JavaScript

JavaScript

vs

TypeScript

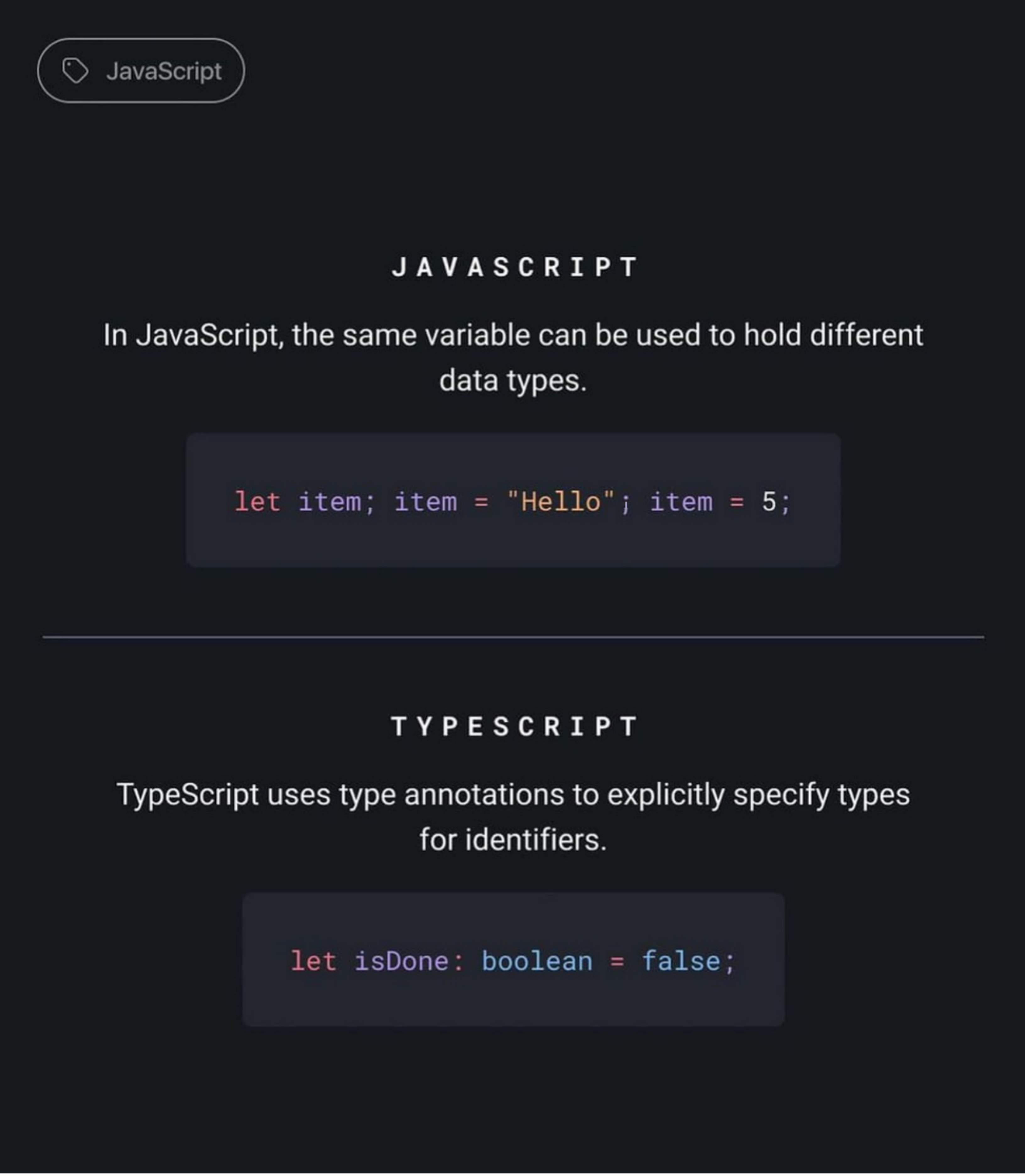
JavaScript

JAVASCRIPT

JavaScript is a high-level, interpreted programming language used to make web pages more interactive.

TYPESCRIPT

TypeScript is a typed superset of JavaScript that compiles to plain JavaScript. It offers static type checking.



JavaScript

JAVASCRIPT

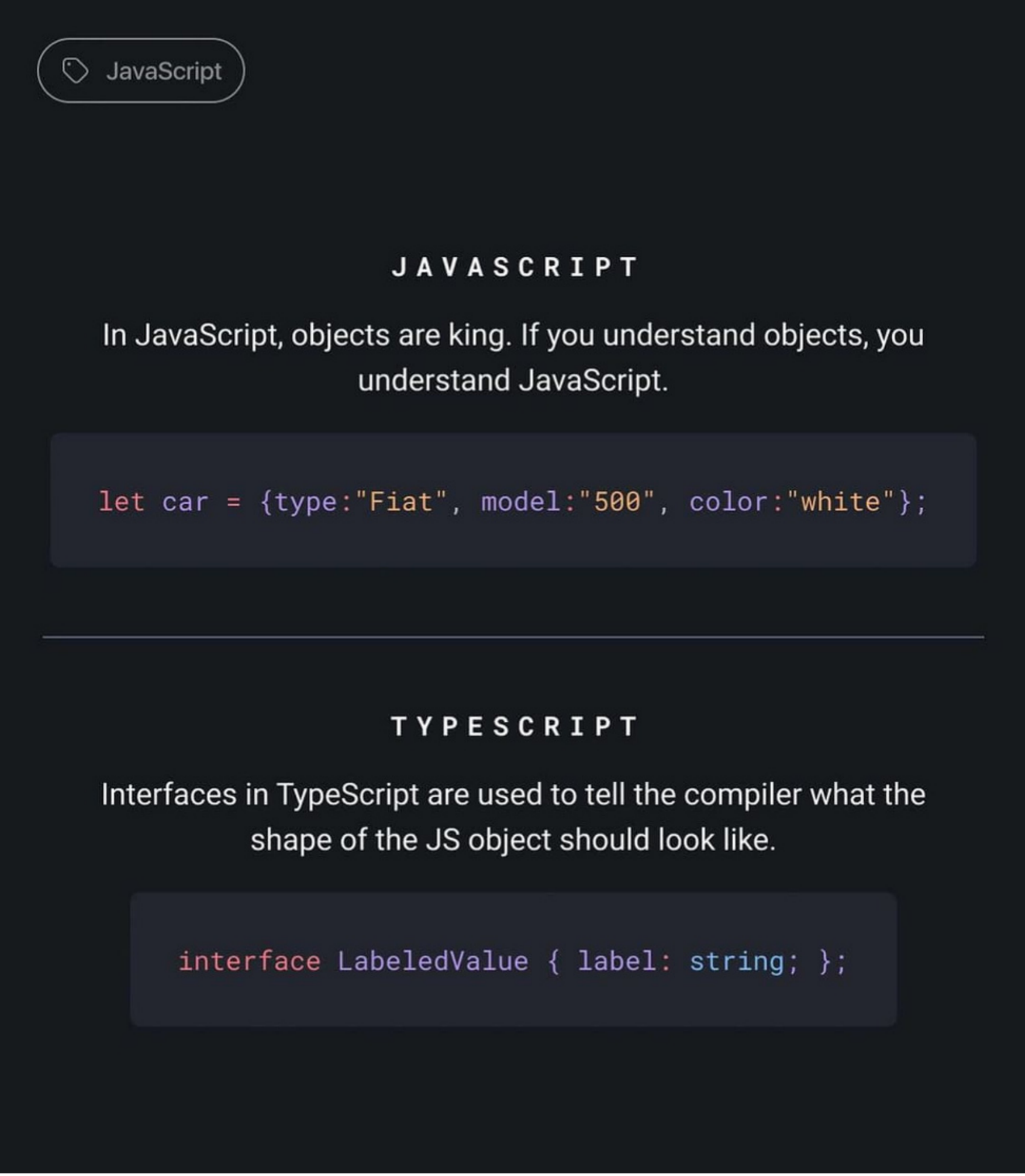
In JavaScript, the same variable can be used to hold different data types.

```javascript
let item; item = "Hello"; item = 5;
```

TYPESCRIPT

TypeScript uses type annotations to explicitly specify types for identifiers.

```typescript
let isDone: boolean = false;
```



JAVASCRIPT

In JavaScript, objects are king. If you understand objects, you understand JavaScript.

```
let car = {type:"Fiat", model:"500", color:"white"};
```

TYPESCRIPT

Interfaces in TypeScript are used to tell the compiler what the shape of the JS object should look like.

```
interface LabeledValue { label: string; };
```

JavaScript

JAVASCRIPT

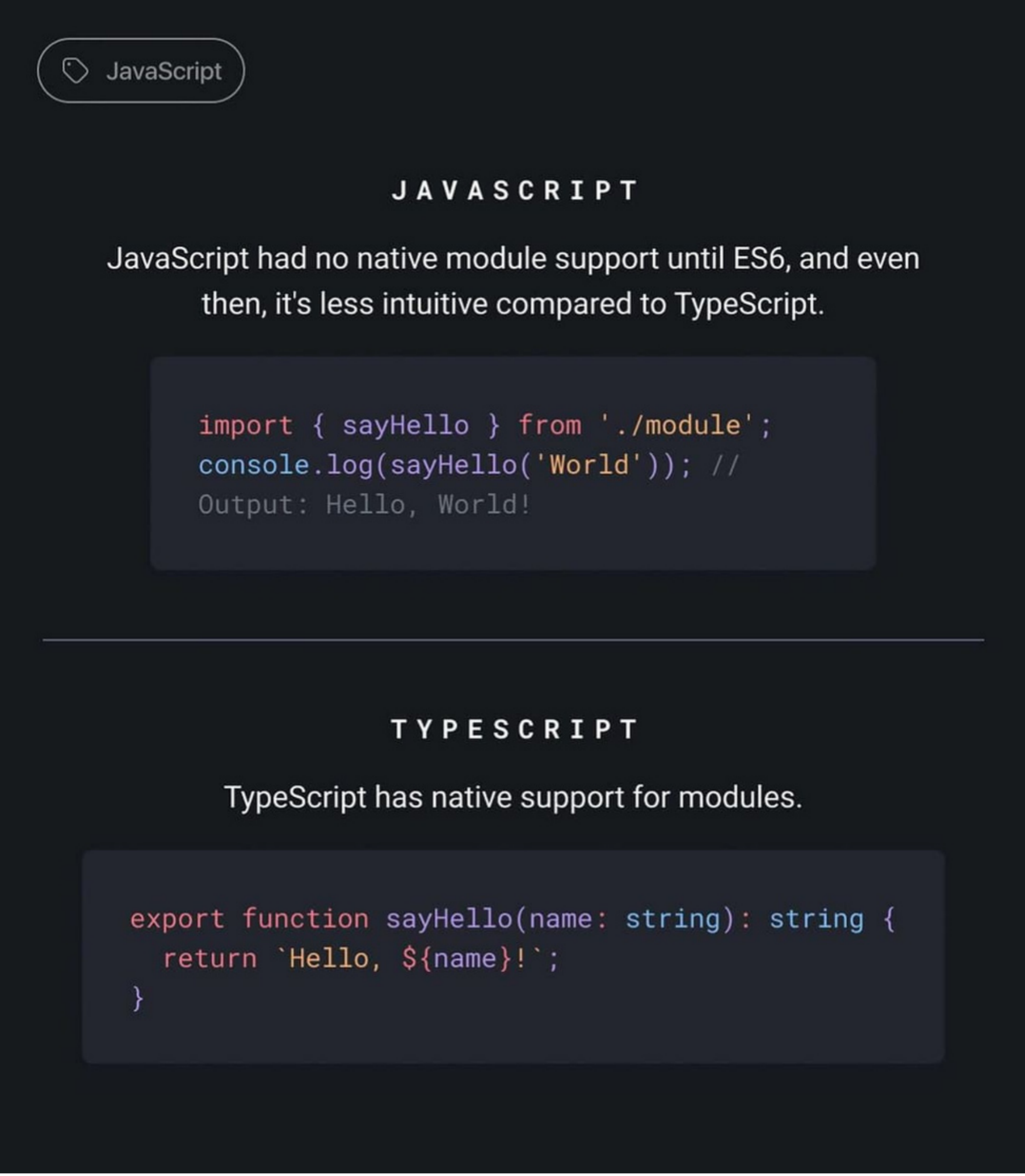
JavaScript classes are "syntactical sugar" over JavaScript's existing prototype-based inheritance.

```
function Employee(name, job) {
this.name = name; this.job = job; }
```

TYPESCRIPT

TypeScript has full support for classes including inheritance, generics, and implements.

```
class Greeter { greeting: string; }
```



JavaScript

JAVASCRIPT

JavaScript had no native module support until ES6, and even then, it's less intuitive compared to TypeScript.

```
import { sayHello } from './module';
console.log(sayHello('World')); //
Output: Hello, World!
```

TYPESCRIPT

TypeScript has native support for modules.

```
export function sayHello(name: string): string {
  return `Hello, ${name}!`;
}
```