Save for later

TypeScript Guide for JavaScript Programmers

@nacercodes

001

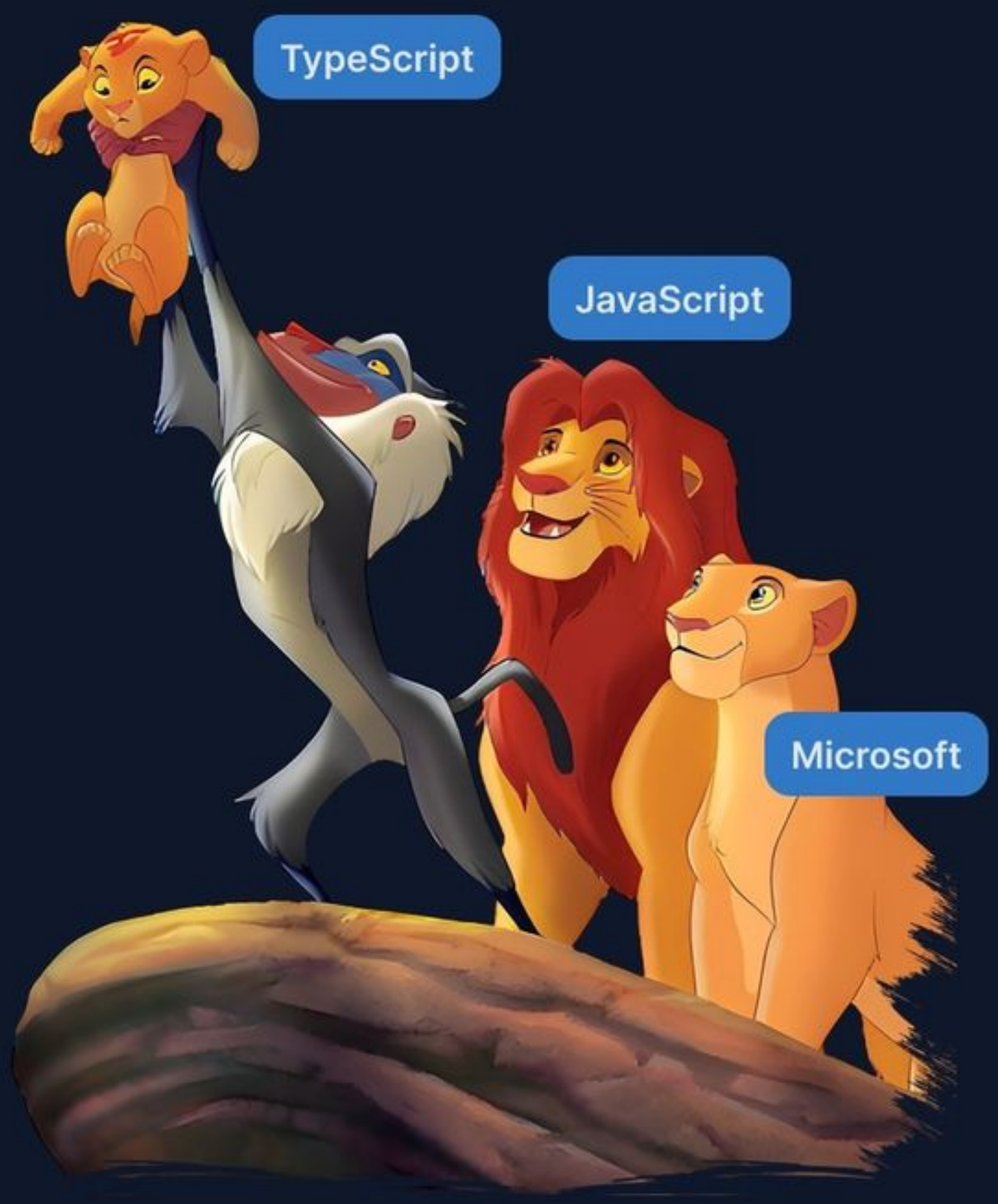
TypeScript = JavaScript + Syntax for Types

This simple equation makes it clear what TypeScript is and its main purpose. To follow along with this post, you need to know the basics of JavaScript and the concept of typing.

002

What is TypeScript?

TypeScript is a variant of JavaScript developed by Microsoft that checks a program for errors before execution.



003

Types by Inference

TypeScript is smart enough to generate types for you in many cases.

```
let calories = 89
    let calories: number

let fruits = ['Banana', 'Orange']
    let fruits: string[]
```

If you want to specify the type during declaration:

```
let calories: number = 89
```

004

Defining Types

You can describe your object shape using an interface declaration.

```ts
interface Fruit {
  id: number
  name: string
}
```

This is what will happen if you provide an object that doesn't match the interface you have provided:

```ts
let fruit: Fruit = {
  description: 'What am I doing here?'
}
```

```
Type '{ description: string; }' is not assignable
to type 'Fruit'.
```

005

Unions

With a union, you can declare that a type could be one of many simple types.

```ts
type FruitSize = 'S' | 'M' | 'L'
```

```ts
let validSize: FruitSize = 'M'

let invalidSize: FruitSize = 'Medium'
```

```
Type '"Medium"' is not assignable to type
'FruitSize'.
```

006

Generics

With generics, you can create type variables to write reusable code.

```
interface Page<T> {
  data: T[]
}
```

```
let fruitsPage: Page<Fruit> = {
  data: [fruit1, fruit2]
}

let vegetablesPage: Page<Vegetable> = {
  data: [vegetable1, vegetable2]
}
```

007

Structural Type System

If two objects have the same shape, they are considered to be of the same type.

```
function eatFruit(fruit: Fruit) {
  console.log(
    `Yummy! ${fruit.name} is delicious`
  )
}

// Passing an object with the same shape
eatFruit({ id: 1, name: 'Banana' })
```

Console

```
Yummy! Banana is delicious
```

Nacer Codes

@nacercodes

Save it or lose it. 🙂