**Answers**

1: - Which is True about React js?

When writing unit tests for React, shallow rendering can be helpful. Shallow rendering lets you render a component “one level deep” and assert facts about what its render method returns, without worrying about the behavior of child components, which are not instantiated or rendered. This does not require a DOM.

2: - Which is True Regarding Component definition in React about class and function?

 Class components are ES6 classes and Functional Components are functions. The only constraint for a functional component is to accept props as an argument and return valid JSX

The key thing that makes this type of component different from a class component is the lack of state and lifecycle methods. This is why functional components are also commonly referred to as stateless components.

3: - Custom Html in React Js Component with innerHTml or DangeouslySetinnerhtml?

dangerouslySetInnerHTML is React’s replacement for using innerHTML in the browser DOM. In general, setting HTML from code is risky because it’s easy to inadvertently expose your users to a cross-site scripting (XSS) attack. So, you can set HTML directly from React, but you have to type out dangerouslySetInnerHTML and pass an object with a \_\_html key, to remind yourself that it’s dangerous. For example

function createMarkup() {

return {\_\_html: 'First &middot; Second'};

}

function MyComponent() {

return <div dangerouslySetInnerHTML={createMarkup()} />;

}

4: - Correct function to render ReactJS component at server side as static html page?

ReactDOMServer.renderToString(element)

Render a React element to its initial HTML. React will return an HTML string. You can use this method to generate HTML on the server and send the markup down on the initial request for faster page loads and to allow search engines to crawl your pages for SEO purposes.

ReactDOMServer.renderToStaticMarkup(element)

Similar to [renderToString](https://reactjs.org/docs/react-dom-server.html" \l "rendertostring), except this doesn’t create extra DOM attributes that React uses internally, such as data-reactroot. This is useful if you want to use React as a simple static page generator, as stripping away the extra attributes can save some bytes.

5: - Which function is invoked just before render?

ComponentWillMount().

6: - Prop Types can be used for?

We can use the propType for **validating** any data we are receiving from props

Also for **TypeChecking.**

7: - Flux Concept?

* Single directional data flow is easy to understand.
* The app is easier to maintain.
* The app parts are decoupled.
* **Actions** − Actions are sent to the dispatcher to trigger the data flow.
* **Dispatcher** − This is a central hub of the app. All the data is dispatched and sent to the stores.
* **Store** − Store is the place where the application state and logic are held. Every store is maintaining a particular state and it will update when needed.
* **View** − The **view** will receive data from the store and re-render the app.