

[Welcome home !](#)[Règlement intérieur](#)[Poster une issue](#)[JavaScript](#)[Environnement de dev](#)[👋 Semaine 1](#)[Terminal](#)[Variables](#)[Conditions](#)[Boucles](#)[👋 Semaine 2](#)[Manipuler des chaînes de caractères](#)[Fonctions](#)[🎥 Vidéos](#)

Introduction et concept

[Notations possibles](#)[Déclaration et Appel](#)[Ajouter des paramètres à vos fonctions](#)[Valeur de retour](#)[Tableaux](#)[👋 Semaine 3](#)[Déroulement d'un programme](#)[Les booléens](#)[Les valeurs truthy et falsy](#)[Notion de scope](#)[Conditions d'existence](#)[Conditions imbriquées](#)[👋 Semaine 4](#)[Objets](#)[Requêtes à API](#)

Introduction

Quand vous programmerez, vous serez souvent amenés à utiliser plusieurs fois la même série d'instructions dans un même programme.

C'est dans ces cas-là que vous aurez besoin d'une fonction. Attention, ce n'est pas comme une boucle, qui va se lancer, et répéter du code en boucle.

Non : le principe de la fonction, c'est de stocker un bout de code que vous voudrez exécuter à plusieurs reprises, et à des moments précis, par la suite.

Imaginons que vous êtes un pizzeria qui doit préparer des pizzas. Pour cela, vous devrez suivre une série d'étapes précises pour préparer chaque pizza. Nous pouvons donc définir des étapes pour la préparation de la pizza :

- Préchauffer le four à la température requise.
- Étaler la pâte à pizza.
- Ajouter la sauce tomate sur la pâte.
- Ajouter le fromage.
- Cuire la pizza.

Appelons cela une procédure : vous répéterez ces 5 instructions - dans l'ordre - chaque fois que vous devrez faire une pizza.

Concept

En JavaScript, c'est un peu pareil : vous aurez des procédures qu'il faudra exécuter à des moments précis. Ce sera un ensemble d'instructions pour accomplir une tâche spécifique. Elle pourra être utilisée chaque fois que vous devez effectuer cette tâche !