

Tailwind CSS

1. What is Tailwind CSS?
2. Describe the utility-first approach of Tailwind CSS.
3. How does Tailwind CSS differ from traditional CSS frameworks like Bootstrap?
4. What are utility classes in Tailwind CSS?
5. How can you customize Tailwind CSS to fit the design requirements of a project?
6. What is the purpose of the Tailwind CSS configuration file?
7. How does Tailwind CSS support responsive design?
8. Give an example of how to apply Tailwind CSS classes to an HTML element to set margin and padding.
9. How can you create reusable components in a project that uses Tailwind CSS?
10. Explain how Tailwind CSS improves development speed and efficiency.

Bootstrap in React

1. What is Bootstrap, and why is it popular among developers?
2. How can you integrate Bootstrap with a React application?
3. What are the benefits of using Bootstrap's pre-designed components in a React project?
4. Describe the responsive grid system provided by Bootstrap.
5. How do you link to the Bootstrap CSS file via CDN in a React application?
6. What are the advantages of using third-party libraries that provide Bootstrap components as React components?
7. How can Bootstrap's grid system be utilized within React components?
8. Give an example of using a Bootstrap button component in a React application.
9. How does integrating Bootstrap with React help maintain consistency and familiarity for developers?
10. Compare and contrast the use of Tailwind CSS and Bootstrap in React applications.