Estructura de Repetición: While

Repaso

- La clase anterior aprendimos a hacer **bucles** con la estructura for.

- Con for podemos repetir un bloque de código, mientras que una **condición** sea verdadera.

- En el for podemos utilizar una variable que actúa como **contador** de las iteraciónes.

Repetir un bloque de código 10 veces

```
> for (let contador = 0; contador < 10; contador++) {
      console.log("Hola " + (contador+1));
  }
  Hola 1
  Hola 2
  Hola 3
  Hola 4
  Hola 5
  Hola 6
  Hola 7
  Hola 8
  Hola 9
  Hola 10
```

While

- Otra estructura de repetición que tenemos en la programación es en while.

- while (condicion) {

  /* bloque while */

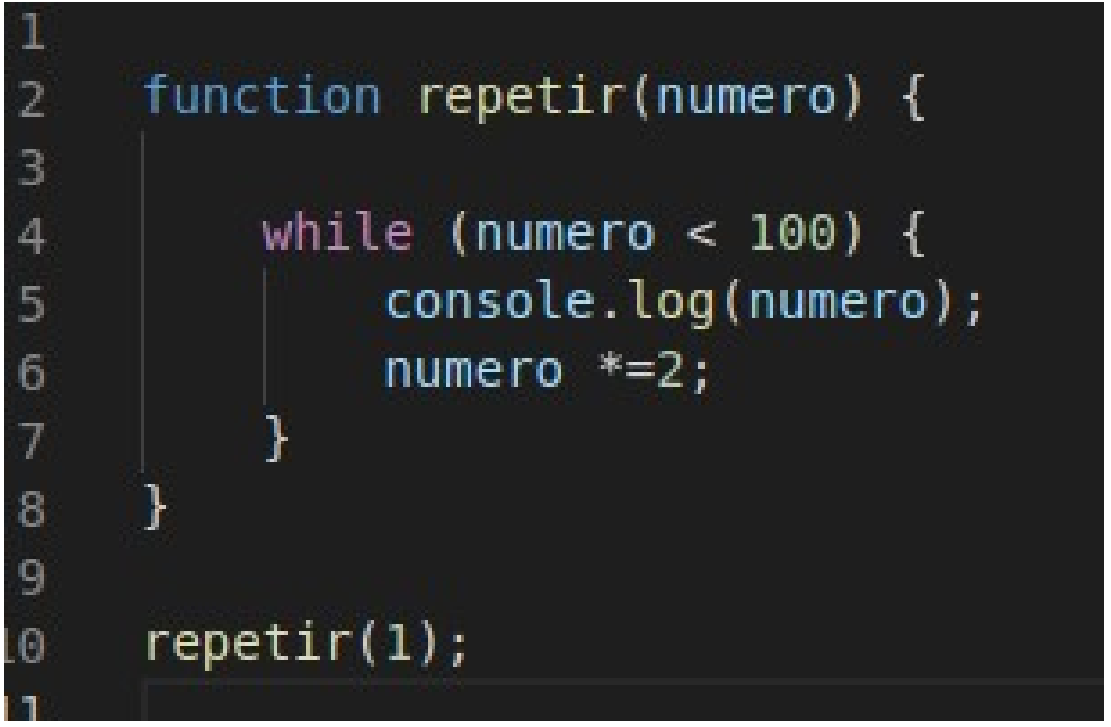
  }

Un ejemplo...

```
let x = 0;

while (x !== 4) {
    alert("La variable x no vale 4 :) ");
    x++;
}
```

Analicemos el siguiente ejemplo...



```
1
2   function repetir(numero) {
3
4       while (numero < 100) {
5           console.log(numero);
6           numero *=2;
7       }
8   }
9
10  repetir(1);
```

Analicemos otro ejemplo...

```
let nombre = null;

while (nombre === null || nombre === "")
{
    nombre = prompt("Ingrese su nombre");
}

alert("Hola " + nombre);
```

Ejercitemos nuestra lógica y creación de algoritmos...

Ejercicio 1

- Programar una función llamada calcularPotencia(), la cual calcule y muestre por consola el resultado de calcular la potencia n ^ x
- Los valores de la base y el exponente (n y x) serán solicitados como parámetros de la función.

```
function calcularPotencia(base, exponente) {
    // programar!
}
```

Una posible resolución:

```
function calcularPotencia(base, exponente)
{
    let resultado = 1;
    for (let i = 0; i < exponente ; i++) {
        resultado *= base;
    }
    console.log(resultado);
}
```

Ejercicio 2:

- Programar una función que solicite dos números, y determine cuál es el menor.

Ejercicio 2:

- Programar una función que solicite dos números, y determine cuál es el menor.

```
function numeroMenor(a, b){
    if (a === b) {
        console.log("Ambos numeros son iguales");
    } else if (a > b) {
        console.log(b + " es menor que " + a);
    } else {
        console.log(a + " es menor que " + b);
    }
}
```

Ejercicio 3: parte a)

- Crear un programa que solicite tres notas al estudiante. Verifique que sean notas menores a 12. En caso que las notas sean validas, llamar a una función que calcule el promedio entre las tres notas.
- Si el promedio es Mayor o igual a 8, mostrar el mensaje “Exonerado”
- Si el promedio está entre 4 y 7, mostrar el mensaje “Examen diciembre”
- Si el promedio es menor que 4, mostrar el mensaje “Examen febrero.”
-

```
function promedio(nota1, nota2, nota3) {
    const promedio = (nota1 + nota2 + nota3) / 3;

    if (promedio >= 8) {
        alert("EXONERADO");
    }
    else if (promedio >= 4) {
        alert("EXAMEN DICIEMBRE");
    } else {
        alert("EXAMEN FEBRERO");
    }
}

const nota1 = parseInt(prompt("Ingrese nota 1 :"))
const nota2 = parseInt(prompt("Ingrese nota 2 :"))
const nota3 = parseInt(prompt("Ingrese nota 3 :"))

if (nota1 <= 12) {
    if (nota2 <= 12) {
        if (nota3 <= 12) {
            promedio(nota1, nota2, nota3);
        } else {
            alert("ERROR: La nota 3 debe ser menor que 12");
        }
    } else {
        alert("ERROR: La nota 2 debe ser menor que 12");
    }
} else {
    alert("ERROR: La nota 1 debe ser menor que 12");
}
```

Ejercicio 3, parte b)

- Agregar otra funcionalidad, para informarle al usuario si su condición es reglamentada o libre
- Un estudiante queda en calidad de reglamentado si sus faltas son menores a 25, en caso contrario, queda en calidad de libre
- Pedir las inasistencias, hacer el calculo, y mostrar un mensaje.

Ejercicio 3, parte b), resolución:

```
    alert("ERROR: La nota 1 debe ser menor que 12");
}


function condicionEstudiante(faltas) {
    if (faltas < 25)
        alert("Calidad: REGLAMENTADO");
    else
        alert("Calidad: LIBRE");
}

const faltas = parseInt(prompt("Ingrese sus inasistencias"));
condicionEstudiante(faltas);
```

Próxima clase:

- Veremos otra estructura de repetición: **Do – While**
- Trabajaremos con **funciónes**