Villarba Justin R. IT3-3 MC

Instructions

1. What's your favorite component, and what's the coolest prop you've discovered for it?

My Favorite React Native Component and Its Coolest Prop

React Native has revolutionized mobile app development by offering a rich set of components and APIs that simplify building cross-platform applications. Among the many components available, one that stands out to me is the **`FlatList`** component. Not only is it a powerful tool for displaying lists of data efficiently, but it also comes with a range of features that enhance performance and user experience.

Why FlatList is My Favorite Component

One of the primary reasons **`FlatList`** stands out is its ability to handle large data sets efficiently. In mobile applications, performance is a crucial factor, especially when displaying long lists of items. Traditional `ScrollView` components load all the items in a list at once, which can lead to significant memory usage and slow performance, particularly when dealing with thousands of items. `FlatList`, on the other hand, implements **lazy loading**—it only renders items that are currently visible on the screen (along with a few extra items for smooth scrolling). This optimization leads to reduced memory consumption and faster rendering times, ensuring that the app runs smoothly even with extensive data.

In addition to its performance benefits, `FlatList` provides several built-in features, like **pull-to-refresh**, **infinite scrolling**, and the ability to handle complex layouts with custom item renders. This flexibility makes it an essential component in many apps, from social media feeds to e-commerce product listings.

```
import React, { useState } from "react";

import { View, Text, FlatList, SafeAreaView, TouchableOpacity } from
"react-native";



export default function MC_FlatList_Villarba() {

  const [items, setItems] = useState([

    { id: "1", title: "Item 1" },

    { id: "2", title: "Item 2" },
```

```
    { id: "3", title: "Item 3" },
    { id: "4", title: "Item 4" },
  ]);

  const [refreshing, setRefreshing] = useState(false);

  const handleRefresh = () => {
    setRefreshing(true);
    setTimeout(() => {
      setItems([...items, { id: Date.now().toString(), title: "New Item"
}]);
      setRefreshing(false);
    }, 1200);
  };

  const loadMore = () => {
    setItems([
      ...items,
      { id: Date.now().toString(), title: "More Item" },
    ]);
  };

  return (
    <SafeAreaView style={{ flex: 1, padding: 20 }}>
```

```
      <FlatList
        data={items}
        keyExtractor={(item) => item.id}
        refreshing={refreshing}
        onRefresh={handleRefresh}
        onEndReached={loadMore}
        onEndReachedThreshold={0.2}
        renderItem={({ item }) => (
          <TouchableOpacity
            style={{
              padding: 15,
              backgroundColor: "#e3e3e3",
              marginBottom: 10,
              borderRadius: 8,
            }}
          >
            <Text style={{ fontSize: 16 }}>{item.title}</Text>
          </TouchableOpacity>
        )}
      />
    </SafeAreaView>
  );
}
```

[redacted]