ASP .NET QUESTIONS FOR INTERVIEW

1. What is .NET Framework? What are its main components?
2. Explain the difference between .NET Framework, .NET Core, and .NET 5.
3. What is C#? How does it relate to .NET?
4. What are the basic features of C#?
5. What is the Common Language Runtime (CLR)?
6. What is an assembly in .NET?
7. What are namespaces in C#? Why are they used?
8. Explain the difference between value types and reference types in C#.
9. What is object-oriented programming (OOP)? Explain the four pillars of OOP.
10. What is inheritance? How is it implemented in C#?
11. What is polymorphism? Explain with examples.
12. What is encapsulation? Why is it important in OOP?
13. What is abstraction? How is it achieved in C#?
14. What are interfaces in C#? How are they different from abstract classes?
15. What is a delegate in C#? How is it different from an interface?
16. What is a lambda expression in C#? How is it used?
17. What is LINQ (Language Integrated Query)?
18. What is exception handling in C#? How is it done using try-catch blocks?
19. Explain the concept of garbage collection in .NET. How does it work?
20. What are the different access modifiers in C#? Explain each one.

SQL QUESTIONS FOR INTERVIEW

1. What is SQL?
2. What is a Database Management System (DBMS)?
3. What are the different types of SQL commands?
4. Explain the difference between SQL, MySQL, and SQL Server.
5. What are the basic SQL data types?
6. What is a table in SQL?
7. How do you create a table in SQL?
8. What is a primary key?
9. How do you define a primary key in a table?
10. What is a foreign key?
11. How do you define a foreign key in a table?
12. What is a SELECT statement?
13. How do you retrieve all records from a table in SQL?
14. Explain the WHERE clause in SQL.
15. What is the difference between the WHERE and HAVING clauses?

16. How do you filter results using the WHERE clause?
17. What is the ORDER BY clause used for?
18. How do you sort query results in ascending and descending order?
19. What is the LIMIT clause used for?
20. What is the difference between the DISTINCT and GROUP BY clauses?
21. What is an aggregate function in SQL?
22. Explain the difference between COUNT, SUM, AVG, MIN, and MAX functions.
23. What is a JOIN in SQL?
24. Explain different types of joins in SQL (INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN).
25. What is a subquery?
26. How do you write a subquery in SQL?
27. What is a UNION operator?
28. How do you use the UNION operator in SQL?
29. What is normalization in databases?
30. Explain the concept of ACID properties in database transactions.
31. What is a lambda expression in C#? How is it used?
32. What is LINQ (Language Integrated Query)?
33. What is exception handling in C#? How is it done using try-catch blocks?
34. Explain the concept of garbage collection in .NET. How does it work?
35. What are the different access modifiers in C#? Explain each one.

Javacscript QUESTIONS FOR INTERVIEW

1. What is JavaScript?
2. What are the basic data types in JavaScript?
3. How do you declare variables in JavaScript?
4. What is the difference between `var`, `let`, and `const`?
5. How do you comment in JavaScript?
6. What is the difference between == and === operators?
7. What is a function in JavaScript? How do you define a function?
8. What is the difference between function declarations and function expressions?
9. What is scope in JavaScript?
10. What is hoisting in JavaScript?
11. Explain the concept of closures in JavaScript.
12. What are JavaScript callbacks? How are they used?
13. What are arrow functions in JavaScript? How do they differ from regular functions?
14. What is an array in JavaScript? How do you declare and manipulate arrays?
15. What is an object in JavaScript? How do you create and access object properties?
16. How do you iterate over arrays and objects in JavaScript?
17. What is the DOM (Document Object Model)?

18. How do you select elements in the DOM using JavaScript?
19. What are events in JavaScript? How do you handle events?
20. What is event delegation in JavaScript?
21. What is asynchronous programming in JavaScript?
22. What are promises in JavaScript? How do you use them?
23. What is `async/await` in JavaScript? How does it work?
24. What are closures in JavaScript? Can you give an example?
25. Explain the concept of prototypes in JavaScript.
26. What is the `this` keyword in JavaScript? How is it used?
27. What are JavaScript modules? How do you export and import modules?
28. What are some common methods of string manipulation in JavaScript?
29. How do you handle errors in JavaScript?
30. What tools and techniques do you use for debugging JavaScript code?

Angular QUESTIONS FOR INTERVIEW

1. What is Angular?
2. What are the key features of Angular?
3. What is the difference between AngularJS and Angular?
4. Explain the architecture of an Angular application.
5. What is TypeScript? Why is it used in Angular?
6. How do you create a new Angular project using Angular CLI?
7. What are components in Angular? How do you create a component?
8. Explain the role of templates in Angular.
9. What is data binding in Angular? Explain the different types of data binding.
10. What are directives in Angular? Explain the difference between structural and attribute directives.
11. What is interpolation in Angular?
12. What is property binding in Angular?
13. What is event binding in Angular?
14. What are services in Angular? How do you create a service?
15. Explain the concept of dependency injection in Angular.
16. What is a module in Angular? Why are modules used?
17. What is Angular CLI? How does it help in Angular development?
18. What is routing in Angular? How do you implement routing in an Angular application?
19. What is lazy loading in Angular routing?
20. What are Angular forms? How do you implement template-driven forms and reactive forms?
21. What are pipes in Angular? Explain with examples.

22. What is Angular Material? How do you use Angular Material components in an Angular application?
23. What is Angular HttpClient? How do you make HTTP requests in Angular?
24. What is Angular testing? How do you write unit tests and end-to-end tests in Angular?
25. What are Angular lifecycle hooks? Explain some commonly used lifecycle hooks.
26. Explain the concept of ViewChild and ContentChild in Angular.
27. What is Angular Ivy? How does it improve performance in Angular applications?
28. What is the difference between ngIf and ngShow/ngHide?
29. Explain the difference between ngOnInit and constructor in Angular components.
30. What are some best practices for Angular development?

React QUESTIONS FOR INTERVIEW

1. What are functional components in React?
2. How do you create a functional component in React?
3. What are the benefits of using functional components over class components in React?
4. Explain the useState() hook in React. How do you use it to manage state in functional components?
5. What is the useEffect() hook in React? How do you use it for side effects in functional components?
6. What are the rules of hooks in React? Why is it important to follow them?
7. Explain the useReducer() hook in React. How is it different from useState()?
8. What are custom hooks in React? How do you create and use them?
9. Explain the useCallback() hook in React. When and why would you use it?
10. What is the useContext() hook in React? How do you use it to consume context in functional components?
11. What is useMemo() hook in React? How do you use it to memoize values?
12. What is useRef() hook in React? How do you use it to create mutable values that persist across renders?
13. Explain the concept of functional programming in React.
14. What are pure components in React? How do they relate to functional components?
15. How do you handle component lifecycle events in functional components?
16. What are the differences between class components and functional components in React in terms of lifecycle methods?
17. Explain the concept of higher-order components (HOCs) in React. How do you implement them in functional components?
18. What are forward refs in React? How do you use them with functional components?
19. What is error boundary in React? How do you implement error boundaries with functional components?

20. How do you handle routing in React with functional components?
1.