Objects and Arrays

Mark Zamoyta

SOFTWARE DEVELOPER AND EDUCATOR

@markzamoyta

Introduction

Constructor Functions

Prototypes

Expanding Objects Using Prototypes

JavaScript Object Notation – JSON

Array Iteration

Constructor Functions

Constructor Function

```
function Car() {


}


let car = new Car();
```

this Keyword in Constructor Function

```javascript
function Car(id) {

    this.carId = id;
}


let car = new Car(123);

console.log( car.carId );   // 123
```

Methods

```
function Car(id) {

    this.carId = id;

    this.start = function() {

        console.log('start: ' + this.carId);

    };
}

let car = new Car(123);

car.start();   // start: 123
```

Prototypes

Without Prototypes

```javascript
function Car(id) {

    this.carId = id;

    this.start = function() {

        console.log('start: ' + this.carId);

    };
}


let car = new Car(123);

car.start();   // start: 123
```

Prototypes

```javascript
function Car(id) {
    this.carId = id;
}


Car.prototype.start = function() {

    console.log('start: ' + this.carId);

};


let car = new Car(123);

car.start();  // start: 123
```

Expanding Objects Using Prototypes

String.prototype

```javascript
String.prototype.hello = function() {

    return this.toString() + ' Hello';

};


console.log('foo'.hello());   // foo Hello
```

JSON – JavaScript Object Notation

Convert to JSON

```javascript
let car = {

    id: 123,

    style: 'convertible'

};

console.log( JSON.stringify(car) );
// {"id":123,"style":"convertible"}
```

Convert Array to JSON

```javascript
let carIds = [
    { carId: 123 },
    { carId: 456 },
    { carId: 789 }
];


console.log( JSON.stringify(carIds) );
// [{"carId":123},{"carId":456},{"carId":789}]
```

Parsing JSON

```
let jsonIn =
`

 [
   {"carId" : 123},
   {"carId" : 456},
   {"carId" : 789}
 ]
`;

let carIds = JSON.parse(jsonIn);
console.log( carIds );
```

Array Iteration

Array Iteration

```javascript
let carIds = [
  { carId: 123, style: 'sedan' },
  { carId: 456, style: 'convertible' },
  { carId: 789, style: 'sedan' }
];

carIds.forEach( car => console.log( car ));

carIds.forEach((car,index) => console.log( car, index ));
```

Array Filtering

```javascript
let carIds = [
  { carId: 123, style: 'sedan' },
  { carId: 456, style: 'convertible' },
  { carId: 789, style: 'sedan' }
];
let convertibles = carIds.filter(
    car => car.style === 'convertible'
);

console.log(convertibles);
```

Array Testing

```javascript
let carIds = [
  { carId: 123, style: 'sedan' },
  { carId: 456, style: 'convertible' },
  { carId: 789, style: 'sedan' }
];
let result = carIds.every(
    car => car.carId > 0
);

console.log(result);
```

Locate the First Match

```
let carIds = [
  { carId: 123, style: 'sedan' },
  { carId: 456, style: 'convertible' },
  { carId: 789, style: 'sedan' }
];
let car = carIds.find(
    car => car.carId > 500
);

console.log(car);
```

Summary

Constructor Functions
– new Car()
– creates new **this** value

Prototypes

Expanding Objects Using Prototypes

JavaScript Object Notation – JSON

Array Iteration
– forEach, filter, every, find