Integration Testing Quiz

Based on the following article links:

- https://medium.com/%40keployio/integration-testing-a-comprehensive-guide-d2626cfbec71
- https://bk10895.medium.com/a-comprehensive-guide-to-integration-testing-in-react-concept-and-example-2d17443b53bf
- https://katalon.com/resources-center/blog/integration-testing?utm_source=chatgpt.com

Section 1: Quiz

1. What is the primary goal of integration testing?

2. True or False? Integration testing is done before unit testing.

3. Which of the following is NOT a common integration testing approach?
   A) Big Bang
   B) Incremental
   C) Systematic
   D) Sandwich

4. What's a key advantage of incremental over Big Bang integration?
   A) Less planning required
   B) Easier debugging
   C) Tests entire system at once
   D) No need for stubs/drivers

5. Describe the 'sandwich' integration approach.

6. True or False? Big Bang integration is ideal for complex systems with many interdependencies.

7. In React, what does integration testing typically focus on?

8. Which test sits between unit and end-to-end tests in React?
   A) Snapshot testing
   B) Integration testing

C) Static type testing
D) Accessibility testing

9. True or False? React integration tests replace the need for end-to-end tests.

10. Name two aspects you'd test in a React integration test.

11. Which library is best suited for React integration tests?
    A) Enzyme
    B) React Testing Library
    C) Selenium
    D) Cypress

12. Why prioritize integration tests in a React app?

13. In the testing pyramid, integration tests form the _____ layer between unit tests and E2E tests.

14. What percentage of tests are typically integration tests, per the testing pyramid?
    A) 5–10%
    B) 15–20%
    C) 40–50%
    D) 70–80%

15. True or False? Integration testing is usually slower and more expensive than E2E testing.

16. Which tool is NOT mentioned as top for integration testing?
    A) Postman
    B) SoapUI
    C) Selenium
    D) Mocha

17. Why might you use stubs or drivers in integration testing?

18. What is interface testing in the context of system integration testing?

19. Bottom-up integration testing starts with:
    A) High-level modules first
    B) Highest and lowest together
    C) All modules simultaneously
    D) Lowest-level modules first

20. True or False? Integration testing tools often include reporting, parallel execution, and CI/CD integration.

21. Give one challenge commonly faced during integration testing.

22. How does integration testing differ from unit testing in scope?

23. What role do mock services play in integration testing of APIs?

24. Why is integration testing critical in microservices architecture?

25. How can CI/CD pipelines support better integration testing practices?

Section 2: Answer Key

1. To verify that multiple modules/components work together as expected.

2. False.

3. C) Systematic.

4. B) Easier debugging.

5. A hybrid method combining top-down and bottom-up testing.

6. False.

7. Component interactions, state changes, and data flow.

8. B) Integration testing.

9. False.

10. Component interaction, API integration, UI update behavior.

11. B) React Testing Library.

12. They catch behavior bugs early, balance cost and coverage.

13. Middle.

14. B) 15–20%.

15. False.

16. D) Mocha.

17. To simulate components that aren't ready yet.

18. Testing communication protocols and data exchange between components.

19. D) Lowest-level modules first.

20. True.

21. Debugging interface issues due to complex dependencies.

22. Integration testing checks how units interact, not just individual correctness.

23. To simulate external systems and ensure consistent test conditions.

24. Ensures services communicate reliably and correctly with each other.

25. Automates tests, ensures quick feedback and regression detection.