JavaScript 50 Q&A: this, Constructor, Prototype, Inheritance

Q1 [Interview] What is 'this' in JavaScript?

Ans: 'this' refers to the context in which a function is called.

Q2 [Interview] What does 'this' refer to in global scope?

Ans: It refers to the window object (in browser).

Q3 [Interview] What does 'this' refer to in strict mode?

Ans: It is undefined in functions (non-methods).

Q4 [Interview] How does 'this' behave in arrow functions?

Ans: Arrow functions inherit 'this' from their parent.

Q5 [Interview] What is a constructor function?

Ans: A function used to create objects using 'new'.

Q6 [Interview] How to define a constructor?

Ans: Using capitalized function name.

```
function Person(name) { this.name = name; }
```

Q7 [Interview] What is a prototype?

Ans: An object shared by all instances created via constructor.

Q8 [Interview] How to add methods to prototype?

Ans: Use Constructor.prototype.methodName.

Q9 [Interview] What is __proto__?

Ans: It points to an object's prototype.

Q10 [Interview] Difference between __proto__ and prototype?

Ans: __proto__ is internal reference, prototype is property.

Q11 [Interview] What is inheritance in JS?

Ans: Objects inherit from other objects via prototype chain.

Q12 [Interview] How to inherit properties?

Ans: Use Object.create() or prototype chaining.

Q13 [Interview] What is the prototype chain?

Ans: A lookup chain for properties/methods.

Q14 [Interview] Can a function be both constructor and normal?

Ans: Yes, based on how it is invoked.

Q15 [Interview] What does new keyword do?

Ans: Creates new object, sets prototype, binds this.

Q16 [DSA] Create an object using constructor.

Ans:

```
function Car(model) { this.model = model; } let c = new Car('X');
```

Q17 [DSA] Add a method to prototype of constructor.

Ans:

```
Car.prototype.drive = function() { return 'driving'; };
```

Q18 [DSA] Check if property is from prototype.

Ans: Use hasOwnProperty and in operator.

Q19 [DSA] Implement inheritance using Object.create().

Ans: let B = Object.create(A);

Q20 [DSA] Simulate class with function and prototype.

Ans:

Q21 [DSA] Check if object instanceof a constructor.

Ans: obj instanceof Constructor

Q22 [DSA] Log all properties (own + inherited).

Ans: Use for...in loop.

Q23 [DSA] Create nested constructor inheritance.

Ans:

Q24 [DSA] Add static method to constructor function.

Ans:

Q25 [DSA] Override method in child object.

Ans:

Q26 [DSA] Use arrow function in prototype method.

Ans:

Q27 [DSA] Chain multiple prototype methods.

Ans:

Q28 [DSA] Access parent method from child object.

Ans:

Q29 [DSA] Create new object with shared method via prototype.

Ans:

Q30 [DSA] Compare prototypes of two objects.

Ans:

Q31 [LeetCode] typeof this in global scope?

Ans: 'object'

Q32 [LeetCode] this === window?

Ans: true in non-strict mode (browser)

Q33 [LeetCode] What does new Date() instanceof Date return?

Ans: true

Q34 [LeetCode] Is prototype a property of instances?

Ans: No, only of functions.

Q35 [LeetCode] Can you use __proto__ in modern JS?

Ans: Yes, but not recommended.

Q36 [LeetCode] Arrow functions and this?

Ans: Don't bind their own 'this'.

Q37 [LeetCode] Can you overwrite prototype?

Ans: Yes, with a new object.

Q38 [LeetCode] What is typeof prototype?

Ans: 'object'

Q39 [LeetCode] Does object inherit from Object.prototype?

Ans: Yes.

Q40 [LeetCode] Is prototype chain infinite?

Ans: No, ends at null.

Q41 [Conceptual] Does 'this' vary based on how a function is called?

Ans: Yes.

Q42 [Conceptual] Can a prototype method access instance vars?

Ans: Yes using this.varName.

Q43 [Conceptual] Can primitive types have prototype?

Ans: Yes, via boxing.

Q44 [Conceptual] What does Object.create(null) mean?

Ans: Creates object without prototype.

Q45 [Conceptual] Can constructor return object explicitly?

Ans: Yes, replaces 'this'.

Q46 [Conceptual] What is benefit of prototype methods?

Ans: Shared across instances (memory efficient).

Q47 [Conceptual] Can we change an object's prototype?

Ans: Yes using Object.setPrototypeOf().

Q48 [Conceptual] Do arrays use prototype?

Ans: Yes.

Q49 [Conceptual] Does function have prototype by default?

Ans: Yes.

Q50 [Conceptual] Is Object.getPrototypeOf(x) same as x.__proto__?

Ans: Yes.