Q.1. What is the difference between composition and inheritance? Which paradigm is preferred in React? Write a brief note about it in your own words.

Inheritance and composition are two programming techniques developers use to establish relationships between classes and objects. Whereas inheritance derives one class from another, composition defines a class as the sum of its parts.

Classes and objects created through inheritance are tightly coupled because changing the parent or superclass in an inheritance relationship risk breaking your code. Classes and objects created through composition are loosely coupled, meaning that you can more easily change the component parts without breaking your code.

Because loosely coupled code offers more flexibility, many developers have learned that composition is a better technique than inheritance, but the truth is more complex. Choosing a programming tool is similar to choosing the correct kitchen tool: You wouldn't use a butter knife to cut vegetables, and in the same way you shouldn't choose composition for every programming scenario.

In computing, reactive programming is a declarative programming paradigm concerned with data streams and the propagation of change.