# Expressions and Operators

- **Arithmetic Operators**

1. addition ( + )
2. subtraction ( - )
3. multiplication ( * )
4. division ( / )
5. remainder ( % )
6. exponentiation ( ** ) to square root use " ** 1/2 "
7. increment ( ++ )
8. decrement ( -- )

- **Assignment (Atribuição)**

1. num1 = num1 + 2
2. num1 += 2
3. num1 -= 2
4. num1 *= 2
5. num1 /= 2
6. num1 &= 2
7. num1 **= 2

```
let num1 = 3
let num2 = 2

console.log(num1 + num2)
console.log(num1 - num2)
console.log(num1 * num2)
console.log(num1 / num2)
console.log(num1 % num2)
console.log(num1 ** num2)

console.log(--num1)
console.log(++num1 + num2)
```

- **Comparison Operators (Boolean)**

1. >  >=
2. <  <=
3. ==  !=
4. ===  !==

**Ex:**

```
console.log(num1 > num2)
console.log(num1 <= num2)
console.log(num1 == num2)
console.log(num1 != num2)
```

- **Logical Operators (Boolean)**

1. and ( && )
2. or ( || )
3. not ( ! )

**Ex: Ir para a Walt Disney?**

```
let economizarDinheiro = true
let juntarDinheiro = true

console.log(economizarDinheiro && juntarDinheiro)

let guardarPoupanca = false
let pegarEmprestimo = true

console.log(guardarPoupanca || pegarEmprestimo)

console.log(!guardarPoupanca)
```

# TRUTH TABLE

## Conjunction, Disjunction

| P | Q | P ^ Q | P v Q |
|---|---|---|---|
| V | V | V | V |
| V | F | F | V |
| F | V | F | V |
| F | F | F | F |

## Negation

| P | Q | ~P | ~Q |
|---|---|---|---|
| V | V | F | F |
| V | F | F | V |
| F | V | V | F |
| F | F | V | V |

## Exercise

| P | Q | ~P ^ Q | P ^ ~Q |
|---|---|---|---|
| V | V | | |
| V | F | | |
| F | V | | |
| F | F | | |

| P | Q | ~P v Q | ~(P v ~Q) |
|---|---|---|---|
| V | V | | |
| V | F | | |
| F | V | | |
| F | F | | |

| P | Q | P v (Q ^ P) | (P ^ Q) v ~Q |
|---|---|---|---|
| V | V | | |
| V | F | | |
| F | V | | |
| F | F | | |

- **Operators**

1. Binary

```
let n1 = 8
let n2 = 7

console.log(n1 + n2)
console.log("Test " + n2)
```

2. Unary

```
console.log(n1++)
console.log(n1)
console.log(++n1)
console.log(typeof n2)

const fruits = ['banana', 'maçã', "Uva"]
delete fruits[1]
console.log(fruits)
```

3. Ternary

**test ? true : false**

```
let avarege = 7
console.log(avarege >= 7 ? 'Aprovado': 'Reprovado')
```

- **Truthy e False**

* Cuidado com valores onde o booleano é considerado obrigatório (condicionais e loops)

| Truthy | False |
|---|---|
| 1, 1.5, -1 | 0 |
| " ", "0", "false" | “” |
| { } | null |
| [ ] | undefined |
| Infinity, -Infinity | NaN |

```
console.log( 1 ? 'yes' : 'no')
console.log( "" ? 'yes' : 'no')
console.log( "0"? 'yes' : 'no')
console.log( {} ? 'yes' : 'no')
console.log( null ? 'yes' : 'no')
console.log( Infinity ? 'yes' : 'no')
```

- **Expressions Grouping Operator**

```
let total = (2 + 3) * 3
console.log(total)
```

- **Operator Precedence**

| | |
|---|---|
| **Grouping** | ( ) |
| **Negation, Increment and Decrement** | ! ++ -- |
| **Multiplication and Division** | * / |
| **Addition and Subtraction** | + - |
| **Comparison** | < <= > >= |
| **Equality** | == != === !== |
| **And** | && |
| **Or** | \|\| |
| **Conditional** | ? : |
| **Assignment** | = += -= *= %= |

```
console.log(7 + 8 / 2)
console.log((7 + 8) / 2)
console.log(3 > 2 && 4 < 10)
console.log(3 < 2 || 4 < 10)
console.log(6 < 2 || 4 > 10)
console.log(3 > 2 > 1) // true == 1 false == 0
console.log(3 > 2 && 2 > 1)
console.log(true == 1)
console.log(true === 1)
```

**Referência:** https://developer.mozilla.org/pt-BR/docs/Web/JavaScri