Quiz

What is the purpose of the `new` keyword?

Low Easy JavaScript OOP

Edit on GitHub

TL;DR

The `new` keyword in JavaScript is used to create an instance of a user-defined object type or one of the built-in object types that has a constructor function. When you use `new`, it does four things: it creates a new object, sets the prototype, binds `this` to the new object, and returns the new object.

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

const person1 = new Person('Alice');
console.log(person1.name); // Alice
```

▷ Run

The purpose of the `new` keyword

Creating a new object

The `new` keyword is used to create a new instance of an object. When you call a constructor function with `new`, it creates a new object.

```
function Car(model) {
  this.model = model;
}

const myCar = new Car('Toyota');
console.log(myCar.model); // Toyota
```

Run

All practice questions 0/291

Mark complete

The new object's internal `[[Prototype]]` property is set to the constructor function's `prototype` property. This allows the new object to inherit properties and methods from the constructor's prototype.

```
function Animal(type) {
  this.type = type;
}

Animal.prototype.speak = function () {
  console.log(`${this.type} makes a sound`);
};

const dog = new Animal('Dog');
dog.speak(); // Dog makes a sound
```

▷ Run

Binding `this` to the new object

Inside the constructor function, `this` refers to the new object that is being created. This allows you to add properties and methods to the new object.

```
function Book(title) {
  this.title = title;
}

const myBook = new Book('JavaScript Essentials');
console.log(myBook.title); // JavaScript Essentials
```

▷ Run

Returning the new object

The `new` keyword implicitly returns the new object created by the constructor function. If the constructor function explicitly returns an object, that object will be returned instead.

```
function Gadget(name) {
  this.name = name;
  return { type: 'Electronic' };
}

const myGadget = new Gadget('Smartphone');
console.log(myGadget.type); // Electronic
console.log(myGadget.name); // undefined
```

Run

Further reading

- MDN Web Docs: new operator
- JavaScript.info: Constructor, operator "new"
- Eloquent JavaScript: Constructors

Report an issue

Edit on GitHub

SWAG overflow
value: any;

greatfrontend

Subscribe to our newsletter

Enter your email Notify me

Practice

Get started

JavaScript functions

User interface coding

System design

Quiz

Guides

Front End Interview Playbook

Front End System Design Playbook

React Interview Playbook

Behavioral Interview Playbook

Study Plans

1 Week Plan

1 Month Plan

3 Months Plan

Company

Pricing

Roadmap

About

Team

Contact us

Advertise with us New

Become an affiliate

Careers

Blog

Medium

DEV Community

English (US)