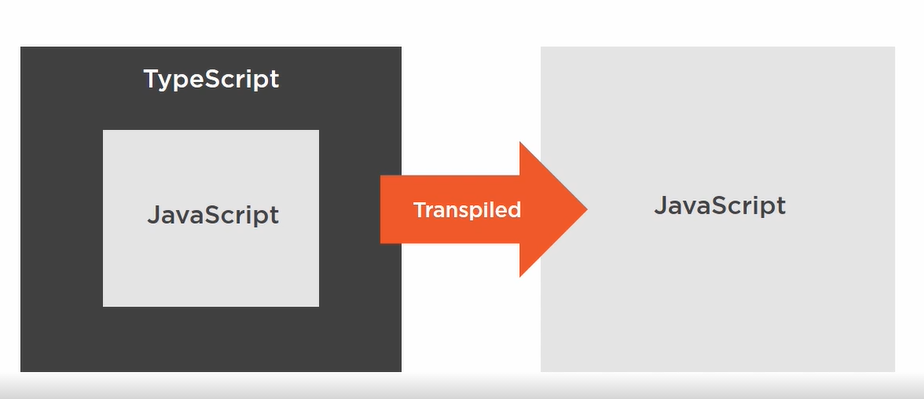
**Typescript:**

Overview



Features

1. Static Typing
2. Interfaces
3. Class Properties
4. Public and Private Accessibility

*Static Typing:*

Provide datatype to the variable, properties or object will not allow to store or pass wrong value type.

let name: String

let age: number

*Interfaces:*

Defining interface for an object that will define the shape of object

interface ICat {

name: string

age: number

}

let varName: ICat

varName = {

name: “Navnath”

age: 38

}

We can add **?** to any property to make optional

interface ICat {

name: string

age?: number

}

varName = {

name: “Navnath”

}

*Class Properties:*

Note that under class we declare name, color properties with data type

Class Cat {

name: string

color: string

constructor (name){

this.name = name

}

}

But below will also work and give us type safety

Class Cat {

name

color

constructor (name){

this.name = name

}

}

*Public and Private Accessibility:*

Class members are public by default you have to explicitly make them as private if you want to use it within the class.

Class Car {

name: string

play() { console.log( “Car Name is :” + this.name) }

}

let myCar = new Car();

myCar.play()

Above code will work as it is public by default

Class Car {

private name: string

private play() { console.log( “Car Name is :” + this.name) }

}

let myCar = new Car();

myCar.play()

The above code will not work because class properties are mark as private