

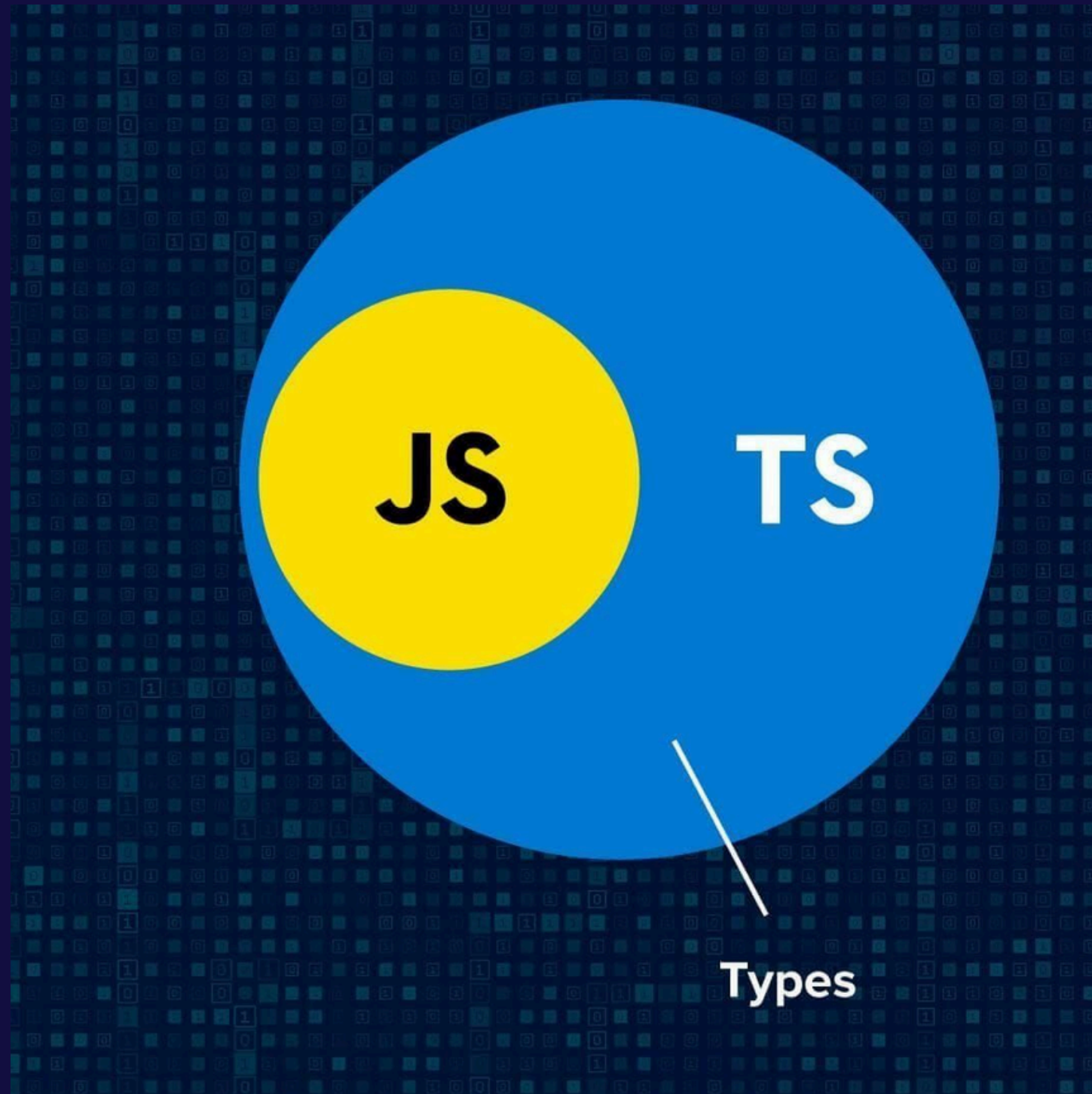
TypeScript

TypeScript Nedir?

The TypeScript logo, consisting of the letters 'TS' in a bold, white, sans-serif font, centered within a solid blue square.

Typescript, Microsoft tarafından geliştirilen açık kaynak bir programlama dilidir. JavaScript diline katı kurallar eklemeyi sağlar.

TypeScript - JavaScript



JavaScript'in tüm özelliklerini içinde barındıran kapsayıcı bir küme gibidir. Ek olarak **tip belirlemesini zorunlu** kılar.

TypeScript'in Avantajları

Statik Tipler

- TypeScript, statik veri tiplemesine sahiptir. JavaScript'te ise verilerin tipi dinamik olarak yürütme aşamasında belirlenir.


Nesne Yönelimli

- Nesne yönelimli programlamayı destekler. Sınıflar, metotlar, modüller vb. kullanarak çok daha iyi bir kod mimarisi sunar.


Derleme

- JavaScript'te derleme aşaması olmadığından kod çalışana kadar hata tespiti yapılamaz. TypeScript ise derleme aşamasında hataları tespit eder.

Güncel Versiyonu


typescript 

5.5.4 • Public • Published 3 days ago

 Readme




 Code Beta

 0 Dependencies

 54.998 Dependents

 3.140 Versions

TypeScript

 CI passing  npm package 5.5.4  downloads 238M/month  openssf scorecard 8.6

TypeScript is a language for application-scale JavaScript. TypeScript adds optional types to JavaScript that support tools for large-scale JavaScript applications for any browser, for any host, on any OS. TypeScript compiles to readable, standards-based JavaScript. Try it out at the [playground](#), and stay up to date via [our blog](#) and [Twitter account](#).

Find others who are using TypeScript at [our community page](#).

Installing

For the latest stable version:

```
npm install -D typescript
```


Install

```
> npm i typescript
```

Repository

 github.com/Microsoft/TypeScript

Homepage

 www.typescriptlang.org/

Weekly Downloads

55.901.198

Version

5.5.4

License

Apache-2.0

Type'lar

Boolean

true ve false değerler için kullanılır.

```
let isLoading: boolean = true;
```

Type'lar

Number

Sayısal veri türleri için kullanılır.

```
let num: number = 5;
```

Type'lar

String

Metinsel veri türleri için kullanılır.

```
let job: string = "Doctor";
```


Type'lar

Array

Dizi tanımlamaları için kullanılır.

```
let list: number[] = [1,2,3];
```

Type'lar

Tuple

Bir array içindeki veri tipleri birbirinden farklı ise kullanılır.

```
let user: [string, number];  
user = ["Can", 25]; // Çalışır.  
user = [25, "Can"]; // Çalışmaz.
```

Type'lar

Enum

Obje içerisindeki key'lere veya key değerlerine daha kolay erişmek için kullanılır.

```
enum Color {Red, Green, Blue}  
let c: Color = Color.Green
```

Type'lar

Any

Bilinmeyen değişken tipleri için kullanılır.
Örneğin, dinamik değerler içeren değişkenler için kullanılabilir.

```
let notSure: any = 4;
```

When you don't know what type you should use in TypeScript



Type'lar

Void

Geriye herhangi bir değer return etmeyen fonksiyonlarda ve null veya undefined değer içeren değişkenlerde kullanılır.

```
function warnUser(): void {  
  console.log("AA");  
}
```

Return Eden Fonksiyon

```
function add(a: number, b: number): number {  
  return a + b  
}
```

```
function test() : string | number {  
  return "test"  
  // return 5  
}
```

Interface

- Objelere ve sınıflara atanabilirler.

```
interface User {  
    name: string,  
    age: number,  
}  
  
let user: User = {  
    name: "Hasan",  
}
```

```
interface User {  
    name: string,  
    age?: number,  
}  
  
let user: User = {  
    name: "Hasan",  
}
```

Interface ve Extend Kavramı

```
interface User1 {  
    name: string,  
    age: number,  
}  
  
interface User2 extends User1 {  
    done: boolean;  
}  
  
let user: User2 = {  
    name: "Hasan",  
    // age: 23,  
    done: true  
}
```