

L'opérateur conditionnel (ternaire)

```
let condition;
let vraie;
let faux;

// condition ? vraie : faux

function getFee(isMember) {
  return (isMember ? '$2.00' : '$10.00');
}

console.log(getFee( isMember: true));
// expected output: "$2.00"

console.log(getFee( isMember: false));
// expected output: "$10.00"

console.log(getFee( isMember: null));
// expected output: "$10.00"
```

"Le prix est : " + (estMembre ? "15 €" : "30 €")

La condition est de definir le prix, si la personne est membre alors le prix sera de 15 € et si la personne n’est pas membre alors le prix sera de 30 €

Opérateur de coalescence des nuls (Nullish)

```
const valeurNulle = null;
const texteVide = ""; // falsy
const unNombre = 42;
const undef = undefined;

const valA = valeurNulle ?? 0;
const valB = texteVide ?? "valeur par défaut pour B";
const valC = unNombre ?? 44;
const valD = undef ?? valeurNulle;

console.log(valA); // "valeur par défaut pour A"
console.log(valB); // "" (car la chaine vide n'est ni `null` ni `undefined`)
console.log(valC); // 42
console.log(valD); // null
```

Cet opérateur renvoi la valeur de droite si la valeur de gauche a une valeur null ou undefined.
Si la valeur de gauche est indéfinie et la valeur de droite est égal à null la console renvoi la valeur null car l’opérateur renverra toujours la valeur de droite.
Dans le cas où la valeur est vide et que l’autre valeur est une chaîne de caractère, l’opérateur retournera la valeur vide car il n’y a pas de valeur null ou indéfini.

Chaînage optionnel

```
const adventurer = {
  name: 'Alice',
  cat: {
    name: 'Dinah'
  }
};

const dogName = adventurer.dog?.name;
console.log(dogName);
// expected output: undefined

console.log(adventurer.someNonExistentMethod?.());
// expected output: undefined
```

Cet opérateur permet de savoir si une valeur existe ou non et si elle est null ou undefined.