

# Premiers pas en JavaScript

## Sommaire:

- I. [Introduction](#)
- II. [Préparation du logiciel](#)
- III. [Programmation en java](#)
- IV. [Conclusion](#)

## I. Introduction

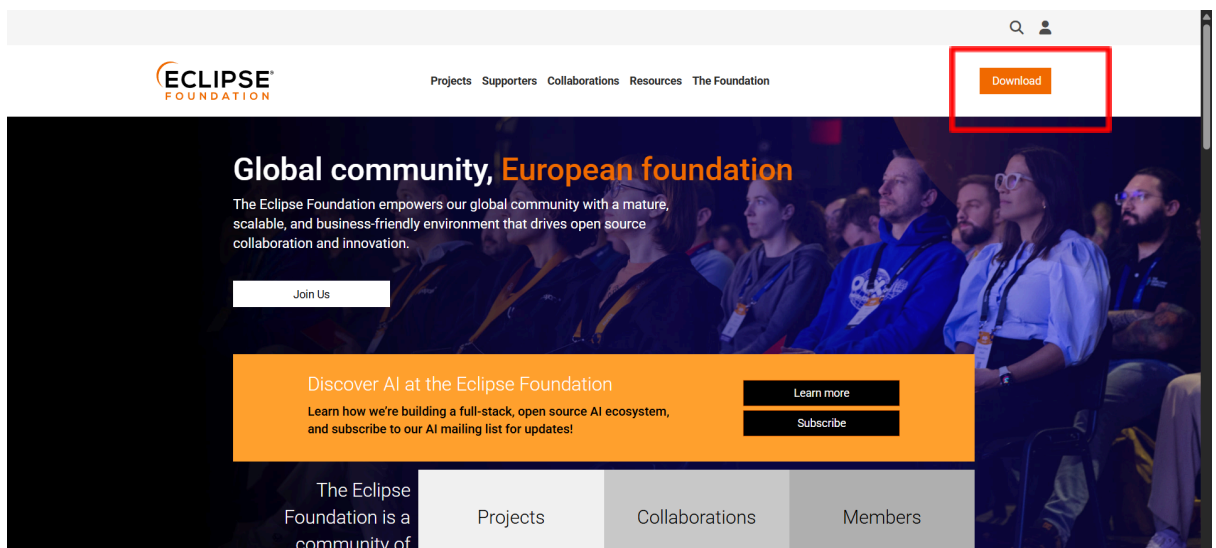
Le langage Java est aujourd'hui l'un des plus utilisés dans le développement logiciel, que ce soit pour des applications bureautiques, web ou mobiles. Avant de pouvoir programmer en Java, il est indispensable d'installer et de configurer un environnement adapté, comprenant le kit de développement (JDK) et un environnement de développement intégré (IDE) tel qu'Eclipse.

Ce premier TP a pour objectif de nous familiariser avec ces outils et d'acquérir les bases nécessaires pour créer et exécuter un programme Java simple. Il permet également de faire le lien entre la logique algorithmique vue en cours et sa traduction concrète dans le langage Java.


## II. Préparation du logiciel

Pour commencer le développement il faut que je télécharge un compilateur (il va permettre de traduire le code Java pour le système Windows de mon ordinateur portable) ainsi qu'une JVM ( Java Virtual Machine qui va permettre de lire et exécuter les programmes Java ). Pour cela nous allons télécharger le logiciel en passant par AdoptOpenJDK et en choisissant OpenJDK 8.

Puis nous allons sur le site [eclipse.org](https://eclipse.org), puis download,




puis on télécharge eclipse:



ProjectsSupportersCollaborationsResourcesThe Foundation


Home » Downloads » Eclipse downloads - Select a mirror

All downloads are provided under the terms and conditions of the [Eclipse Foundation Software User Agreement](#) unless otherwise specified.



Download from: Netherlands - LiteServer (https)  
File: [eclipse-inst-jre-win64.exe](#) | [SHA-512](#)  
[» Select Another Mirror](#)

Sponsored Ad



An Open-Source Strategy

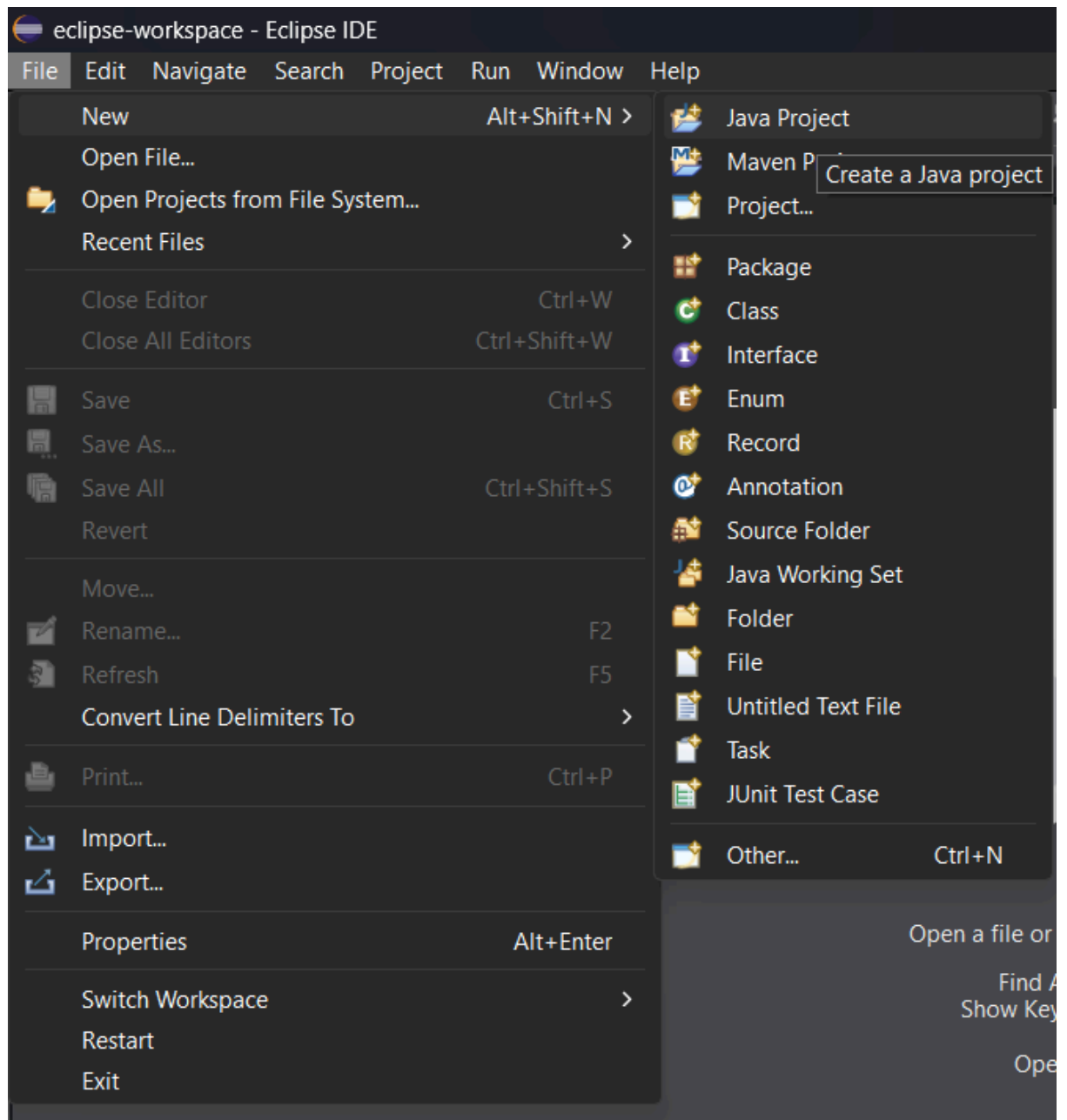
[Advertise Here](#)

Other options for this file

- [All mirrors \(xml\)](#)
- [Direct link to file](#)  
(download starts immediately from best mirror)

### III. Programmation en Java

Pour commencer le programme je crée un projet:

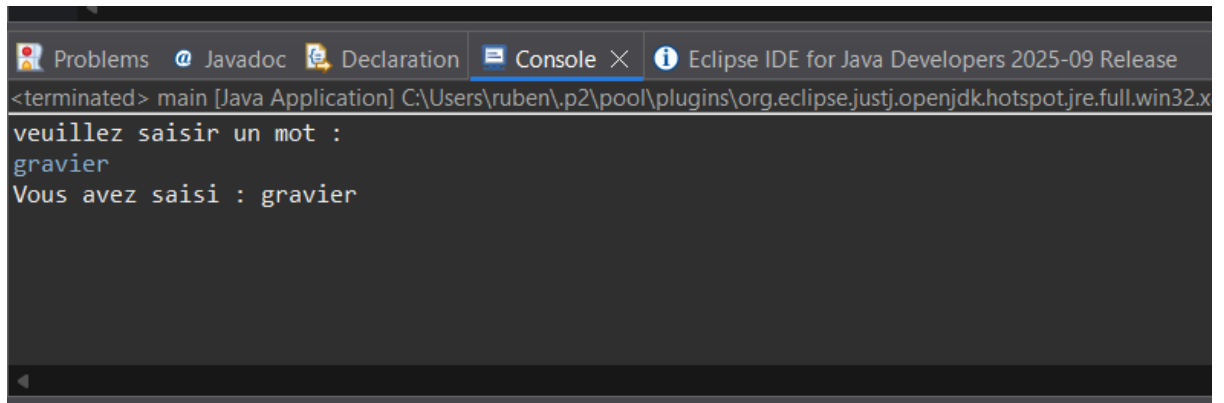


Puis je passe à l'algorithme. Un algorithme très simple en java qui permet d'utiliser scanner pour récupérer le mot inscrit par l'utilisateur et le réafficher:

```

1 package premiers_pas_en_java;
2 // Déclare le package dans lequel se trouve la classe
3
4 // Importation de la bibliothèque Scanner pour lire les entrées clavier
5 import java.util.Scanner;
6
7 public class main { // Déclaration de la classe principale
8
9     public static void main(String[] args) { // Méthode principale exécutée au lancement du programme
10
11         // Création d'un objet Scanner pour lire les entrées utilisateur depuis le clavier
12         Scanner sc = new Scanner(System.in);
13
14         // Affiche un message demandant à l'utilisateur de saisir un mot
15         System.out.println("veuillez saisir un mot :");
16
17         // Lecture de la ligne saisie par l'utilisateur et stockage dans la variable str
18         String str = sc.nextLine();
19
20         // Affiche le texte saisi par l'utilisateur
21         System.out.println("Vous avez saisi : " + str);
22     }
23 }

```



```
<terminated> main [Java Application] C:\Users\ruben\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x
veuillez saisir un mot :
gravier
Vous avez saisi : gravier
```

On constate que le programme renvoie bien le mot écrit on peut donc dire qu'il fonctionne.

## IV. Conclusion

Ce TP était très simple. Cependant bien pratique pour se rappeler des bases du JavaScript notamment l'étape du compilateur car si l'on sort des programmes plus simples style python on peut passer à côté de cette étape. Donc je dirai qu'il n'était pas difficile du tout mais nécessaire pour reprendre le java.