**TYPESCRIPT PROTECTED KEYWORD**

# private vs protected keyword:

* The **private** and **protected** keyword makes the property inaccessible outside the class.
* The **private** property cannot be accessed even in the child class, while the **protected** property can be accessed in the child class.

# The **protected** keyword:

This keyword is used to make a property accessible in the extended (child) classes but not outside the class.

//? Protected Keyword.

*class* School {

*constructor*(protected readonly *name*: *string*, public readonly *address*: *string*, public readonly *phone*: *number*) { }

    getDetails(){

        console.log(`The ${this.name} is located at ${this.address} and contact number ${this.phone}.`);

    }

}

*class* DPS extends School {

*constructor*(*name*: *string*, public *standards*: *string*[],) {

        super(*name*, 'Indore (M.P)', 9407541209);

    }

    //? Method overriding.

    getDetails(){

        console.log(`The ${this.name} is located at ${this.address} and contact number ${this.phone}, teaching for standards ${this.standards}`);

    }

}

*const* delhiPublicSchool = new DPS('Delhi Public School', ['1st', '2nd']);

//! delhiPublicSchool.name [Error]: property cannot be accessed.

delhiPublicSchool.getDetails();