

5th Class

Polyfills

- filter,
 - reduce
 - call, bind & apply
 - flat
-

* let arr = [18, 19, 17, 16];

* arr.filter((element) => {
 return item > 18;
})

★ sum = 0 , \rightarrow accumulator

★ for (let i=0; i < length; i++) {
 return sum = [sum + arr[i]]
}

★ return sum

\downarrow
logic will be
given by
end user

latest value
of sum
depends
on ?

\Rightarrow prev sum
+ curr

[1 , 2 , 3 , 4]

before sum \Rightarrow 0

after sum \Rightarrow 0 + 1 = 3

polynis > JS reduce.js > ...

```
1 let arr = [1, 2, 3, 4, 5];
2
3 let sum = arr.reduce((accumulator, currEle) => {
4   | return accumulator + currEle;
5   | });
6
7 console.log(sum);
8 |
```



accumulator \Rightarrow sum
 \downarrow
0

	1	2	3	4	5
accum	0	1			
accumulator	0 + 1	1 + 2			

function sum() {

}

* obj.tempFn = this

* sum.call();

* tempFn

obj

let obj = { tempFn = sumDefi }	function sum() { // some thing }
--------------------------------------	--

* sum.call(obj);

* obj.tempFn = sumDefi(this)

* const res = obj.tempFn();

* delete obj.tempFn

* return res