Easy Level (20 Questions)

1. Create an object `person` with properties `name` and `age`.
2. Add a new property `gender` to the `person` object.
3. Update the `age` property of the `person` object.
4. Delete the `gender` property from the `person` object.
5. Check if the `name` property exists in the `person` object.
6. Create an empty object and add three properties to it dynamically.
7. Create an object with a method that logs "Hello, world!" to the console.
8. Access the value of a property using bracket notation.
9. Loop through all properties of an object using a `for...in` loop.
10. Create a copy of an object using `Object.assign()`.
11. Merge two objects using the spread operator.
12. Get all the keys of an object using `Object.keys()`.
13. Get all the values of an object using `Object.values()`.
14. Get both keys and values of an object using `Object.entries()`.
15. Check if an object is empty (has no properties).
16. Freeze an object using `Object.freeze()` and try modifying it.
17. Seal an object using `Object.seal()` and try deleting a property.
18. Write a function to create an object with `name` and `age` properties.
19. Convert an object to a JSON string.
20. Parse a JSON string back into an object.

---

1. Create a nested object to represent a family tree.
2. Write a function to find the number of properties in an object.
3. Create a function to calculate the sum of all numeric properties in an object.
4. Write a function to merge two objects, giving priority to the second object.
5. Create a deep copy of a nested object using recursion.
6. Create an object with a getter and a setter for a `fullName` property.
7. Write a function to remove all properties from an object.
8. Create an object to represent a car with methods to start and stop the engine.
9. Write a function to check if two objects are equal.
10. Sort an array of objects by a specific property.
11. Write a function to convert an array of key-value pairs into an object.
12. Write a function to flatten a nested object.
13. Create an object prototype for a `Person` with a method `introduce`.
14. Write a class `Student` with properties `name` and `grade` and a method `getDetails`.
15. Create a function to update nested properties of an object dynamically.
16. Create an object to represent a library with methods to add and remove books.
17. Use `Object.defineProperty()` to create a read-only property.
18. Create a factory function that returns objects with a specific structure.

19. Write a function to count the frequency of words in a string using an object.
20. Create a function to group an array of objects by a specific property.
21. Convert an object into a Map and back to an object.
22. Write a function to find the longest key in an object.
23. Create an object to represent a user profile with default values.
24. Write a function to find all properties with undefined values in an object.
25. Add methods to an object dynamically using `Object.prototype`.
26. Write a function to capitalize all string values in an object.
27. Write a function to find the intersection of keys in two objects.
28. Create a constructor function for a `Product` with properties `id` and `price`.
29. Write a function to filter an array of objects based on a condition.
30. Use destructuring to extract properties from an object and rename them.