MCQ on ReactJS  
WDPF Round 39----

1. React is a \_\_\_\_\_\_\_\_
2. Javascript library
3. Javascript framework
4. Both of above
5. None of above

Note: ReactJS is a javascript library for building user interfaces. Its not a framework

1. ReactJS covers \_\_\_\_\_\_\_\_\_
2. User Interface layer in an application
3. Data layer in an application
4. Both a and b
5. None of above

Note: ReactJS is a javascript library for building user interfaces. it covers only user interface layer of an application

1. ReactJS uses \_\_\_\_\_ to increase performance
2. Virtual DOM
3. Original DOM
4. Both of above
5. None of above

Note: ReactJS uses virtual DOM to increase performance

1. React is \_\_\_\_\_\_\_
2. Component based
3. Module based
4. Both of above
5. None of above
6. Consider following example

var Helloworld=(props)=>

{  
 return(  
 <div> hello world 1</div>  
 <div> Hello world 2</div>  
);  
}

ReactDOM.render(<Helloworld/>,mountNode)

What will be the output?

1. Hello world 1
2. Hello world 2
3. Hello world 1 Hello world 2
4. Error

Note: Two adjecent element in JSX should be wrapped in an element so this program will result an error

1. Number of elements, a valid react component can return
2. 1
3. 2
4. 3
5. 4

Note: in react component we can return only one element

1. State in react is\_\_\_\_\_\_\_\_
2. A permanent storage
3. Internal storage of the component
4. Both of above
5. None of above

Note: state in react is the storage of component

1. props in react can\_\_\_\_\_\_\_\_
2. Be changed inside the component
3. Not be changed in the component
4. Be changed in side other component
5. None of above

Note: props cannot be changed in the component

1. In ReactJS, props can be used to pass
2. Properties to the component
3. Event handler to component
4. Both of above
5. None of above

Note: in ReactJS props can be used to pass properties and event handlers to react component

1. The state in react can be updated by call to setState method. These calls are
2. Synchronous in nature
3. Asynchronous in nature
4. Are asynchronous but can be made synchronous when required
5. None of above

Note: Call to setState in reactjs is asynchronous in nature and multiple calls can be batched for better performance

1. As soon as the state of react component is changed, component will
2. Does nothing , you have to call render method to render the component again
3. re-renders the component
4. be created again from scratch
5. None of above

Note: As soon as state is updated, component re-renders itself

1. React Lifecycle method static getDerivedSateFromProps(props , state) is called when \_\_\_\_
2. Component is created for the first time
3. State of the component is updated
4. Both of above
5. None of above

Note: static getDerivedSateFromProps(props , state) is called in both cases , when component is created and when component is updated

1. componentDidMount lifecycle method is called when\_\_\_\_
2. Component is created for the first time
3. Component is updated
4. Both of above
5. None of above
6. Keys are given to a list of elements in react. These keys should be
7. Unique in the DOM
8. Unique among the siblings only
9. Do not requires to be unique
10. None of above

Note: Keys given to the list of elements in react should be unique among the siblings only.

1. For controlled components in react
2. Source of truth is DOM
3. Source of truth is component state
4. Source of truth can be anything
5. None of above

Note: For controlled components in react, source of truth is Component state

1. For uncontrolled components in react
2. Source of truth is DOM
3. Source of truth is component state
4. Source of truth can be anything
5. None of above.

Note: For un-controlled components in react, source of truth is DOM.

1. To upload a file from react component, A developer will require to write
2. A controlled component
3. An uncontrolled component
4. Can be done from both
5. None of above

Note: File upload cannot be done from controlled components because it requires user input

1. Ref in react component is used to assign reference to DOM elements. Ref can be created by
2. Calling React.createRef method
3. Assigning a value to ref attribute in JSX
4. Both of above
5. None of above

Note: reference can be created either by calling React.createRef or by assigning the value to ref attribute of an element

1. Everything in React is a \_\_\_\_\_\_\_\_\_\_\_\_\_
2. Module
3. Component
4. Package
5. Class
6. In which directory React Components are saved?
7. Inside js/components/
8. Inside vendor/components/
9. Inside external/components/
10. Inside vendor/
11. How many elements does a react component return?
12. 2 Elements
13. 1 Element
14. Multiple Elements
15. What is state in React?
16. A persistent storage.
17. An internal data store (object) of a component.
18. What is Babel?
19. A transpiler.
20. An interpreter
21. A Compiler
22. Both Compiler and Transpilar
23. What does the "webpack" command do?
24. Transpiles all the Javascript down into one file
25. Runs react local development server.
26. A module bundler
27. What port is the default where the webpack-dev-server will run?
28. 3000
29. 8080
30. 3306
31. What is ReactJS?
32. Server side Framework
33. User-interface framework
34. A Library for building interaction interfaces
35. What are the two ways that data gets handled in React?
36. state & props
37. services & components
38. In React what is used to pass data to a component from outside?
39. setState
40. render with arguments
41. props
42. PropTypes
43. We can go for keys when there is possibility that our user could change the data.
44. Keys
45. ref
46. both
47. none of above
48. JSX is typesafe.
49. True
50. False
51. React merges the object you provide into the current state using \_\_\_\_\_\_\_\_\_\_.
52. setState()
53. State()
54. Arbitrary inputs of components are called \_\_\_\_\_\_\_\_\_\_.
55. Keys
56. Props
57. Elements
58. Ref
59. \_\_\_\_\_\_\_\_\_ can be done while more than one element needs to be returned from a component.
60. Abstraction
61. Packing
62. Insulation
63. Wrapping
64. Which of the following needs to be updated to achieve dynamic UI updates?
65. State
66. Props
67. Lifecycle methods are mainly used \_\_\_\_\_\_\_\_\_\_\_.
68. To keep track of event history
69. to enhance components
70. freeup resources
71. none of the options
72. State can be initialized when code is loaded or state can be set on event changes.
73. False
74. True
75. ref is used to refer an element / component returned by \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
76. react()
77. render()
78. reduce()
79. refer()
80. In JSX most of the errors can be caught during \_\_\_\_\_\_\_\_\_.
81. Interpretation
82. Execution
83. Compilation
84. Build
85. Components cannot refer to other components in their output.
86. True
87. False
88. JSX is faster because it performs \_\_\_\_\_\_\_\_\_\_\_\_ while compiling code to JavaScript
89. Modification
90. Compression
91. Optimization
92. Encryption
93. If our elements are dynamic, react can keep track of the changes using keys.
94. True
95. False
96. Function that does not change its results for the same set of inputs are called \_\_\_\_\_\_\_\_\_\_.
97. Pure functions
98. Impure Functions
99. What is the smallest building block of ReactJS?
100. none of the options
101. props
102. elements
103. components
104. An altered component may be uniquely identified with the help of ref.
105. True
106. False
107. React considers everything as \_\_\_\_\_\_\_.
108. User interface
109. elements
110. components
111. Objects
112. React keeps track of what items have changed, been added, or been removed from a list using \_\_\_\_\_\_\_\_.
113. state
114. props
115. keys
116. ref
117. React is mainly for building \_\_\_\_\_\_\_\_\_\_\_\_\_.
118. Database
119. Connectivity
120. User interface
121. Design platform
122. React supports all the syntax of \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
123. ES6
124. Redux
125. None of options
126. Native Java
127. In React state can be accessed using \_\_\_\_\_\_\_\_.
128. current
129. state
130. current()
131. state()
132. How can we prevent default behavior in React?
133. None of the options
134. using revokeDefault()
135. Using preventDefault()
136. using avoidDefault()
137. Invoked once, only on the client, after rendering occurs.
138. componentWillUnmount
139. shouldComponentUpdate
140. componentWillMount
141. ComponentDidMount
142. What is used in ReactJs to increase performance?
143. Both B and c
144. Original DOM
145. Virtual DOM
146. None of the above
147. Which of the following below act as the input of a class-based component?
148. class
149. factory
150. render
151. Props
152. What is the declarative way to render a dynamic list of components based on values in an array
153. Using the reduce array method
154. Using the <Each /> component
155. Using the Array.map() method
156. With a for/while loop
157. Must API for every ReactJs component
158. SetinitialComponent
159. renderComponent
160. render
161. All of the above
162. Who develop react.js?
163. Apple
164. Facebook
165. Twitter
166. Google