alternate content that displays in browsers that do not support frames.
<noscript>	Defines alternative content to display when the browser doesn't support scripting.
<object>	Defines an embedded object.
<ol>	Defines an ordered list.
<optgroup>	Defines a group of related options in a selection list.
<option>	Defines an option in a selection list.
<output>	Represents the result of a calculation.
<p>	Defines a paragraph.
<param>	Defines a parameter for an object or applet element.
<picture>	Defines a container for multiple image sources.
<pre>	Defines a block of preformatted text.
<progress>	Represents the completion progress of a task.
<q>	Defines a short inline quotation.
<rp>	Provides fall-back parenthesis for browsers that that don't support ruby annotations.
<rt>	Defines the pronunciation of character presented in a ruby annotations.
<ruby>	Represents a ruby annotation.
<s>	Represents contents that are no longer accurate or no longer relevant.
<samp>	Specifies text as sample output from a computer program.
<script>	Places script in the document for client-side processing.
<section>	Defines a section of a document, such as header, footer etc.
<select>	Defines a selection list within a form.
<small>	Displays text in a smaller size.
<source>	Defines alternative media resources for the media elements like <audio> or <video>.
<span>	Defines an inline styleless section in a document.
<strike>	Displays text in strikethrough style.
<strong>	Indicate strongly emphasized text.
<style>	Inserts style information (commonly CSS) into the head of a document.
<sub>	Defines subscripted text.
<summary>	Defines a summary for the <details> element.
<sup>	Defines superscripted text.
<svg>	Embed SVG (Scalable Vector Graphics) content in an HTML document.
<table>	Defines a data table.
<tbody>	Groups a set of rows defining the main body of the table data.
<td>	Defines a cell in a table.
<template>	Defines the fragments of HTML that should be hidden when the page is loaded, but can be cloned and inserted in the document by JavaScript.
<textarea>	Defines a multi-line text input control (text area).
<tfoot>	Groups a set of rows summarizing the columns of the table.
<th>	Defines a header cell in a table.
<thead>	Groups a set of rows that describes the column labels of a table.
<time>	Represents a time and/or date.
<title>	Defines a title for the document.
<tr>	Defines a row of cells in a table.
<track>	Defines text tracks for the media elements like <audio> or <video>.

<tt>	Displays text in a teletype style.
<u>	Displays text with an underline.
<ul>	Defines an unordered list.
<var>	Defines a variable.
<video>	Embeds video content in an HTML document.
<wbr>	Represents a line break opportunity

6. What are cookies in backend

Ans : Cookies are tiny pieces of data that the backend can store in the user's browsers. User tracking, personalization, and most important, authentication, are the most common use cases for cookies. Cookies have a lot of privacy concerns, and have been subject to strict regulation over the years.

7. What is json

Ans :

- JSON stands for JavaScript Object Notation
- JSON is a lightweight format for storing and transporting data
- JSON is often used when data is sent from a server to a web page
- JSON is "self-describing" and easy to understand
- JSON data is written as name/value pairs, just like JavaScript object properties.

8. Understand SOAP.

Ans : SOAP is an acronym for Simple Object Access Protocol. It is an XML-based messaging protocol for exchanging information among computers. SOAP is an application of the XML specification.

- SOAP is a communication protocol designed to communicate via Internet.
- SOAP can extend HTTP for XML messaging.
- SOAP provides data transport for Web services.
- SOAP can exchange complete documents or call a remote procedure.
- SOAP can be used for broadcasting a message.
- SOAP is platform- and language-independent.
- SOAP is the XML way of defining what information is sent and how.
- SOAP enables client applications to easily connect to remote services and invoke remote methods.

9. What is Node.js

Ans : Node.js is an open-source, cross-platform, JavaScript runtime environment that executes JavaScript code outside of a web browser. Node.js is a popular, lightweight web framework for beginners, and it is used by many big companies like Netflix and Uber.

Node.js is an important tool for any JavaScript developer to understand.

10. React vs angular

Ans :

- AngularJS is a structural framework for developing dynamic web apps, whereas React is a javascript library that allows you to build UI components.
- Angular JS is based on MVC (Model View Controller) whereas React is based on Virtual DOM.
- Angular is based on Typescript and React is based on Javascript.
- AngularJS doesn't provide adding javascript library to the source code while React allows adding javascript library to the source code.
- AngularJS provides testing and debugging for a complete project with a single tool whereas React requires a set of tools to perform different types of testing.

11. Es 5 vs Es 6

Ans :

ES5	ES6
ES5 is the fifth edition of the ECMAScript (a trademarked scripting language specification defined by ECMA International)	ES6 is the sixth edition of the ECMAScript(a trademarked scripting language specification defined by ECMA International).
It was introduced in 2009.	It was introduced in 2015.
ES5 supports primitive data types that are string, number, boolean, null, and undefined.	In ES6, there are some additions to JavaScript data types. It introduced a new primitive data type 'symbol' for supporting unique values.
In ES5, we could only define the variables by using the var keyword.	In ES6, there are two new ways to define variables that are let and const.
As ES5 is prior to ES6, there is a non-presence of some features, so it has a lower performance than ES6.	Because of new features and the shorthand storage implementation ES6 has a higher performance than ES5.
A wide range of communities supports it.	It also has a lot of community support, but it is lesser than ES5.
ES5 is time-consuming than ES6.	Due to destructuring and speed operators, object manipulation can be processed more smoothly in ES6.
In ES5, both function and return keywords are used to define a function.	An arrow function is a new feature introduced in ES6 by which we don't require the function

	keyword to define the function.
In ES5, there is a use of for loop to iterate over elements.	ES6 introduced the concept of for...of loop to perform an iteration over the values of the iterable objects.

12. What is Rest controller and controller

Ans : The @Controller is a common annotation that is used to mark a class as Spring MVC Controller while @RestController is a special controller used in RESTFul web services and the equivalent of @Controller + @ResponseBody.

The @Controller annotation indicates that the class is a “Controller” e.g. a web controller while the @RestController annotation indicates that the class is a controller where @RequestMapping methods assume @ResponseBody semantics by default i.e. servicing REST API.

13. Library vs framework

Ans : The primary difference between a library and a framework is the “inversion Control.” Simply put, this refers to who is in control of the programming process. With a code library, the developer can make a call to the library whenever they want. However, with a framework, the developer is fully incorporated in its workflow.

14. What is responsive web application and how to create it

Ans : A responsive web app refers to a design that responds effectively to the behavior of the users and the environment depending on the size of the screen, orientation and platform. A responsive web app design has a wide range of flexible layouts, grids and images. When users want to switch from the iPad to laptop, the responsive web application switches automatically to accommodate for resolution, scripting abilities and image size. It has the technology to allow it respond automatically to suit the preferences of the users.

Responsive web application development does not only focus on the adjustability of the screen resolutions and resizing of images.

15. How the traffic is divided from front end and back end

Ans : Securing front-end and back-end traffic
You can use an SSL certificate to secure the traffic that flows between the ISA server and the OWA front-end server. However, you can’t use SSL to encrypt the traffic flowing between the OWA front-end and the Exchange back-end servers. Instead, you’ll have to rely on the IPSec protocol.

16. How does javascript run in all browsers

Ans : JavaScript is an interpreted language where code is explained and executed at the runtime. Additionally, we know that web browsers understand HTML and CSS and converting those languages into a visual display on the screen. The part of the web browser that understands HTML and CSS is called the rendering engine. However, most browsers also have a *JavaScript interpreter*. That’s the part of the browser that understands JavaScript and run JavaScript programs.