

[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](#)[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](#)[Backend & Strapi](#)[HTML / CSS](#)[Vue.js](#)[Après la formation](#)

Déroulement d'un programme

En JavaScript, le code est toujours lu du haut vers le bas. C'est ainsi que si vous écrivez ces lignes, la valeur "finale" (c'est à dire une fois tout le programme exécuté) assignée à la variable password est bien `"654321"` :

```
let password = "azerty"; // exécutée en 1
password = "123456"; // exécutée en 2
password = "654321"; // exécutée en 3
```

Mais si vous voulez modifier cette "ordre d'exécution", il existe des façons d'ordonner son code pour que certaines lignes (ou instructions) soient ignorées, exécutées uniquement sous certaines conditions, ou encore répétées plusieurs fois. Il est même possible, si vous utilisez une `fonction` ou une `boucle`, d'exécuter du code se trouvant bien plus "haut" dans votre script.