Abstract equality (==) 29 September 2022 20:08	
Ignore (!) in Abstract equality	
Ext. ! To Number (y) (avoid!) (this is	not a logical operator).
-> typeOt NaN -> // Number	
-> It doesn't actually mean not a r	number, rather it gives notion
of an invalid number.	
Ex:- "Laxman" -> To Number -> N	aN
NaN ==== NaN -> False	
NaN == NaN (calls === again)	
NaN is the only primitive value in Js identity property and is not equal to	that doesn't blow itself.
> KKKKK function compute() [
returns Val;	
let (x) = compute();	
Now suppose result of compute functions and we have to detect if x is NAN or	thion can be NaN. not ??
> Solotion: - we have isNaN() 11 h	some issues.
this is NaN function coerces the argument of NaN. Now string is not NaN &	ment before checking
isNaN("abc") // gives true	
Number·is NaN (NaN) // true Number·is NaN ("abc") // false	This function doesn't do Locreion
Browser is a runtime, conso part of runtime.	le.log doesn't comes from Js.
part of runtime.	
When to use == 1 when t	O USE === ??
I Shile Aeveloping and whicht	encounter multiple programming
While developing apps, we might	
i, Based on how the language exercised language in the following:-	ectues the code we can categorise
	complete code & throw if any error) the code.
	the code.
b) Interpreted: - shell script (2) Hubrid: - balls compiled s	Testernosted (TC mullion)
C) Hybrid: - both compiled &	and product (33) figures.
Console·log ("Hello") 7. console·log ("World");	conjoic.log("Hello") conjoic.log(a)
doesn't print anything (looks like compiled)	prints flello & then throws error
(looks like compiled)	(looks like interpreted)
A JS reads your code twice	
1) First compiles	
2) Second time execuses	
	(Not present in official documendation)
DUT	It is a phenomenon in 15 when, it becomes
let $z=9$ x.toStringL)//'9'	trying to make 9 behave as an object his is what boxing means.
"abc": to String(); // abc'	thus is what boxing means.

typeof won't change even after making that behave as an object