

1. Exo

Peut-on utiliser directement le code ***let currentTempC = 100;*** dans une expression ?

SOS On vérifie que ***let currentTempC = 100;*** ne retourne pas de valeur. Pour cela, il vous suffit de faire appel à la console de votre navigateur qui interprète votre code JS.

Regardez bien le retour de cette instruction¹.

```
let currentTempC = 100;
undefined
```

 Il faut d'abord déclarer la variable ***currentTempC*** pour pouvoir l'utiliser dans une expression.

COOL Il n'est pas possible d'utiliser directement ***let currentTempC = 100;***

2. Exo

Peut-on utiliser directement le code ***currentTempC*** dans une expression ?

SOS Une fois déclarée, on peut écrire une expression utilisant l'identifiant ***currentTempC***

```
currentTempC = 100;
100
```

COOL ***currentTempC*** est donc utilisable dans une expression.

3. Exo

Peut-on utiliser la valeur **42** dans une expression ?

SOS On vérifie :

```
currentTempC = 42;
42
```

currentTempC = 42 est une expression ; **42** est une valeur directement utilisable dans une expression.

COOL On peut utiliser 42 dans une expression.

¹ Une instruction ne retourne pas de valeur !

4. Exo

Voici deux affirmations

- 1) Il est possible d'écrire en JS un test de condition en utilisant une expression.
- 2) Il est possible d'écrire en JS un test de condition en utilisant une instruction.

Tentez de trouver le code correspondant à ces deux affirmations.

 On étudiera [l'opérateur ternaire conditionnel](#).

 Voici une écriture d'une condition avec une instruction

Instruction if

```
let mâle = true, salutation;

if ( mâle ) {
  salutation = 'Mr.';
} else {
  salutation = 'Mrs.';
}

> "Mr."
```

 Voici une écriture d'une condition avec une expression.

Expression if

```
let mâle = true, salutation;

salutation = ( mâle ? 'Mr.' : 'Mrs.' );

> "Mr."
```

 Les deux affirmations sont vrai !

5. exo

Une fonction a des arguments qui sont des valeurs.

Quelle écriture de l'exercice précédent sera utilisable comme argument d'une fonction ?

L'instruction if	L'expression if
<pre> let mâle = true ; function test(x){ console.log(x); return x; } test(instruction if); /*test(if (mâle) { salutation = 'Mr.'; } else { salutation = 'Mrs.'; })*/</pre>	<pre> let mâle = true ; function test(x){ console.log(x); return x; } test(expression);</pre>

 [Code](#)

 [Code](#)

Cas 1 Une instruction comme argument.

Soit l'instruction :

```

if (mâle ) {
    salutation = 'Mr.';
} else {
    salutation = 'Mrs.';
}
```

Passons cette instruction en argument dans la fonction **test**.

```
test( if (mâle ) {  
  salutation = 'Mr.';  
} else {  
  salutation = 'Mrs.';  
})
```

[En action](#)

Résultat :

"error"

"SyntaxError: Unexpected token if"

Cas 2 Une expression comme argument.

Soit l'expression :

salutation = (mâle ? 'Mr.' : 'Mrs.')

Passons cette expression en argument de la fonction **test**.

```
test(salutation = (mâle ? 'Mr.' : 'Mrs.'));
```

[En action](#)