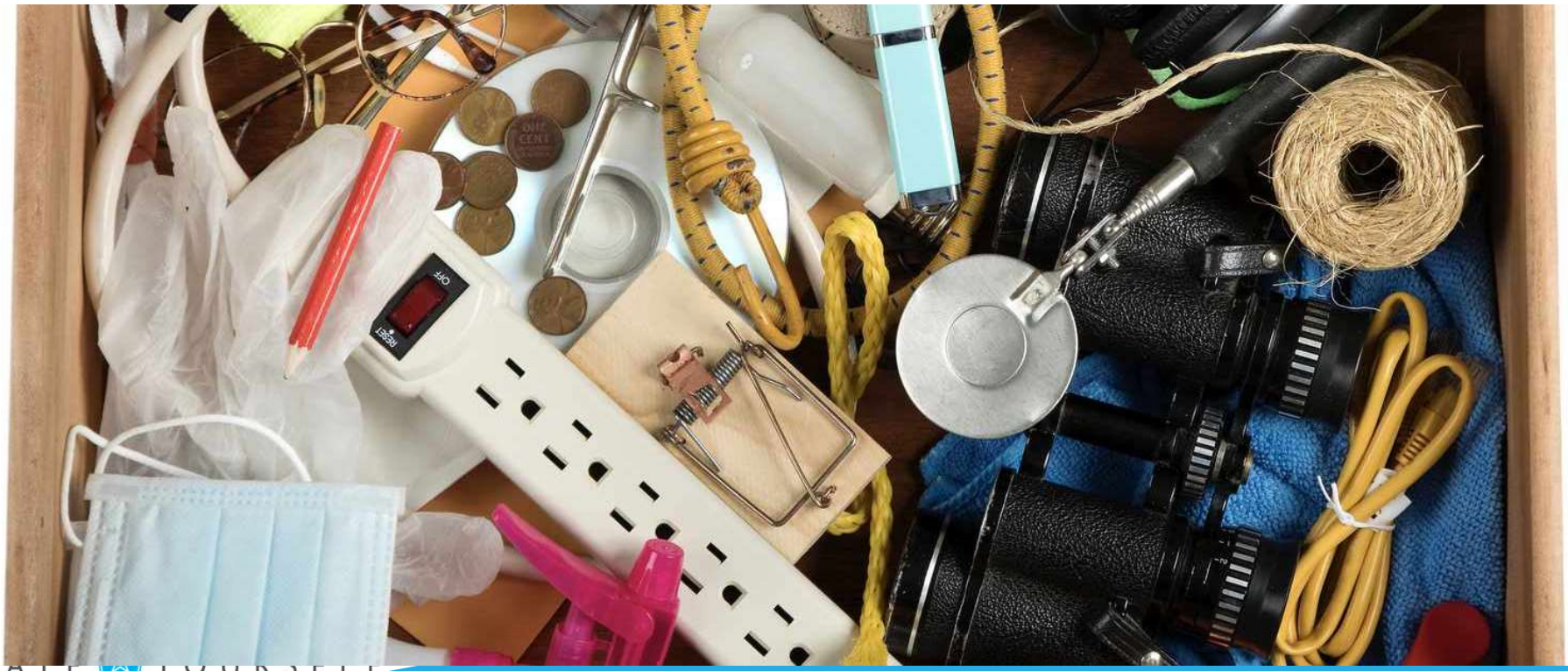
MODULE 3: FRONT END

JavaScript Essentials Part 2

TECH ELEVATOR
A Stride Company

JavaScript Language Features



Object Literals

```
const obj = {
   firstName: “Henry",
   lastName: “Edwards",
   age: 40
 };
```

- Simple Key:Value pairs

Object Prototype

- First, we write a method to use as a constructor:

```
const obj = {
   firstName: “Henry",
   lastName: “Edwards",
   age: 40
 };
```

```
function Person(firstName,lastName,age)  {
   this.firstName = firstName;
   this.lastName = lastName;
   this.age = age;
 };
```

Object Prototype

- JavaScript added the ***new*** keyword to create a special type of object.

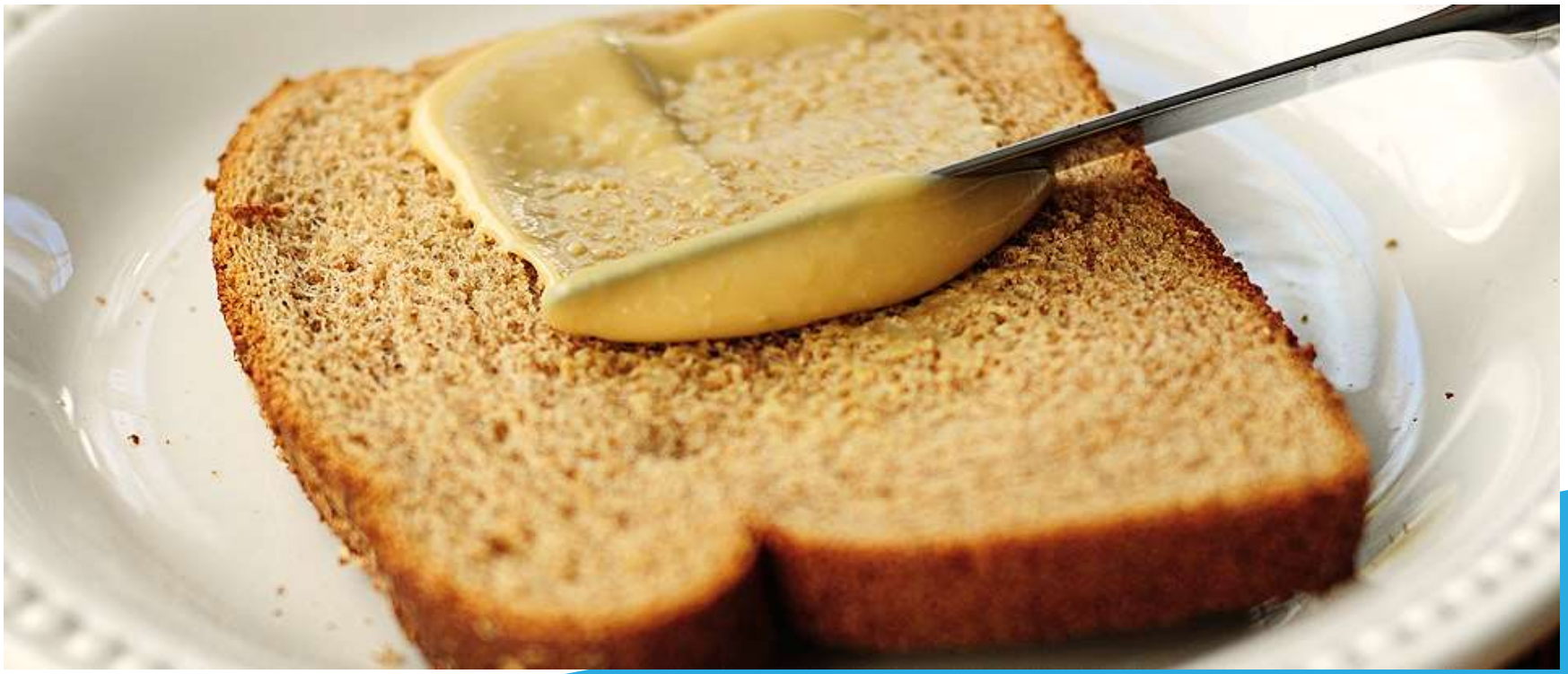
```
function Person(firstName,lastName,age)  {
   this.firstName = firstName;
   this.lastName = lastName;
   this.age = age;
 };

const henry = new Person('Henry','Edwards',32);
const mimi = new Person('Mimi','Malone',32);
```

Object Prototype

- Now, let's add methods to our object:

```
function Person(firstName,lastName,age)  {
   this.firstName = firstName;
   this.lastName = lastName;
   this.age = age;
 };

Person.prototype.fullName = function() {
      return `${this.firstName} ${this.lastName}`;
}
const henry = new Person('Henry','Edwards',32);
const mimi = new Person('Mimi','Malone',32);

console.log(henry.fullName());
```

Spread Syntax



Spread Syntax – Arrays

... is the syntax to spread out an array or object

```
const letters = ['a', 'b', 'c'];
const numbers = [1, 2, 3];

const combined = [...numbers, ...letters];
console.log(combined); // [1, 2, 3, 'a', 'b', 'c']


const numbers = [99, 23, 37];

const newNumbers = [0, ...numbers, 4, 5];
console.log(newNumbers); // [0, 99, 23, 37, 4, 5]
```

Spread Syntax – Objects

- Same as with arrays, except we are working with object -- {} not []

```
// A person object literal
const person = {
  firstName: 'Alex',
  lastName: 'Rodriguez',
};

// An address object literal
const address = {
  streetAddress: '312 Oak St',
  city: 'Paterson',
  state: 'NJ',
  zip: '07501',
};

// Creating a combined object literal with properties from both
const mailingAddress = { ...person, ...address };
```

```
// This is mailingAddress
{
  firstName: 'Alex',
  lastName: 'Rodriguez',
  streetAddress: '312 Oak St',
  city: 'Paterson',
  state: 'NJ',
  zip: '07501',
}
```

Destructuring

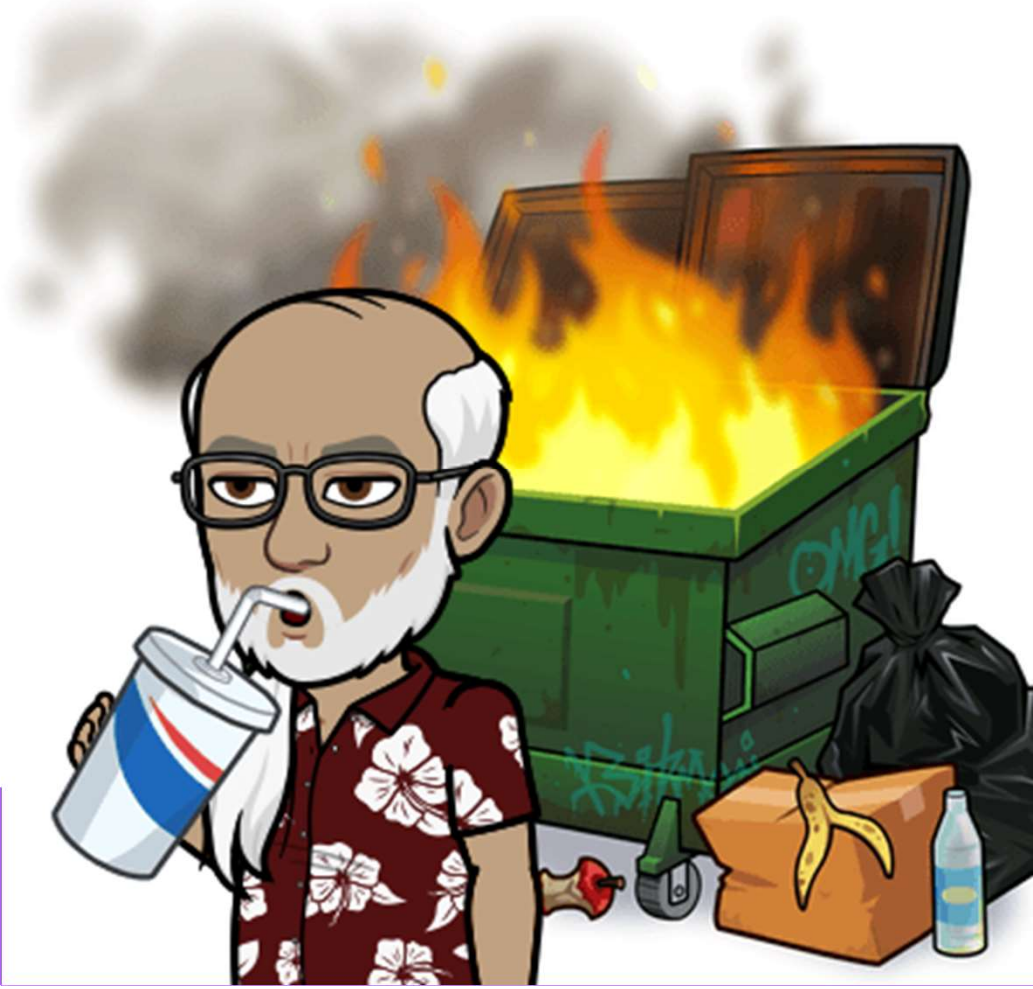
Destructuring Syntax – Arrays

```
const numbers = [1, 2, 3, 4, 5];

const [x, y] = numbers;
console.log(x); // prints 1
console.log(y); // prints 2
console.log(numbers); // prints [1, 2, 3, 4, 5]
```

Destructuring Syntax – Objects

```
const person = {
  firstName: 'Alex',
  lastName: 'Rodriguez',
  phone: '987-654-3210',
};

const { firstName, phone } = person;
console.log(firstName); // prints Alex
console.log(phone); // prints 987-654-3210
```

How to keep it straight?



ELEVATE YOURSELF

JavaScript Modules

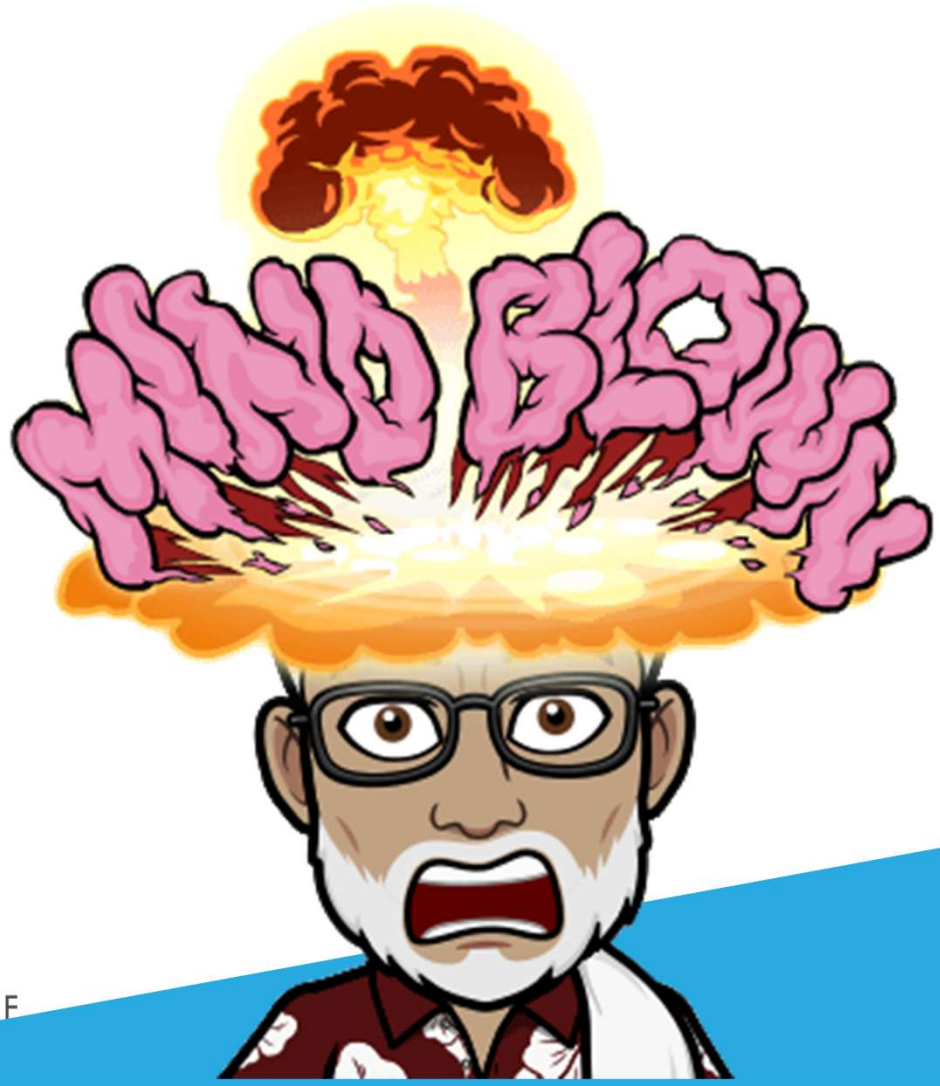
ECMAScript 2015 introduced modules (ES6)

Uses the **export** keyword to share functions, constants, or object types

A Deck of Cards Module

```
// Card object constructor function
function Card(suit, rank) {
        this.suit = suit;
        this.rank = rank;
        Object.freeze(this);
}
// Deck object constructor function
function Deck() {
        this.cards = [];
        for (const suit of suits) {
                for (const rank of ranks) {
                        this.cards.push(new Card(suit, rank));
                }
        }
        Object.freeze(this);
}
// Exporting Card and Deck
export { Card, Deck };
```

```
Import {Card,Deck} from “./deck.js”;

const kingOfHearts = new Card(“Hearts”,”King”);
```



ELEVATE YOURSELF

LET'S CODE!

WHAT QUESTIONS DO YOU HAVE?

Reading for tonight:

JavaScript Essentials Part 2