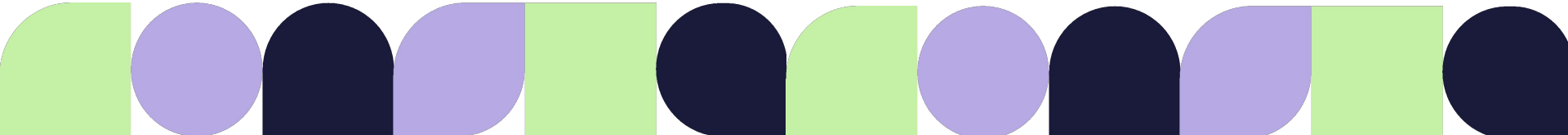


# Loops

### While



# Loops



- Mostrar todos los números de 1 a n aumentando de 1 en 1 donde n lo ingresa el usuario en un prompt.
- Mostrar todos los números de 1 a N aumentando de 2 en 2 donde n lo ingresa el usuario en un prompt.
- Mostrar todos los números de N a 1 disminuyendo de 1 en 1 donde n lo ingresa el usuario en un prompt.
- Escribir utilizando console.log la tabla del 9 hasta 9x10.
- Pedir al usuario que ingrese un número en un prompt, hacer la suma de todos los dígitos, validar que el número ingresado no contenga letras.
- Realizar la suma de todos los números pares entre N y M donde N y M los ingresa un usuario.
- Realizar la sumatoria de los primeros N números, donde N es ingresado por el usuario.
- Realizar el factorial de los primeros N números.



# Ejercicios de divisores con while (o for)

- Encontrar todos los divisores de un número.
- Determinar si un número ingresado por el usuario en un loop es primo o no, validar que el número ingresado sea mayor o igual a dos.
- Crear un programa que determine si un número es perfecto o no, (se dice que un número es perfecto si el número es igual a sus divisores, ejemplos  $6 = 1 + 2 + 3$ )

## ### Doble loop

- Generar los primeros N números primos, donde n es ingresado por el usuario.
- Generar los primeros N números perfectos.

# Patrones con Loop anidados

Dibujar los siguientes patrones ocupando document.write, para rellenar los espacios vacíos se debe imprimir un espacio vacío.

Cuadrado lleno:

```
*****
*****
*****
*****
*****
```

Cuadrado hueco:

```
*****
*   *
*   *
*   *
*   *
*****
```

Tablero de ajedrez:

```
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
```

Pirámide izquierda:

```
*
**
***
****
*****
```

Diamante:

```
*
***
*****
*****
*****
*****
***
*
```

Pirámide invertida:

```
*****
*****
*****
***
*
```

Pirámide centrada:

```
  *
 ***
*****
*****
*****
```