

Q1. What is Swift and what UI problem or paradigm does it address?

Answer:

Q2. Explain the component model in Swift and how components communicate with each other.

Answer:

Q3. How does Swift handle state management for complex applications?

Answer:

Q4. What is Swift's rendering strategy (CSR, SSR, SSG, ISR) and what are the tradeoffs?

Answer:

Q5. How does Swift handle routing and navigation in a single-page application?

Answer:

Q6. How do you fetch data and handle loading/error states in a Swift application?

Answer:

Q7. What performance optimization techniques are specific to Swift (memoization, virtualization, code splitting)?

Answer:

Q8. How do you test components in Swift (unit, integration, snapshot, e2e)?

Answer:

Q9. How do you handle forms and form validation in Swift?

Answer:

Q10. What accessibility (a11y) practices should you follow when building UI with Swift?

Answer: