28. OOPS-4 : Call, Apply, Bind

Agenda :

Polyfills ⟶ Call, Bind & Apply

Native application → ✓✓✓

⟩ • function & how we invoke them.

 # Call method :

- Invoke a function & explicitly set the content (this) value within that function.

```
function fun () {
    console.log ('Hi')
}

fun();.
```

○ ALLOWS TO SPECIFI VALUE OF this WITHIN A FUNCTION ,

⌐ func.call (obj, p1, p2, p3....)

apply :

- Same as call
- except → parameters are passed in form
  of array (instead of comma
  separated)

  Eg. func.apply (obj, [p1,p2,p3...])

Dynamic Argument Lists (thing of Apply)

  numbers = [3,7,5,6,7].
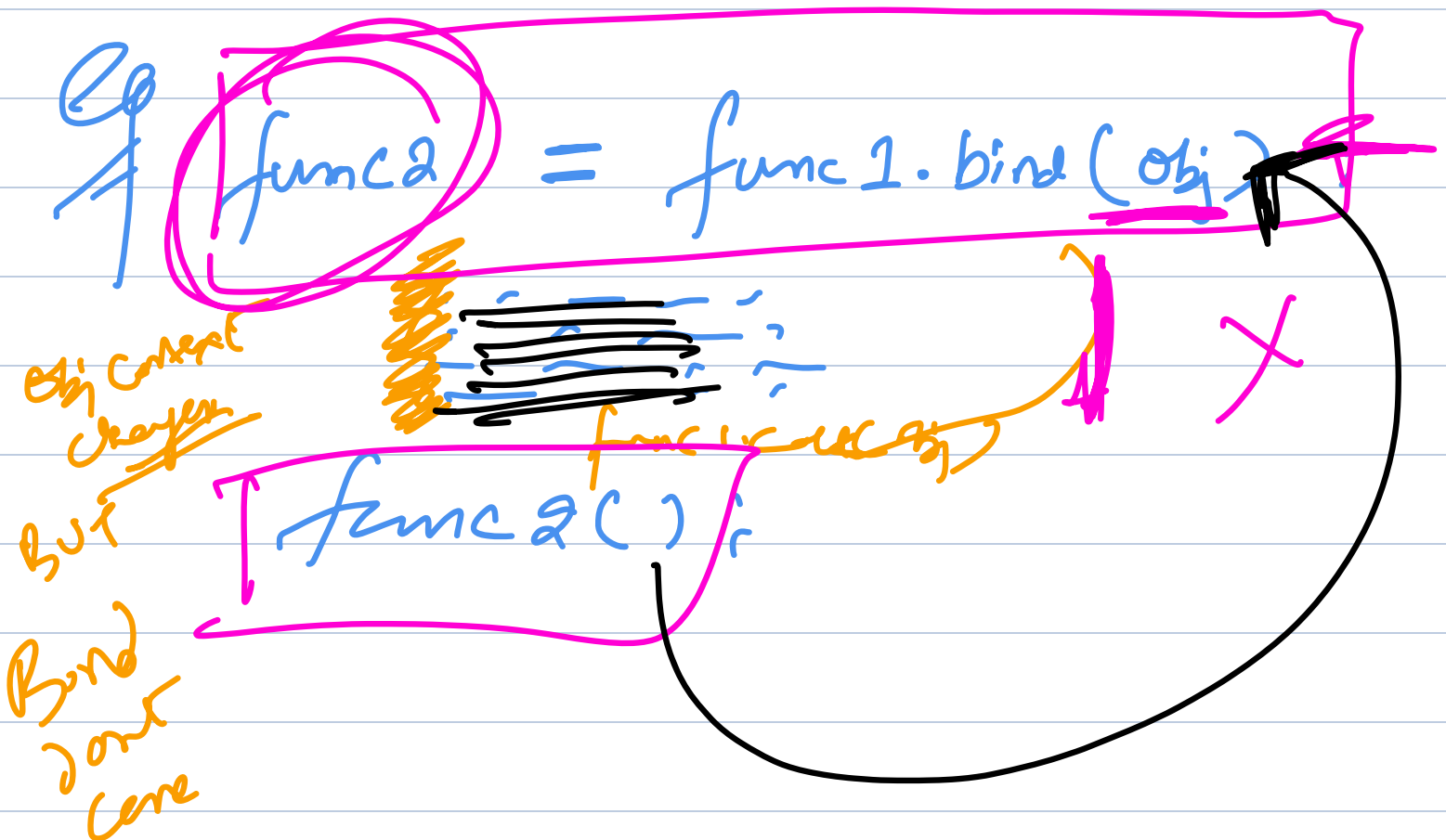
Eg Math.max. apply (null, numbers)

  (17,   ,

bind

- Calling a function with a context ; given 'this' value

- Don't call immediately
- Saves reference / context for future.
- Call later.

$$ \text{eg} \quad \boxed{func2 = func1.bind(obj)} $$

obj context
changes

BUT

Bind
don't
care

$$ func2() $$

X

CALL	APPLY	BIND
Immediate Executing fnne.	Immediate Executing fnne.	Later exec of function

< Pass Context , this value >.

Bounds to given this value for future use

| Params Comma Separated | Param in Array form | Comma Separate |

Polyfills intyitm

$$\downarrow$$

[array . map] (el => --- :)

~~goto~~

[1,2] . forEach (. —)

func . call (.-- ~)

'Hello' . toUppercase ()

T 17 Oct

S 19 Oct

18 Oct

→ Polyfill

→ Promise Constructor

Shallow / Deep
copy

2y oc

T 22 Oct

23rd

2 weeks

Diwali +

Backlog +

Revise
+
ConKot

Tuesday
5 Nov

Reans ↑↑