

# 0 - Introduction

---

## Qu'est-ce qu'une fonction ?

---

Un tuyaux de mutation, de transformation

Un tuyaux qui prend des arguments typés et délivre un type en sortie

```
myFunc :: Int -> Int
convertStringToInt :: String -> Int
myPlusFunction :: Int, Int -> Int
myErrorSupportiveFunction :: Int -> Int, Error
```

## A quoi ça sert de tester ?

---

s'assurer de la bonne implémentation d'une fonction

donner un mode d'emploi d'une fonction ou d'un ensemble logique

faciliter la découverte de régressions dans le code

## Un exemple

---

Exprimer un besoin fonctionnel:

je dois calculer la somme de x et de y.

Du besoin fonctionnel, déterminer la structure d'une fonction add() correctement typée.

Rédiger un test pour vérifier l'implémentation de la fonction add().

```
function add(a, b) {  
  return a + b;  
}  
  
function addTest() {  
  let expected = 5;  
  let result = add(2, 3);  
  
  console.log(expected === result);  
}  
  
addTest();
```

## Point définition

---

simple

prédictive

sans effet de bord

avec idempotence

## Fonction impure

---

```
let count = 1;

function impureIncrementCount() {
    count += 1;
}

print(count);
impureIncrementCount();
print(count);
```

## Fonction pure

---

```
function pureIncrementCount(count) {
    return count += 1;
}

let count = 1;
print(count);
let incremented = pureIncrementCount(count);

// la variable count n'est pas modifiée
print(count);
print(incremented);
```

la fonction `impureIncrementCount` modifie une variable en dehors de son scope, présente dans le scope global.

la fonction `pureIncrementCount` renvoie une nouvelle variable qui devra être assignée pour être gardée.

la fonction impure fait un effet de bord sur la variable extérieure au scope.

## Créer un test pour la fonction increment

---

```
function pureIncrementCount(count) {  
    return count += 1;  
}  
  
function testPureIncrementCount() {  
    let testCount = 1;  
    let expectedCount = 2;  
  
    let testResult = pureIncrementCount(testCount);  
    console.log(testResult === expectedCount);  
}  
  
testPureIncrementCount();
```