



# INSTITUTE OF AERONAUTICAL ENGINEERING (Autonomous)

Dundigal, Hyderabad - 500 043

## COMPUTER SCIENCE AND ENGINEERING(AI & ML)

### DEFINITION AND TERMINOLGY

Course Title	WEB APPLICATION DEVELOPMENT				
Course Code	AITC09				
Program	B.Tech				
Semester	IV	CSE(AI & ML)			
Course Type	Core				
Regulation	UG20				
Course Structure	Theory			Practical	
	Lecture	Tutorials	Credits	Laboratory	Credits
	3	-	3	-	1.5
Course Coordinator	Mrs. M. Kalaiarasi, Assistant Professor				

### COURSE OBJECTIVES:

The students will try to learn:

I	The characteristics, systematic methods, model for developing web applications.
II	The fundamentals of HTML and CSS to design static and dynamic web pages.
III	The concepts of client side programming with Bootstrap ,JavaScript, Ajax , JSX.
IV	The MVC architecture, about Reactjs and built single and multiple page applications using REACT with REDUX.

### COURSE OUTCOMES:

After successful completion of the course, students should be able to:

CO 1	<b>Summarize</b> HTML elements and attributes for structuring and presenting content of webpage based on the user requirement .	Understand
CO 2	<b>Make use of</b> CSS properties, for formatting webpages.	Apply
CO 3	<b>Develop</b> responsive webpage using Bootstrap for viewing web pages in various devices.	Apply
CO 4	<b>Utilize</b> the concepts of JS with event actions for displaying information on webpages.	Apply
CO 5	<b>Identify</b> UI binding library elements for deploying a reusable complex UI.	Apply
CO 6	<b>Develop</b> a native web application with the help of React framework.	Apply

## DEFINITION AND TERMINOLOGY:

S.No	DEFINITION	CO's
MODULE I		
INTRODUCTION TO WEB APPLICATION AND HYPERTEXT MODELLING		
1	<b>Are the HTML tags and elements the same thing?</b>	CO 1
	No. HTML elements are defined by a starting tag, may contain some content and a closing tag. For example, <code>&lt;h1&gt;Heading 1&lt;/h1&gt;</code> is a HTML element but just <code>&lt;h1&gt;</code> is a starting tag and <code>&lt;/h1&gt;</code> is a closing tag.	
2	<b>What are HTML Entities?</b>	CO 2
	In HTML some characters are reserved like ' <code>&lt;</code> ', ' <code>&gt;</code> ', ' <code>/</code> ', etc. To use these characters in our webpage we need to use the character entities called HTML Entities. Below are a few mapping between the reserved character and its respective entity character to be used.	
3	<b>What is the 'class' attribute in HTML?</b>	CO 2
	The class attribute is used to specify the class name for an HTML element. Multiple elements in HTML can have the same class value. Also, it is mainly used to associate the styles written in the stylesheet with the HTML elements.	
4	<b>What is the difference between the 'id' attribute and the 'class' attribute of HTML elements?</b>	CO 2
	Multiple elements in HTML can have the same class value, whereas a value of id attribute of one element cannot be associated with another HTML element.	
5	<b>What are the different kinds of Doctypes available?</b>	CO 2
	The three kinds of Doctypes which are available: Strict Doctype Transitional Doctype Frameset Doctype	
6	<b>How can we include audio or video in a webpage?</b>	CO 1
	HTML5 provides two tags: <code>&lt;audio&gt;</code> and <code>&lt;video&gt;</code> tags using which we can add the audio or video directly in the webpage.	

7	<b>What is CSS?</b>	CO 2
	CSS stands for cascading style sheet. It is the web designing language used for formatting the look and structure of the web page written in HTML(Hypertext markup language). The application is known as XHTML. CSS is used with different XML documents such as plain XML, SVG, and UXL. It makes the page attractive and presentable. It is mainly used with HTML and Javascript for developing user interfaces in web applications and mobile applications	
8	<b>How to apply a background color to a specific element.</b>	CO 2
	The background-color property is used to specify the background color of the element..	
9	<b>What is the universal selector?</b>	CO 2
	The universal selector is used for selecting all the HTML elements, and it applies the style declaration to all the elements.	
10	<b>What is the use of opacity in CSS?</b>	CO 2
	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.	
11	<b>What is W3C?.</b>	CO 1
	W3C is the world wide web consortium. Its main work is to deliver the information to the worldwide web. It develops the guideline and rules for the web.	
12	<b>Explain the logical tag</b>	CO 1
	Logical tags are mainly used to indicate by the visually impaired and put emphasis on the text. They are useless for appearances. They are old and concentrate on the content.	
13	<b>Explain the physical tag..</b>	CO 2
	Physical tags are mainly used to indicate how a particular character is to be formatted. They are also referred to as presentational markup. It is a newer version.	
14	<b>What is the origin of CSS?</b>	CO 2
	SGML (Standard Generalized Markup Language) is the origin of CSS. It is a language that defines markup languages.	
15	<b>Why background and color are the separate properties if they should always be set together?.</b>	CO 2
	There are two reasons behind this: It enhances the legibility of style sheets. The background property is a complex property in CSS, and if it is combined with color, the complexity will further increase. Color is an inherited property while the background is not. So this can make confusion further. .	
16	<b>What is RWD?.</b>	CO 2

	RWD stands for Responsive Web Design. This technique is used to display the designed page perfectly on every screen size and device, for example, mobile, tablet, desktop and laptop. You don't need to create a different page for each device.	
17	<b>What are the benefits of CSS sprites?</b> If a web page has a large number of images that take a longer time to load because each image separately sends out an HTTP request. 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	CO 1
18	<b>What is the difference between logical tags and physical tags?.</b> Physical tags are referred to as presentational markup while logical tags are useless for appearances. Physical tags are newer versions, on the other hand, logical tags are old and concentrate on content.	CO 1
19	<b>Nesting Elements and Indentation?</b> We've seen in previous examples that we can nest elements inside of tags. In fact, this is a major part of writing HTML. Nested elements are referred to as children and the top level elements as parents. You will see this type of structure throughout your time writing HTML (and all code for that matter). One thing to note is that while indentation is not a requirement of HTML (nor of CSS and JavaScript), it is still often practiced as a readability issue. We not only write code for ourselves, but also for our teammates and those that come after us. Indentation lets your code be much more easily read and understood.	CO 1
<b>MODULE II</b>		
<b>BUILD INTERFACES USING BOOTSTRAP</b>		
1	<b>What is a Bootstrap Container, and how does it work?</b> A bootstrap container is a handy class that generates a central region on the page where we can put our site content. The bootstrap .container has the advantage of being responsive and containing all of our other HTML code. Containers are used to pad the content within them, and there are two types of containers: The .container class creates a fixed-width container that is responsive. The .container-fluid class creates a full-width container that spans the entire viewport width	CO 3
2	<b>What are the default Bootstrap text settings??</b>	CO 3

The default font size in Bootstrap 4 is 16px, with a line-height of 1.5.
--

The default font family is "Helvetica Neue," which includes Helvetica, Arial, and other sans-serif fonts.
---

Margin-top: 0 and margin-bottom: 1rem are also set on all <code>ip</code> elements (16px by default).
---

3	<b>In Bootstrap 4, what is flexbox? .</b>	CO 3
	Flexbox is a layout module for flexible boxes. Without using float or positioning, you can quickly create a flexible layout design with flexbox.	
4	<b>What contextual classes can be used to style the panels?</b>	CO 5
	To make a panel more meaningful to a specific context, use contextual state classes like panel-primary, panel-success, panel-info, panel-warning, and panel-danger.	
5	<b>What do you mean by the Bootstrap well?</b>	CO 3
	The Bootstrap well is simply a container that makes the content appear sunken. It can also produce an inset effect on the webpage. With the aid of <code>&lt;div&gt;</code> and class, a developer may create a well and also wrap the content in the well. The content will be displayed according to your preferences.	
6	<b>Explain the affix plugin</b>	CO 6
	The affix plugin allows you to affix a <code>&lt;div&gt;</code> to a particular location on the page. You may also use this plugin to turn on and off the pinning. Social icons are a good example of this. They'll start in one position, but when the page reaches a certain point, the <code>&lt;div&gt;</code> will be frozen in place and will no longer scroll with the rest of the page	
7	<b>What are Bootstrap collapsing elements?</b>	CO 3
	You may collapse any element with Bootstrap collapsing elements without creating any JavaScript code or accordion markup. To automatically delegate control of a collapsible element in bootstrap, add <code>data-toggle= "collapse"</code> to the controller element along with a <code>data-target</code> or <code>href</code> . You may also do the same thing with <code>.collapse (options)</code> , <code>.collapse ('show')</code> , or <code>.collapse ('hide')</code> .	
8	<b>In Bootstrap, what is a scrollspy?.</b>	CO 5
	It's an auto-updating nav component that allows you to grab sections of the page based on where you're scrolling. Based on the scroll position, the <code>.active</code> class will update from one nav item to the next	
9	<b>Name key components of Bootstrap 4.</b>	CO 3
	Scaffolding: Bootstrap built on a responsive grid system consists of a 12-column grid. JS Plugins: Contains JS and jQuery plugins. CSS: Contains CSS files	
10	<b>What do you understand by the term container?</b>	CO 3
	Containers are the most fundamental components in Bootstrap. It behaves like a container where you can put HTML code, and it is a part within the page where the content of the site can be placed to make it responsive and fast.	
11	<b>What do you mean by Bootstrap ClassLoader?</b>	CO 3

Bootstrap ClassLoader is a part of Java and a main parental class of class loader
---

12	<b>What is the fluid layout?</b>	CO 3
	The fluid layout is useful when you need to make an app that involves the full width of the screen. The fluid layout adjusts itself according to the browser size	
13	<b>How can you display code in Bootstrap?</b>	CO 3
	Using the <code>&lt;code&gt;</code> tag – This method is used to display code as inline code. Using the <code>&lt;pre&gt;</code> tag – This method is used to display code as a standalone block.	
14	<b>How can you make an image round in Bootstrap 4?</b>	CO 3
	You can use the <code>.rounded</code> class to make an image round.	
15	<b>What is a progress bar?</b>	CO 3
	A progress bar is an indicator to show the progress of a particular process.	
16	<b>What are responsive utility classes in Bootstrap?</b>	CO 3
	Responsive utility classes in Bootstrap are a set of classes that are used to conceal or exhibit the HTML elements based on screen resolution that discerns by media query in Bootstrap. Example: “hidden-md-down”	
17	<b>Define Bootstrap 4 thumbnails.</b>	CO 3
	It is a way to use the layout images, videos, text, etc. in a grid system. We can create thumbnails by adding a tag with the <code>.img-thumbnail</code> class around the image.	
18	<b>Explain the modal in Bootstrap4</b>	CO 3
	A model is an inherited window that is layered over its parent window. This is used to augment the user experience and add different functionalities. Model windows are created with the help of the modal plugin.	
19	<b>Which are the classes used for pagination in Bootstrap4?</b>	CO 3
	The classes used for pagination are the <code>.pagination</code> class, the <code>.page-item</code> class, the <code>.page-link</code> class, the <code>.pagination-sm</code> class, the <code>.pagination-lg</code> class, the <code>.active</code> class and the <code>.disabled</code> class.	
20	<b>Explain what Bootstrap’s collapsing elements is?</b>	CO 3
	It allows you to collapse any particular element without using any JavaScript code. To use this feature in Bootstrap, you have to add <code>data-toggle=”collapse”</code> to the controller element along with a data target to automatically assign the control of a collapsible element. We can use this by writing <code>.collapse(options)</code> etc.	



MODULE III		
INTERACTIVE USER INTERFACE AND WEB APPLICATION DEVELOPMENT		
1	<b>What is JavaScript?</b> JavaScript is a lightweight, interpreted programming language with object-oriented capabilities that allows you to build interactivity into otherwise static HTML pages. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers.	CO 4
2	<b>What are the data types supported by JavaScript?</b> Undefined Null Boolean String Symbol Number Object	CO 5
3	<b>What are the features of JavaScript?</b> It is a lightweight, interpreted programming language. It is designed for creating network-centric applications. It is complementary to and integrated with Java. It is an open and cross-platform scripting language.	CO 3
4	<b>Is JavaScript a case-sensitive language?</b> Yes, JavaScript is a case sensitive language. The language keywords, variables, function names, and any other identifiers must always be typed with a consistent capitalization of letters.	CO 6
5	<b>What is the purpose of 'This' operator in JavaScript?</b> The JavaScript this keyword refers to the object it belongs to. This has different values depending on where it is used. In a method, this refers to the owner object and in a function, this refers to the global object.	CO 4
6	<b>What is Callback?</b> A callback is a plain JavaScript function passed to some method as an argument or option. It is a function that is to be executed after another function has finished executing, hence the name 'call back'. In JavaScript, functions are objects. Because of this, functions can take functions as arguments, and can be returned by other functions.	CO 5
7	<b>How to read a cookie using JavaScript?</b> Reading a cookie is just as simple as writing one, because the value of the document.cookie object is the cookie. So you can use this string whenever you want to access the cookie. The document.cookie string will keep a list of name = value pairs separated by semicolons, where name is the name of a cookie and value is its string value. You can use strings' split() function to break the string into key and values.	CO 3

8	<b>How to delete a cookie using JavaScript?</b>	CO 4
	If you want to delete a cookie so that subsequent attempts to read the cookie in JavaScript return nothing, you just need to set the expiration date to a time in the past. You should define the cookie path to ensure that you delete the right cookie. Some browsers will not let you delete a cookie if you don't specify the path.	
9	<b>What is the difference between Attributes and Property?</b>	CO 4
	Attributes- provide more details on an element like id, type, value etc. Property- is the value assigned to the property like type="text", value='Name' etc.	
10	<b>In how many ways a JavaScript code can be involved in an HTML file?</b>	CO 4
	Inline Internal External	
11	<b>What is a Typed language?</b>	CO 3
	Typed Language is in which the values are associated with values and not with variables. It is of two types: Dynamically: in this, the variable can hold multiple types; like in JS a variable can take number, chars. Statically: in this, the variable can hold only one type, like in Java a variable declared of string can take only set of characters and nothing else.	
12	<b>What is the difference between null &amp; undefined?</b>	CO 4
	Undefined means a variable has been declared but has not yet been assigned a value. On the other hand, null is an assignment value. It can be assigned to a variable as a representation of no value. Also, undefined and null are two distinct types: undefined is a type itself (undefined) while null is an object.	
13	<b>What is the difference between innerHTML &amp; innerText?</b>	CO 4
	innerHTML – It will process an HTML tag if found in a string innerText – It will not process an HTML tag if found in a string	
14	<b>Difference between “ == “ and “ === “ operators.</b>	CO 4
	Both are comparison operators. The difference between both the operators is that, “==” is used to compare values whereas, “===” is used to compare both value and types.	
15	<b>List some of the disadvantages of JavaScript.</b>	CO 3
	No support for multithreading No support for multiprocessing Reading and writing of files is not allowed No support for networking applications	
16	<b>Can an anonymous function be assigned to a variable?</b>	CO 4
	Yes, you can assign an anonymous function to a variable.	

17	<b>In JavaScript what is an argument object?</b>	CO 3
	The variables of JavaScript represent the arguments that are passed to a function.	
18	<b>What is BOM?</b>	CO 6
	BOM stands for Browser Object Model. It provides interaction with the browser. The default object of a browser is a window. So, you can call all the functions of the window by specifying the window or directly. The window object provides various properties like document, history, screen, navigator, location, innerHeight, innerWidth,	
19	<b>What is DOM? What is the use of document object?</b>	CO 3
	DOM stands for Document Object Model. A document object represents the HTML document. It can be used to access and change the content of HTML.	
20	<b>How to write a comment in JavaScript?</b>	CO 4
	Single Line Comment: It is represented by // (double forward slash) Multi-Line Comment: Slash represents it with asterisk symbol as /* write comment here */	
<b>MODULE IV</b>		
<b>UI BINDING LIBRARY FOR REACT</b>		
1	<b>What is React?</b>	CO 5
	React is a front-end JavaScript library developed by Facebook in 2011. 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	<b>What are the features of React?</b>	CO 5
	It uses the virtual DOM instead of the real DOM. It uses server-side rendering. It follows uni-directional data flow or data binding.	
3	<b>What are the limitations of React?</b>	CO 5
	React is just a library, not a full-blown framework Its library is very large and takes time to understand It can be little difficult for the novice programmers to understand Coding gets complex as it uses inline templating and JSX	
4	<b>Why can't browsers read JSX?</b>	CO 5
	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.	

5	<b>“In React, everything is a component.” Explain.</b>	CO 5
	Components are the building blocks of a React application’s UI. These components split up the entire UI into small independent and reusable pieces. Then it renders each of these components independent of each other without affecting the rest of the UI.	
6	<b>What is the purpose of render() in React.</b>	CO 5
	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 <code>&lt;form&gt;</code> , <code>&lt;group&gt;</code> , <code>&lt;div&gt;</code> etc. This function must be kept pure i.e., it must return the same result each time it is invoked.	
7	<b>What is Props?</b>	CO 5
	Props is the shorthand for Properties in React. They are read-only components which must be kept pure i.e. immutable. They are always passed down from the parent to the child components throughout the application. A child component can never send a prop back to the parent component. This help in maintaining the unidirectional data flow and are generally used to render the dynamically generated data.	
8	<b>What is a state in React and how is it used?</b>	CO 5
	States are the heart of React components. States are the source of data and must be kept as simple as possible. Basically, states are the objects which determine components rendering and behavior. They are mutable unlike the props and create dynamic and interactive components. They are accessed via <code>this.state()</code> .	
9	<b>What are synthetic events in React?</b>	CO 5
	Synthetic events are the objects which act as a cross-browser wrapper around the browser’s native event. They combine the behavior of different browsers into one API. This is done to make sure that the events show consistent properties across different browsers.	
10	<b>List some of the cases when you should use Refs.</b>	CO 5
	When you need to manage focus, select text or media playback To trigger imperative animations Integrate with third-party DOM libraries	
11	<b>What are Higher Order Components(HOC)?</b>	CO 6

	Higher Order Component is an advanced way of reusing the component logic. Basically, it's a pattern that is derived from React's compositional nature. HOC are custom components which wrap another component within it. They can accept any dynamically provided child component but they won't modify or copy any behavior from their input components. You can say that HOC are 'pure' components.	
--	---	--

12	<b>What are Pure Components?</b>	CO 5
	Pure components are the simplest and fastest components which can be written. They can replace any component which only has a render(). These components enhance the simplicity of the code and performance of the application.	
13	<b>What is the significance of keys in React?</b>	CO 5
	Keys are used for identifying unique Virtual DOM Elements with their corresponding data driving the UI. They help React to optimize the rendering by recycling all the existing elements in the DOM. These keys must be a unique number or string, using which React just reorders the elements instead of re-rendering them. This leads to increase in application's performance.	
14	<b>What is the significance of keys in React?</b>	CO 5
	Keys are used for identifying unique Virtual DOM Elements with their corresponding data driving the UI. They help React to optimize the rendering by recycling all the existing elements in the DOM. These keys must be a unique number or string, using which React just reorders the elements instead of re-rendering them. This leads to increase in application's performance.	
15	<b>What is Redux?</b>	CO 5
	Redux is one of the most trending libraries for front-end development in today's marketplace. It is a predictable state container for JavaScript applications and is used for the entire applications state management. Applications developed with Redux are easy to test and can run in different environments showing consistent behavior.	
16	<b>Explain the role of Reducer.</b>	CO 5
	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.	
17	<b>What is the significance of Store in Redux?</b>	CO 5
	A store is a JavaScript object which can hold the application's state and provide a few helper methods to access the state, dispatch actions and register listeners. The entire state/ object tree of an application is saved in a single store. As a result of this, Redux is very simple and predictable. We can pass middleware to the store to handle the processing of data as well as to keep a log of various actions that change the state of stores. All the actions return a new state via reducers.	

18	<b>What is React Router?</b>	CO 5
	React Router is a powerful routing library built on top of React, which helps in adding new screens and flows to the application. This keeps the URL in sync with data that's being displayed on the web page. It maintains a standardized structure and behavior and is used for developing single page web applications. React Router has a simple API.	
19	<b>Why is switch keyword used in React Router v4?</b>	CO 5
	Although a <code>div</code> 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 <code>switch</code> 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.	
20	<b>List down the components of Redux.</b>	CO 5
	Action – It's an object that describes what happened. Reducer – It is a place to determine how the state will change. Store – State/ Object tree of the entire application is saved in the Store. View – Simply displays the data provided by the Store.	
MODULE V		
CONNECT TO AN EXTERNAL API		
1	<b>What is Redux?</b>	CO 6
	Redux is an open-source library made using the scripting language JavaScript. Redux's primary use lies in managing and centralizing application state and it is usually used along with JavaScript libraries, for instance, React or Angular in order to build UIs (User Interfaces). It is a predictable state container for applications built using JavaScript. It is based on the Flux design pattern. Redux is very small in size (around 2 kilobytes) and has no dependencies	
2	<b>What is Redux in React js?</b>	CO 6
	Redux in React is the official React binding for Redux which allows the components in React to read data from a Redux Store, and dispatch Actions to the Store for updating the data. The purpose of Redux is to help applications scale well by providing means to manage the state via a unidirectional data flow model	
3	<b>Is it true that Redux can only be used with React?</b>	CO 6
	No, it is not true that Redux can only be used with React. Redux is being used as a data store for lots of UI layers. There are bindings available in Redux for React, Angular, Angular 2, Vue, etc.	

4	<b>What do you understand about Redux Toolkit?</b>	CO 5
	Redux Toolkit is Redux's official, opinionated, batteries included toolset for efficient Redux development. It also consists of the most widely used Redux add-ons, for instance, Redux Thunk for asynchronous logic, Reselect for writing selector functions and many more for making development easy for developers and saving them time.	
5	<b>Is it necessary to keep all the component states in the Redux store?</b>	CO 6
	No, it is not necessary to keep all the component states in the Redux store. We have to keep your application state as small as possible and therefore, we should do it only if it makes a difference for us to keep something there or maybe if it makes the use of Dev Tools easier.	
6	<b>What do you understand by an action in Redux's architecture?</b>	CO 6
	In the Redux architecture, actions are nothing but the plain JavaScript objects which contain a type field. They can be thought of as an event that is used to describe something which has happened in the application. Actions contain only a tiny bit of information that is required to mention what has happened.	
7	<b>Where can we use Redux?</b>	CO 6
	Redux is primarily being used along with React. However, we can also use it along with Angular, Vue, Meteor, etc. using the bindings it has to offer to us.	
8	<b>What are constants in Redux?</b>	CO 6
	Redux Saga functions as a separate thread in our programme which is solely responsible for side effects. Redux Saga is a redux middleware. In other words, it means that it can be started, paused, and aborted from the main application using standard Redux actions, has access to the entire Redux application state, and can also dispatch Redux actions	
9	<b>What is the mental model of redux saga?</b>	CO 6
	It is a documentation that aims to make elevation effects in redux applications easier and superior. It is obtainable in NPM as: npm install --save redux-saga	
10	<b>Name all the Redux Store methods.</b>	CO 6
	getState() subscribe(listener) dispatch(action) replaceReducer(nextReducer)	



11	<b>What do you understand about the Redux Saga?</b>	CO 6
	Redux Saga is a middleware library that can be useful for allowing a Redux store to interact with the resources outside of itself in an asynchronous manner, for example, making HTTP requests to external services, accessing browser storage, executing Input/Output operations and many more. These operations are also called side effects	
12	<b>Define Redux Thunk?.</b>	CO 6
	It is known as middleware that permits you to mark achievement creators that revisit a function in its place of an act. The Thunk can be utilized to hold up the post of an action. The internal function receives the layout methods to transmit and get state () as parameters	
13	<b>What do you mean by Virtual DOM?.</b>	CO 6
	It is an in-recollection illustration of Real DOM. The depiction of a UI is kept in memory and synced with the “real” DOM. It’s a pace that happens among the render function being called and the displaying of basics on the screen.	
14	<b>What do you mean by redux selectors and why to use them?</b>	CO 6
	Selectors are functions that obtain redux state as a dispute and revisit some data to overtake to the constituent. ,.	
15	<b>Why would you use force update in a React?</b>	CO 6
	In order to power a re-render if there is some form, React is not detecting that requires a revision to the UI.	

Course Coordinator:  
Mrs. M. Kalaiarasi M, Assistant Professor

HOD CSE