

React Native

Pre / Post Test

Name: _____ Date: _____

1. What event should you wire up in a `<TextInput>` component?
 - a. `onChange`
 - b. `onKeyUp`
 - c. `onChangeText`
 - d. `onKeyDown`
 - e. I don't know
2. What is the best way to scaffold a React Native app?
 - a. `create-react-app --native`
 - b. `create-react-app --device`
 - c. `create-react-native-app`
 - d. `react-native --generate` (or `-g`)
 - e. `create-react-app --mobile`
 - f. I don't know
3. What is wrong with the following JSX?

```
return (  
  <View>  
    <View>  
      CEO of the Bluth Co.  
    </View>  
    <Image source={require(`../images/george.png`)} />  
  </View>  
)
```

 - a. Views should never be nested
 - b. `require` is a server-side import, not in a device
 - c. Views can only have a single child
 - d. Text can't be placed directly in a view
 - e. None of the above. That JSX is fine
 - f. I don't know
4. Which (if any) of these is not a React Native component?
 - a. `TouchableHighlight`
 - b. `TouchableWithFeedback`
 - c. `TouchableOpacity`
 - d. `TouchableWithoutFeedback`
 - e. `TouchableNativeFeedback`
 - f. All of the above are valid
 - g. I don't know
5. How many cross-platform fonts between iOS and Android can be used in React Native?
 - a. 0 - 10
 - b. 10 - 50
 - c. 50 - 100
 - d. 100 - 255
 - e. More than 255
 - f. None of the above
 - g. I don't know

6. Style inheritance is when you apply a style to a component and it also applies to inner components -- ones that are nested inside of it. Which of the following statement about style inheritance is most correct?
 - a. You must mark a style as inheritable for it to be passed to its inner components.
 - b. Inheritable styles (like list-style-type, border-collapse and color) are inherited automatically. Non-inheritable styles (like padding, margin, and text-size) are not.
 - c. All styles are inherited
 - d. Styles do not inherit in React Native
 - e. I don't know
7. What is the best technique to share view code between React components and React Native components?
 - a. It can't be done. React and React Native don't use the same tags.
 - b. Put your view files in a shared folder. Point the react.config.json file and the react-native.config.json file to that shared folder in the "viewSource" array.
 - c. Put your view files in a shared folder. Create a symbolic link to that folder in each project.
 - d. B and C both work
 - e. I don't know
8. How do you apply a CSS class to a React Native component?
 - a. `<Text class="big">Bluth Co.</Text>`
 - b. `<Text className="big">Bluth Co.</Text>`
 - c. `<Text cssClass="big">Bluth Co.</Text>`
 - d. You can't use CSS classes in React Native; styles must be applied to each component individually.
 - e. I don't know
9. How do you position things side-by-side in React Native?
 - a. `display: 'grid'`,
 - b. `float: 'left'`,
 - c. `flexDirection: 'row'`,
 - d. `display: 'inline'`,
 - e. `display: 'inline-block'`,
 - f. Put them in the same `<tr>` in a `<table>`
 - g. I don't know
10. Choose the best answer. How might you handle rendering things differently on different platforms?
 - a. You don't have to do anything special; React Native will determine the type of platform and render everything properly
 - b. Check if `Platform.OS==='ios'` is truthy and render different JSX
 - c. Make a call to a native API on either iOS or Android in a try/catch. If the catch is hit, you are not on the targeted platform. Render the other platform's JSX
 - d. You can't; React Native is cross-platform by design
 - e. I don't know