Typescript:

Superset

JavaScript is typescript.

Browser does not understand typescript.

Browser works on client-side.

Key feature is type.

Client-side website development.

Typescript support

What is type? Type which describe the properties and method of object and what build-in type-Of operator would be it as

Typescript:

1. Assignability
2. Annotation
3. Type
4. TypeShape

Two type of error type error and syntax error

* Super set of JS
* Type+ js
* Browser does not understand.
* Compiler required.
* Object oriented.
* Initialize the project tsc --init
* tsc –version
* Compile tsc filename.ts
* Execute node filename.js.
* Both compile and execute tsc node filename.ts
* variable name started with lowercase and Class, Interface, Enum start with the uppercase.
* camelCase for the variable
* /\* let and var both for variable declaration but var is function scoped and let is blocked scoped
* var is declared and let cannot re-declared and must be.
* use let and constant not use var.
* \*/
* Template literal `${variable}
* Data type: number, string, null, undefined, void, object (properties and with its function), array, enum (collection of constant values) , any(never using)
* Operator in TS:
  + - Arithmetic (+, -, /, %)
    - Shorthand math (+=, -=, \*=, /=)
    - Increment /decrement (--, ++)
    - Logical operator (&&, ||, ^)
    - Conditional operator (<, > <=, =>,!==)
    - Ternary operator (? :) (condition)? true part: false part
* Conditional statement if -else and switch
* For loop (initialization; condition; increment /decrement)
* For of and For In
* For each => perform operation specific object item
* While (condition){execute and increment and decrement }
* Do while loop.
* Break, continue, Nested loop.
* Function (repetition of code or work multiple time)

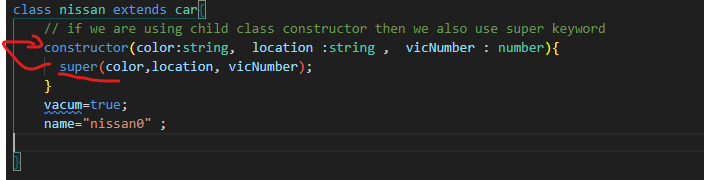
Name function and arrow function

* Object (In TypeScript, the object type represents the non-primitive type that represents any value that is not of type number, string, Boolean, symbol, null, or undefined.)

Firstly, make an interface that consists of data type of values uses in object.

* Math. Random
* Array
* • // array collection of consecutive memory location

We can also make datatype= type myNewType=string |number|null.

* • // collection of similar data
* Push, pop, splice, slice
* Object ki {}
* Array ki square bracket []
* Function ki ()
* An array is an object.
* // object is something possess
* //object has properties and its function
* // notation {}
* // object class is super class
* //class is blueprint
* // example divided of piece in equal pieces every piece is object
* map () function make new array and perform operation on each element and store in new array
* but for each function perform function on each specific element return value in form of Boolean
* filter makes new array make array.
* Zilch =0;
* This keyword is used to indicate that part current object where we have ?
* Class
* This keyword
* Constructor
* Inheritance
* Extend class with keyword extends.
* The super keyword is used to call functions on an object's parent.
* 
* Overriding of function

Overriding of PROPERTY  constructor(color:string,  location :string ,  vicNumber : number| string){

OOP has four major concepts:

[**abstraction**](https://stackify.com/oop-concept-abstraction/), complex remove to make simple**.**

[**encapsulation**](https://stackify.com/oop-concept-for-beginners-what-is-encapsulation/), restrict (public, private )

[**inheritance**](https://stackify.com/oop-concept-inheritance/)  ,link(child parent )

[**polymorphism**](https://stackify.com/oop-concept-polymorphism/). ( one thing perform many action )