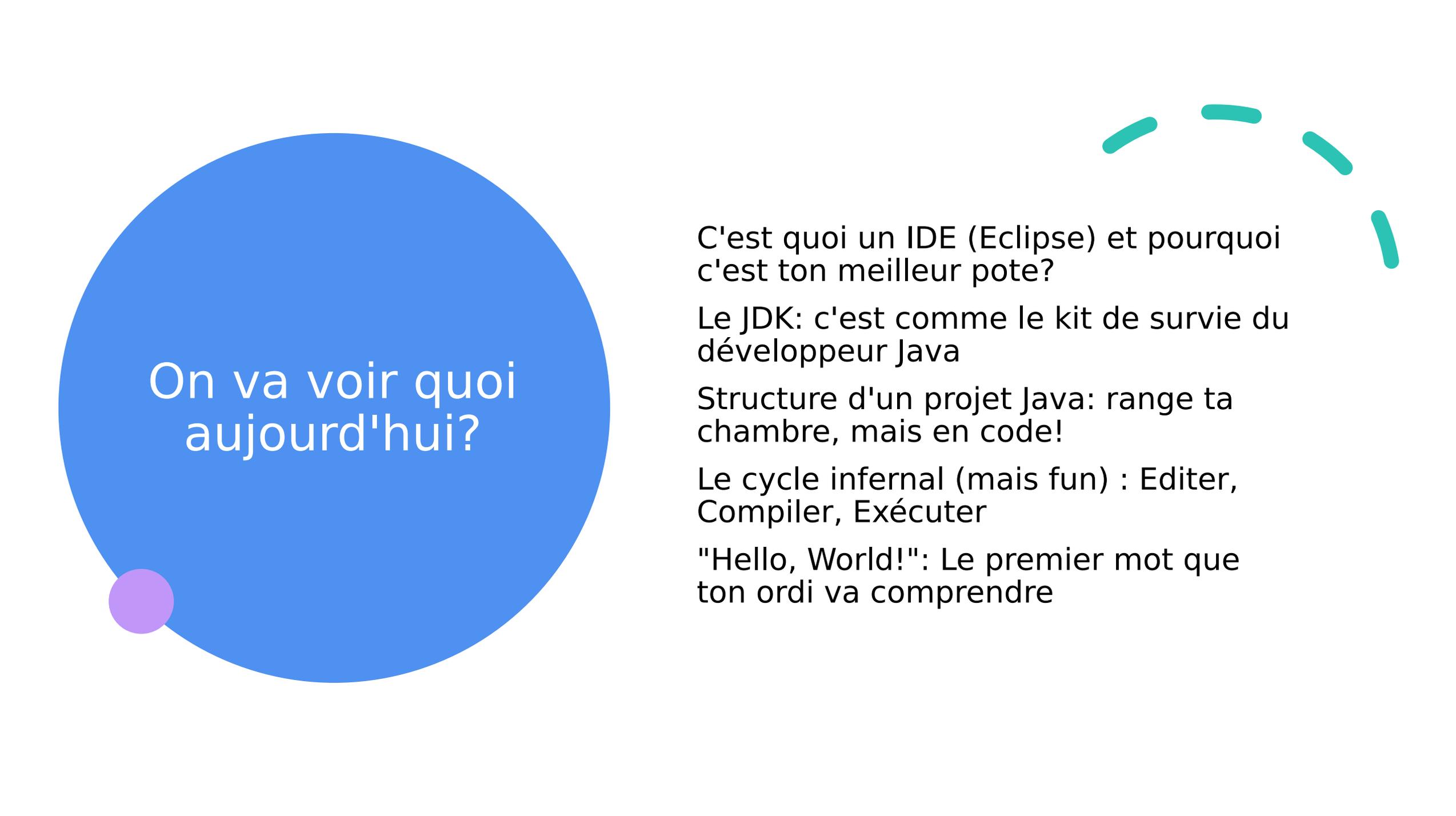




Java & Eclipse: Ton premier pas
dans le code!



On va voir quoi aujourd'hui?

C'est quoi un IDE (Eclipse) et pourquoi c'est ton meilleur pote?

Le JDK: c'est comme le kit de survie du développeur Java

Structure d'un projet Java: range ta chambre, mais en code!

Le cycle infernal (mais fun) : Editer, Compiler, Exécuter

"Hello, World!": Le premier mot que ton ordi va comprendre

IDE: Le QG du développeur!

- IDE = Environnement de Développement Intégré
- C'est une boîte à outils qui te facilite la vie pour coder:
 - Éditeur de code: ça colore ton code pour que tu t'y retrouves
 - Gestion de projet: ça range tous tes fichiers comme un pro
 - Compilateur intégré: ça transforme ton code en langage machine (presque de la magie!)
 - Débogueur: ça t'aide à trouver les erreurs (les fameux bugs!)



[Image Source](#)

Le JDK: Ton kit de survie Java

- JDK = Java Development Kit
- Il contient tout ce qu'il faut pour développer en Java:
 - Le JRE: c'est lui qui fait tourner tes programmes Java
 - Des outils: pour compiler (javac), créer des archives (jar), etc.
- Pense aux variables d'environnement : JAVA_HOME et PATH, c'est important !



[Image Source](#)

Structure d'un projet Java: Rangement niveau expert

- Un projet Java, c'est comme une maison bien rangée:
 - `src/main/java/`: Là où tu mets ton code (le plus important!)
 - `src/main/resources/`: Les images, les fichiers de config, etc.
 - `bin/`: Le code compilé (pas touche!)
 - `lib/`: Les bibliothèques externes (un peu comme des plugins)

Le cycle de vie du code (en mode Fast & Furious)

- 1. Écriture du code: Tu tapes ton code dans l'éditeur
- 2. Compilation: Le code est transformé en langage machine
- 3. Exécution: Ton programme se lance et fait des trucs cool!
- 4. Débogage: Si ça marche pas, tu corriges les erreurs (les bugs)



[Image Source](#)

"Hello, World!": Le B.A.-BA du codeur

- Le code minimal pour afficher un message à l'écran:
 - ```
public class HelloWorld { public static void main(String[] args)
{ System.out.println("Hello, World!"); } }
```
- C'est le premier pas vers la gloire (ou au moins, un programme qui marche!)

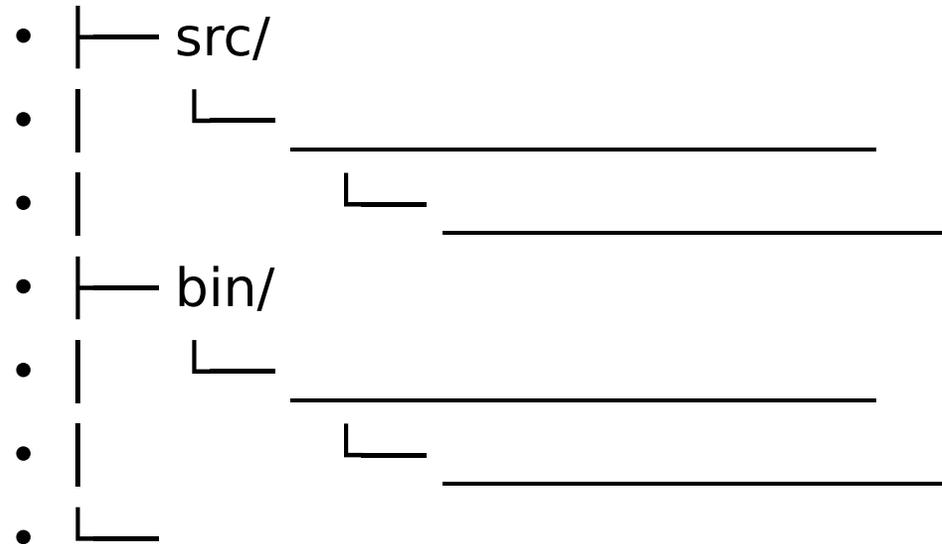
# À toi de jouer! - Exercice 1: Remise en Ordre

- Mets les étapes suivantes dans le bon ordre pour créer un projet Java fonctionnel :
  - a) Télécharger Eclipse
  - b) Installer Java JDK
  - c) Créer un projet Java
  - d) Lancer Eclipse
  - e) Télécharger Java JDK
  - f) Installer Eclipse

# À toi de jouer! - Exercice 2: Complète le Schéma

- Complète le schéma suivant avec les termes appropriés : src, bin, monpackage, MaClasse.java, main, System.out.println("Hello, World!");

- MonProjet/



# À toi de jouer! - Exercice 3: Analyse du Code 'HelloWorld'

- Réponds aux questions suivantes concernant le code 'HelloWorld' :
  - 1. Que signifie 'public class HelloWorld' ?
  - 2. Quel est le rôle de 'main' ?
  - 3. À quoi sert 'System.out.println' ?
  - 4. Où se trouve le fichier dans le projet ?

# À toi de jouer! - Exercice 4: Simulation de Projet

- Décris chaque étape pour créer un projet Java simple :
  - 1. Créer un projet Java.
  - 2. Créer un package 'monappli'.
  - 3. Ajouter une classe 'Test'.
  - 4. Écrire la méthode 'main' qui affiche 'Coucou!'.

# À toi de jouer! - Exercice 5: Quiz Récap'

- Réponds à ces questions pour tester tes connaissances :
  - 1. Que contient le dossier 'src' ?
  - 2. Où sont stockés les fichiers '.class' ?
  - 3. Qu'est-ce qu'un package en Java ?
  - 4. Quel est le rôle de 'System.out.println' ?
  - 5. Quelle est la différence entre une classe et un objet ?
  - 6. Quelles sont les règles de nommage des fichiers '.java' ?

# Besoin d'aide? Ressources utiles

- La doc Java (Oracle): Pour les warriors du code:  
<https://docs.oracle.com/javase/tutorial/>
- Tutoriel Eclipse (Vogella): Le guide du débutant:  
<https://www.vogella.com/tutorials/Eclipse/article.html>
- OpenClassrooms: Apprends Java à ton rythme:  
<https://openclassrooms.com/fr/courses/26832-apprenez-a-programmer-en-java>
- Codingame/HackerRank: Deviens un pro en t'amusant!