In JS objects are in key : value pairs, where a value can also be a function, and for that we name method of the object

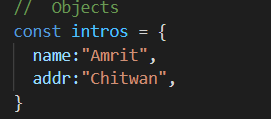
Don’t ever use Arrow functions as object’s methods, use regular functions

We can create objects in JS from the class also by using the new keyword

OR

we can make directly

like



We use const keyword to declare objects

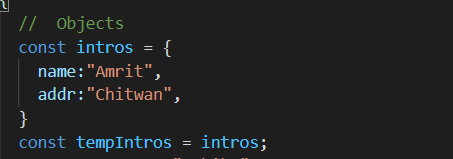
For normal understanding, whatever is initialized using const cannot be changed, i.e remains like a constant

But in memory level

const actually declares a constant memory

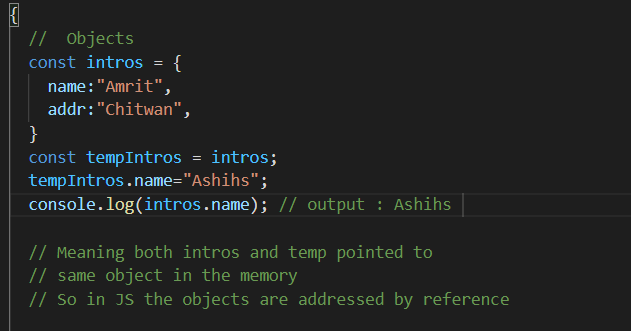
Also

When we do



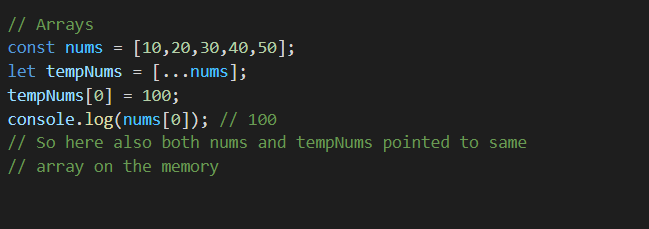
both intros and tempIntros are pointing to the same objects in the memory

**In javascript a variable stores the reference of the object in the memory**



since both intros and tempintros points to the same oject so when we change the value in tempIntros the value in intros also got changed

Same happens in Arrays also

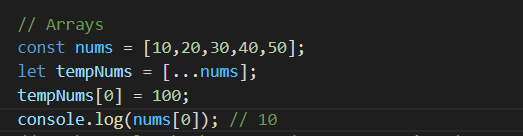


Since here both nums and tempNums points to the same Array on the memory, so when we change the first item in the tempNums the first item of the nums also got changed

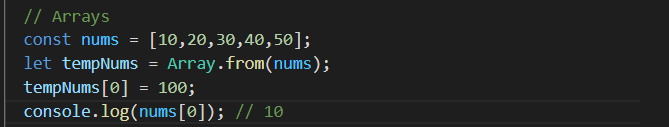
In array if we ever needed to copy the array without copying the reference than we can do a bunch of things

We already discussed in the Array in JS folder

1. We can use … operator

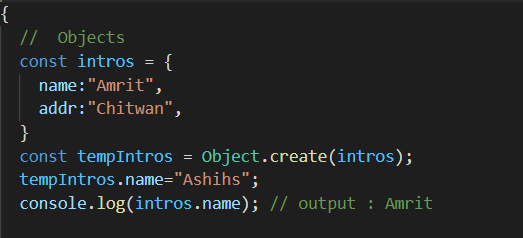


1. We have a built in method for only this, Array.from()



In objects if we needed to copy the object without copying the reference we can do this by

1. Object.create()



1. Here also we can use the spread operator …

