

W2 – Exercise 2 Questions

Green = correct answer

Q1. What's one of the main roles of the built-in <View> component?

- a. Directly output text.
- b. Structure/ Group other child components.
- c. Handle touch events.

Q2. Can you use HTML elements (e.g. <div>, <p>, <input>) in React Native apps?

- a. Yes, React Native is able to handle HTML components, but native components are preferred.
- b. No, React Native doesn't provide any built-in components, you need to build all components from scratch.
- c. No, React Native doesn't recognize these components – it doesn't know how to compile them to native views.

Q3. What's the relation between React Native component styling and CSS (Cascading Style Sheets) for the Web?

- a. React Native styling is inspired by CSS (comparable/ similar property names and values)
- b. React Native uses the exact same syntax but only supports a limited set of features.
- c. There is no connection.

Q4. Which of the following example style rules does NOT work in React Native?

- a. borderColor: 'black'
- b. padding: 10
- c. 'background-color': '#ccc'

Q5. Why would you use const styles = StyleSheet.create({}) instead of a regular JavaScript

object (const styles = {})?

- a. There is no reason to use StyleSheet.
- b. Using a StyleSheet adds validation and potential performance improvements.
- c. Using a StyleSheet unlocks extra styling properties.

Q6. What is "Flexbox"?

- a. A styling property that allows you to order items in a grid.
- b. A component built into React Native which you can use to structure content.
- c. A concept/ set of styling properties that allows you to structure content (i.e. create layouts).

Q7. What's the default styling/ layout behavior of a <View> component?

- a. It organizes child components in a grid.

- b. It doesn't allow any ordering or structuring of child components.
- c. It uses Flexbox to organize its child components.

Q8, If a <View> has flexDirection: 'column' (which is the default) - what does alignItems: 'flex-end' do in that case?

- a. It positions all child elements at the end of the column – on the horizontal axis.
- b. It positions all child elements at the end of the column – on the vertical axis.
- c. It has no effect