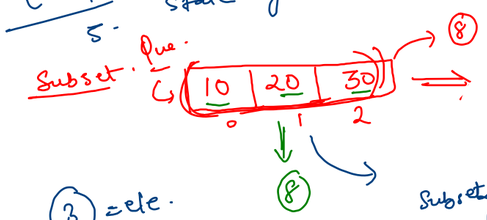
1. Subset. → ID. Many. } Wrap up.
2. 2D - Memory
3. Demo.
4. Matrix Multi.
5. State of Wakanda.

Subset. Que.



Subset

1.	—	—	—
2.	—	—	30
3.	—	20	—
4.	—	20	30
5.	10	—	—
6.	10	—	30
7.	10	20	—
8.	10	20	30

	10	20	30	i
1.	0	0	0	0
2.	0	0	1	1
3.	0	1	0	2
4.	0	1	1	3
5.	1	0	0	4
6.	1	0	1	5
7.	1	1	0	6
8.	1	1	1	7

0 → —
1 → value.

1.	—	—	—
2.	—	—	30
3.	—	20	—
4.	—	20	30
5.	10	—	—
6.	10	—	30
7.	10	20	—
8.	10	20	30

$i \rightarrow 0 < 2^n$

3 = ele. → 2^3

n = ele → 2^n

2^n ~ ele ✓ ✗

⇒ ② ② ② = ⑧

Binary.

Approach

total = 2^n

10 20 30 ⇒

for (i = 0 ⟶ <2n)

{ i = 0 → convert → binary of 'i'

0 0 0

dec → binary.

0 → 000
1 → 001
2 → 010

if (✗ == 0)
{ — ;
}
else { arr[i] + " " ;
}

n → length of arr.

— — — = 2^3

000 → 0 --- with 3 digit.

for (0 ⟶ $<2^3$) // total of subset.

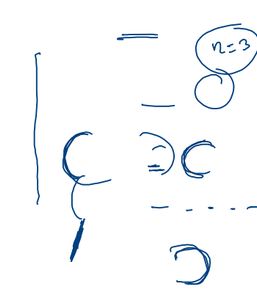
{ ⇒ 0 → binary: 000

if (id = = 0)
{ —
}
else { val;
}
}

⑧ → [0 --- 7]

Que. subset.

int n = 3



```
int total = (int)Math.pow(2,n);

for(int i = 0; i<total; i++){

    // i == 0 --> 0 0 0
    String str = "";
    int temp = i;
    for(int j = n-1; j>=0; j--){
        int r = temp % 2;
        temp = temp/2;

        if(r == 0){
            str = "-\t" + str;
        }
        else{
            str = arr[j] + "\t" + str;
        }
    }
    System.out.println(str);
}
```

i=0 10 20 30

r = 0

i=1
1. r = 1%2 = 1
2. r = t%2 = 0%2 = 0
→ t = t/2 = 0/2 = 0
3. r = t%2 = 0%2 = 0
t = t/2 = 0

i=2 2<8

str = ""
t = ✗✗✗0
j = n-1 = 2 ✓ ✗ -1

j1. 2≥0
r = t%2 = 2%2 = 0
t = t/2 → 2/2 = 1

j2. 1≥0
r = t%2 = 1%2 = 1
t = 1/2 = 0

j3. 0≥0
r = t%2 = 0%2 = 0
t = t/2 = 0/2 = 0

"30 -"
"- 30 -"
"- - 30 -"