

Présentation des outils

IDE (Integrated Development Environment)

- Pendant la première moitié du cours, nous allons programmer avec Visual Studio Code (VSCode).
 - Léger
 - Facile à apprendre
 - Idéal pour des petits projets
- Puis, plus tard dans la session, nous allons utiliser Visual Studio
 - Lourd, prend plus de temps à ouvrir
 - Contient beaucoup plus de fonctionnalité intégré par défaut
 - Idéal pour des gros projets

Créer un nouvel algorithme en c#

- Lorsque nous allons écrire un algorithme en c#, cela se fera sous la forme d'un programme console.
- La commande `dotnet new console` permet d'en créer un à l'emplacement où nous sommes.
 - Sur windows, vous pouvez écrire la commande directement dans la barre de navigation de l'explorateur Windows.
 - Alternativement, vous pouvez écrire `cmd` pour ouvrir une console et ensuite écrire la commande dans la console.
- Un dossier ayant le nom que vous avez choisi sera alors créé.

Ouvrir votre programme console avec VSCode

- Ouvrez ensuite ce dossier avec VSCode. Il y a plusieurs possibilités:
 - Ouvrez VSCode, puis sélectionner le menu File > Open Folder, sélectionner le dossier précédemment créé.
 - Faites un click droit sur le dossier précédemment créé et choisissez: Ouvrir avec code
 - Si vous avez déjà une console d'ouverte écrivez `code .` si vous êtes à l'intérieur du dossier.
- Lorsque le dossier est correctement ouvert, vous devriez voir, à gauche, le nom de l'algorithme comme élément racine de l'explorateur.

Git, GitHub et GitHub Desktop

Qu'est-ce que Git ?

- Système de contrôle de version distribué
- Suivi des modifications de fichiers
- Collaboration sur des projets de développement de logiciels

Pourquoi utiliser Git ?

- Gestion efficace des versions
- Collaboration facilitée entre les développeurs
- Suivi précis des modifications et des contributeurs
- Très utilisé sur le marché du travail.

Qu'est-ce que GitHub ?

- Plateforme de gestion de code source basée sur Git
- Hébergement de projets et de dépôts de code
- Outils pour la collaboration, le suivi des problèmes et plus encore

Pourquoi utiliser GitHub ?

- Centralisation du code source (toujours accessible)
- Facilité de collaboration avec une communauté mondiale
- Intégration avec une variété d'outils de développement

GitHub Desktop

- Interface graphique pour Git
- Simplifie les opérations Git courantes
- Convivial pour les débutants

Comment commencer avec Git, GitHub et GitHub Desktop ?

1. Installer Git sur votre système
2. Créer un compte sur GitHub
3. Installer GitHub Desktop
4. Cloner un dépôt existant ou créer un nouveau dépôt