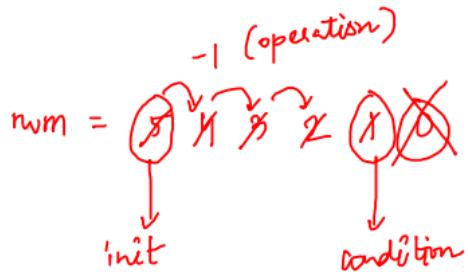
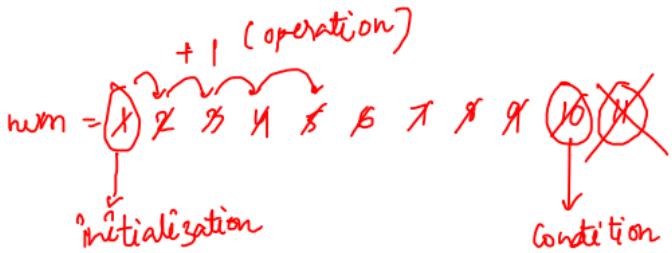


```
419 |   for (let num = 1; num <= 10; num++) {  
420 |     console.log(`pushup ${num}`);  
421 |   }
```

- * In order enter the for loop condition must be true / satisfied
 - * after the code inside for loop is executed it will perform operation.
 - * operation condition cycle repeats
 - * when condition is false, come out of for loop

sp: pushup 1 pushup 5 pushup 9
pushup 2 pushup 6 pushup 10
pushup 3 pushup 7
pushup 4 pushup 8

- ① Initialization, num = 1
 - ② num <= 10 \Rightarrow 1 <= 10 \rightarrow true
 - ③ num++ \Rightarrow num = 2
 - ④ num <= 10 \Rightarrow 2 <= 10 \rightarrow true
 - ⑤ num++ \Rightarrow num = 3
 - ⑥ num <= 10 \Rightarrow 3 <= 10 \rightarrow true
 - ⑦ num++ \Rightarrow num = 4
 - ⑧ num <= 10 \Rightarrow 4 <= 10 \rightarrow true
 - ⑨ num++ \Rightarrow num = 5
 - ⑩ num <= 10 \Rightarrow 5 <= 10 \rightarrow true
 - ⋮
 -  num++ \Rightarrow num = 11
 -  num <= 10 \Rightarrow 11 <= 10 \rightarrow false



$\text{num} = 0$

(or)

```
for (let num = 5; num >= 1; num--) {  
    /* code */  
}
```

```
429 | for (let num = 5; num >= 1; num--) {  
430 |   console.log(`pushup ${num}`);  
431 | }
```

op: pushup 5
pushup 4
pushup 3
pushup 2
pushup 1

- ① num = 5 (only once)
- ② num $\geq 1 \Rightarrow 5 \geq 1 \rightarrow \text{true}$
- ③ num-- $\Rightarrow num = 4$
- ④ num $\geq 1 \Rightarrow 4 \geq 1 \rightarrow \text{true}$
- ⑤ num-- $\Rightarrow num = 3$
- ⑥ num $\geq 1 \Rightarrow 3 \geq 1 \rightarrow \text{true}$
- ⑦ num-- $\Rightarrow num = 2$
- ⑧ num $\geq 1 \Rightarrow 2 \geq 1 \rightarrow \text{true}$
- ⑨ num-- $\Rightarrow num = 1$
- ⑩ num $\geq 1 \Rightarrow 1 \geq 1 \rightarrow \text{true}$
- ⑪ num-- $\Rightarrow num = 0$
- ⑫ num $\geq 1 \Rightarrow 0 \geq 1 \rightarrow \text{false} \times$

Q: Sum of 1st N natural numbers :

Sum = \emptyset
1
2
3
4
5
6
15

Eg: $n = 5$

Op: $(\textcircled{1} + \textcircled{2} + \textcircled{3} + \textcircled{4} + \textcircled{5}) = \textcircled{15}$

Eg: $n = 3$

Op: $1 + 2 + 3 = \textcircled{6}$

let sum = 0;

for (let num = 1; num <= N; num++) {

sum = sum + num;

}

num = ~~1 2 3~~

sum = ~~0 3 6~~

$n = 3, \text{ sum} = \emptyset \times \cancel{1} \cancel{2} \cancel{3} 6$

① num = 1

② num <= 3 $\Rightarrow 1 \leq 3 \rightarrow \text{true}$

③ sum = sum + num $\Rightarrow \text{sum} = 0 + 1 = 1$

④ num++ $\Rightarrow \text{num} = 2$

⑤ num <= 3 $\Rightarrow 2 \leq 3 \rightarrow \text{true}$

⑥ sum = sum + num $\Rightarrow \text{sum} = 1 + 2 = 3$

⑦ num++ $\Rightarrow \text{num} = 3$

⑧ num <= 3 $\Rightarrow 3 \leq 3 \rightarrow \text{true}$

⑨ sum = sum + num $\Rightarrow \text{sum} = 3 + 3 = 6$

⑩ num++ $\Rightarrow \text{num} = 4$

⑪ num <= 3 $\Rightarrow 4 \leq 3 \rightarrow \text{false} \times$

Another Approach :

$$n = 5$$

$$\Rightarrow \frac{n(n+1)}{2}$$

$$\Rightarrow \frac{5(5+1)}{2} = \frac{5 * 6^3}{2} = 15$$

instructions $\rightarrow n$ times

~~2nd approach~~
instructions $\rightarrow 1$ time

* In general, your online IDE's

can run at max 10^8 instructions

(i.e; only 1s
is allotted for
your code)

$10^8 \rightarrow 1$ second

for loop (10^5 - 10^6)
times

(TLE)

Q: Factorial of a number :

eg: $n = 5$

Op: $n! \Rightarrow n * n-1 * n-2 * \dots * 1$

$$5! \Rightarrow 5 * 4 * 3 * 2 * 1 = 120$$



$$\begin{array}{ll} \checkmark \text{prod} = 1 & \checkmark \text{prod} = 1 \\ \cancel{\cancel{\cancel{\cancel{\cancel{}}}}} & \cancel{\cancel{\cancel{\cancel{\cancel{}}}}} \\ 5 & 1 \\ 4 & 2 \\ 3 & 3 \\ 2 & 4 \\ 1 & 5 \\ \hline \text{prod} = 120 & \end{array}$$

```
306 | for (let num = 1; num <= n; num++) {  
307 |   fact = fact * num;  
308 | }  
309 | /*  
310 | for (let num = n; num >= 1; num--) {  
311 |   fact = fact * num;  
312 | }  
313 | */
```

$$n=5 \quad \text{fact} = 1 * 2$$

$$1^{\text{st}} \text{ iter} \Rightarrow \text{num} = 1, \text{num } \leftarrow 5
\\ \text{fact} = 1 * 1 = 1$$

$$2^{\text{nd}} \text{ iter} \Rightarrow \text{num} = 2, \text{num } \leftarrow 5
\\ \text{fact} = 1 * 2 = 2$$

$$3^{\text{rd}} \text{ iter} \Rightarrow \text{num} = 3, \text{num } \leftarrow 5
\\ \text{fact} = 2 * 3 = 6$$

$$4^{\text{th}} \text{ iter} \Rightarrow \text{num} = 4, \text{num } \leftarrow 5
\\ \text{fact} = 6 * 4 = 24$$

$$5^{\text{th}} \text{ iter} \Rightarrow \text{num} = 5, \text{num } \leftarrow 5
\\ \text{fact} = 24 * 5 = 120$$

Q: fren sum:

eg: $n = 12$

op: $2 + 4 + 6 + 8 + 10 + 12 = 42$

init \downarrow operation \downarrow condition

let sum = 0;

for(let num = 1; num <= n; num++) {

if ($num \% 2 == 0$) {

sum = sum + num;

}

}

$n = 12$

num = 1	sum = \emptyset
2	\emptyset
3	\emptyset
4	4
5	9
6	15
7	22
8	30
9	39
10	49
11	59
12	72

Another Approach :

```
let sum = 0;           i <= 12 X  
for(let num = 2; num <= n; num += 2) {  
    sum = sum + num;  
}
```

$n = 12 \quad \text{sum} = \emptyset$
 $\text{num} = 2 \quad \cancel{\emptyset}$
 $X \quad \cancel{12}$
 $\beta \quad 30$
 $\gamma \quad 36$
 $X0 \quad \underline{42}$
 ~~$X1$~~

Formula :

$$\frac{n(n+2)}{4} \Rightarrow \frac{12 * 14}{4} \Rightarrow 42.$$

* handle
when n is
odd

$$\Rightarrow n = 13$$

$$\frac{13(13+2)}{4} = \frac{13 * 15}{4} = 48.75$$

if (n is odd) {
 $n = n - 1$
}

but $n = 13 \rightarrow \text{ans} = 42$ which is same when
 $n = 12$

* break :

→ whenever we see this, Just get out of the current loop

* Continue :

→ whenever we see this, directly move to operation.

i.e; ignore the current iteration and move to next iteration.
(all the lines below it)

Q: check prime :

tg: $n = 5$

op: 5 is a prime number

tg: $n = 10$

op: 10 is not a prime number



$$10 \div 2 = 2^0$$

* n is only divisible by 1 and n
→ prime

If it is divisible by some other numbers
→ not prime

(L) $\Rightarrow n-1$

let isPrime = true;

for (let num = 2; num <= n-1; num++) {

if ($N \% num == 0$) {

isPrime = false;

break;

}

}

```

355 readline.question("", (n) => {
356   n = Number(n);
357
358   let isPrime = true;
359   for (let num = 2; num <= n - 1; num++) {
360     if (n % num == 0) { → not a prime
361       isPrime = false;
362       break; →
363     }
364   }
365
366   if (isPrime == true) {
367     console.log(`#${n} is a prime number`);
368   } else {
369     console.log(`#${n} is not a prime number`);
370   }
371
372   readline.close();
373 });

```

* flag variable technique

$\rightarrow \text{console.log('n is not a prime number')}$;

$\rightarrow \text{console.log('n is a prime number')}$

2, 3, 5

$$5 \cdot 1 \cdot 2 \cdot 2 = 0$$

$$5 \cdot 1 \cdot 3 = 2 \cdot 0$$

$$5 \cdot 1 \cdot 4 = 2 \cdot 0$$

2,

$$10 \cdot 1 \cdot 2 = 0$$

~~10 is not a prime~~
~~10 is a prime number~~