FULL STACK DEVELOPER ASSESSMENT

# Part II

1. Rendering a component A which in turn rendering another component B. You want the component B to hide itself even after the rendering. What will you do to prevent the rendering of B?
   1. Using Composition
   2. Using context
   3. Using state
   4. Using callback

ANSWER:

B. **Using state**

1. In ReactJs what are the differences between Props and State?

ANSWER:

Props are the properties that can be passed to the component by it’s parent and managed by the parent. The State are values that are initialized and managed by the component itself.

1. Enumerate some lifecycle methods in a React component.

ANSWER:

componentDidMount

componentWillUnmount

componentDidUpdate

shouldComponentUpdate

1. What is virtual DOM?
   1. an in-memory representation of Real DOM.
   2. a programming interface for HTML and XML documents.
   3. a markup language for Web pages
   4. an XML or HTML document as a tree structure interface

ANSWER:

A. an in-memory representation of Real DOM.

1. What are the differences between Stateless Components and Stateful Components?

ANSWER:

Stateless components are components that accept properties from their parent and simply returns a React component that can be rendered in the DOM, Stateful components are kinda the same except they have state and they can manage their own state and update or modify their data.

1. What is the difference between Promise and callback?

ANSWER:

They both handle asynchronous tasks, but promise is much cleaner and easier to read.

1. What is React Router?

ANSWER:

It’s used for navigation in react

1. What is Typescript?

ANSWER:

It’s like javascript but with static types and other features, but it’s compiled into javascript before running.

1. Lifecycle methods are mainly used \_\_\_\_
   1. to keep track of event history
   2. to enhance components
   3. free up resources
   4. none of the above

ANSWER:

B.to enhance components

1. What happens when you call setState() inside render() method?
   1. Repetitive output appears on the screen
   2. Stack overflow error
   3. Duplicate key error
   4. Nothing happens. Life goes on!

ANSWER:

B.Stack overflow error

1. The MongoDB explain() method does not support which of the following verbosity mode:
   1. queryPlanner
   2. executionStats
   3. allPlansExecution
   4. customExecutionStats

ANSWER:

B.customExecutionStats

1. What does the following aggregate query perform?

|  |
| --- |
| db.posts.aggregate( [  { $match : { likes : { $gt : 100, $lte : 200 } } },  { $group: { \_id: null, count: { $sum: 1 } } } ] ); |

* 1. Calculates the number of posts with likes between 100 and 200
  2. Groups the posts by number of likes (101, 102, 103) by adding 1 every time
  3. Fetches the posts with likes between 100 and 200 and sets their \_id as null
  4. Fetches the posts with likes between 100 and 200, sets the \_id of the first document as null and then increments it 1 every time

ANSWER:

A.Calculates the number of posts with likes between 100 and 200

1. If you have created a compound index on (A, B, C) which of the following access pattern will not be able to utilize the index?
   1. A, B, C
   2. A, B
   3. B, C
   4. A

ANSWER:

c. B, C

1. Which type of indexes does MongoDB support?
   1. Compound Indexes
   2. Multikey Indexes
   3. Geospatial Indexes
   4. All of the above

ANSWER:

D.All of the above

1. Consider that the posts collection contains an array called ratings which contains ratings given to the post by various users in the following format:

|  |
| --- |
| {  \_id: 1,  post\_text: "This is my first post",  ratings: [5, 4, 2, 5],  //other elements of document  } |

Which of the following query will return all the documents where the ratings array contains elements that in some combination satisfy the query conditions?

* 1. db.inventory.find( { ratings: { $elemMatch: { $gt: 3, $lt: 6 } } } )
  2. db.inventory.find( { ratings: { $gt: 5, $lt: 9 } } )
  3. db.inventory.find( { ratings.$: { $gt: 5, $lt: 9 } } )
  4. db.inventory.find( { ratings: { $elemMatch: { $gte: 3, $lte: 6 } } } )

ANSWER:

D.db.inventory.find( { ratings: { $elemMatch: { $gte: 3, $lte: 6 } } } )

1. Which prop takes a function to be called every time there is a text changed for input type text in ReactJS
   1. onChangeText
   2. ChangeText
   3. onText
   4. None of the above

ANSWER:

D.None of the above

1. Which prop type in a React form component, will validate a value for an attribute is passed and of type function
   1. React.Prop.func
   2. React.PropTypes.func
   3. React.PropTypes.func.isRequired
   4. React.Prop.func.isRequired

ANSWER:

B.React.PropTypes.func

1. Rendering of JSX file, requires
   1. JQuery
   2. React-DOM
   3. React
   4. Browserify

ANSWER:

D.Browserify

1. Which attribute in React, facilitates efficient handling of a list of items
   1. key
   2. head
   3. value
   4. next

ANSWER:

A.key

1. What is a projection in MongoDB queries?
   1. It is the second argument in the find() method that may either specify a list of fields to return or list fields to exclude in the result documents.
   2. It is the query plan to optimize the query
   3. It is the storage format to store the data into the hard disk
   4. It is a condition which specifies the equality condition

ANSWER:

1. It is the second argument in the find() method that may either specify a list of fields to return or list fields to exclude in the result documents