amU Faculté des sciences Aix Marseille Université

Année 2025-26

Domaine Sciences et Technologies
Licence Informatique : math-info
Initiation au Génie logiciel : TP 9
Codes UE : SIN5U34
Application graphique pour gérer un annuair

ATTENTION : le sujet n'est pas définitif ! Si vous avez fini les TP de 1 à 8, vous pouvez commencer le TP 10.

Annuaire

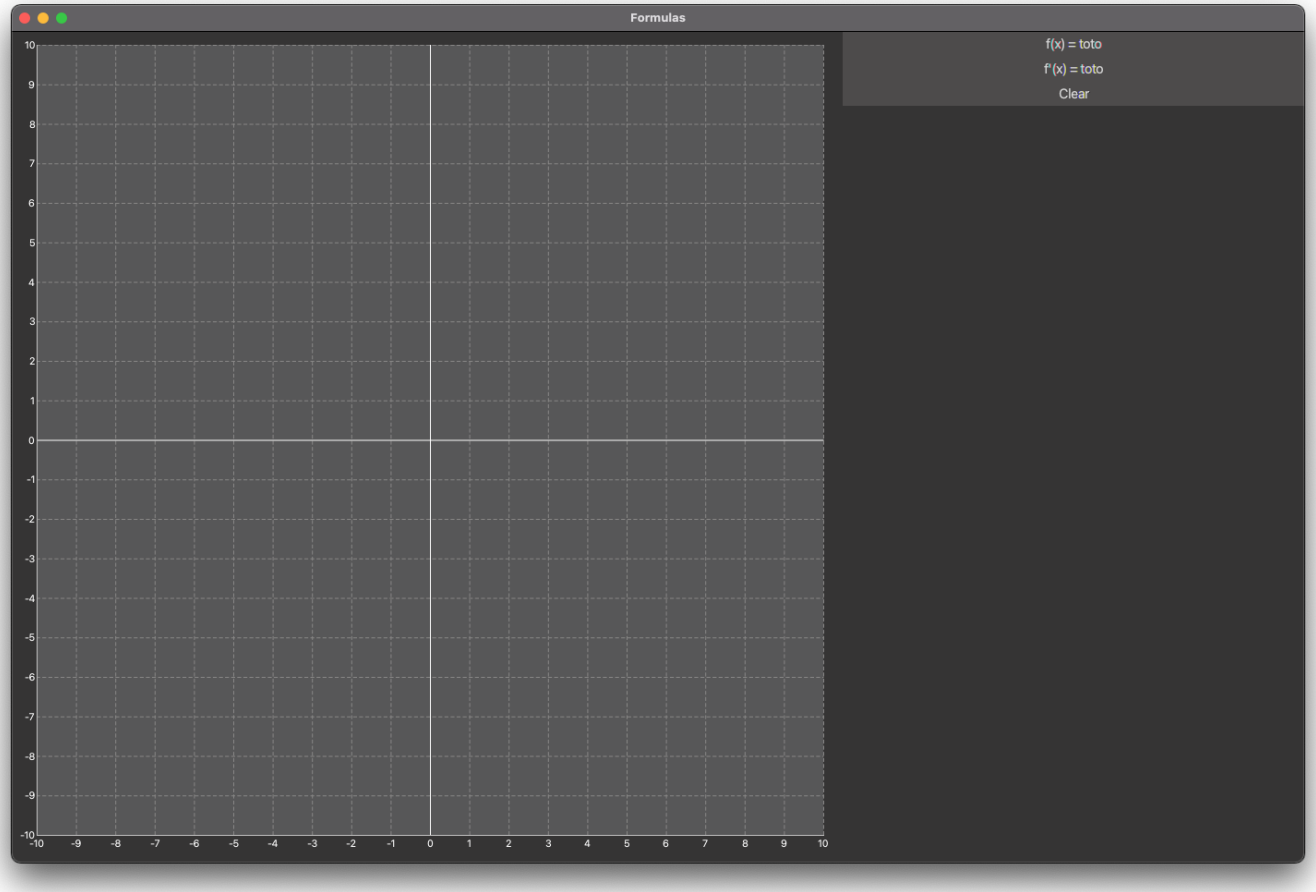
Ce tutoriel pour apprendre à utiliser `javafx` pour faire une interface graphique en python est basé sur le tutoriel suivant : JavaFX Tutorial et sa traduction en Français

Ce tutoriel vous guide pour les phases de conception, de programmation et de déploiement d'une application de répertoire d'adresse. Voici à quoi ressemblera l'application :

Consignes pour démarrer le TP

Comme pour les TP précédents, on va utiliser git pour la gestion de versions. Il vous faut donc vous reporter aux consignes du premier TP.

Une fois le dépôt téléchargé, vous pourrez compiler et exécuter le programme en cliquant deux fois sur `adressapp-***` -> `application` -> `run` (en n'oubliant pas d'ajouter votre projet en tant que projet *Gradle*).



Lien vers la classroom du TP : lien.

Ce que vous allez apprendre

- Créer et lancer un projet JavaFX
- utiliser le format `fxml` pour
- Structurer une application suivant le schéma Modèle Vue Contrôleur (MVC)
- Utiliser des `ObservableLists` pour mettre à jour automatiquement l'interface utilisateur
- Utiliser une `TableView` et réagir au changements de sélection dans la table
- Créer une fenêtre de dialogue popup pour éditer les données de personnes
- Valider les entrées utilisateur
- Personnaliser votre application JavaFX avec CSS
- Gérer des données persistantes en XML
- Créer un graphique JavaFX pour afficher des statistiques

C'est déjà beaucoup! Donc, après être allé au bout de cette série de tutoriaux, vous devriez être prêt à coder des applications sophistiquées avec JavaFX.

Tâche 1 : première vue

Création de paquets

Dès maintenant nous allons suivre les bonnes pratiques de programmation. Un principe très important est le Model-View-Controller (MVC). En accord avec ceci, nous divisons notre code en trois parties. Vous allez donc créer deux paquets et un répertoire :

— un package `model` (dans `src/`) qui contiendra les classes modèles,
— un package `controller` qui contiendra les classes contrôleurs,
— un répertoire `view` dans le dossier ressource qui contiendra des fichiers `fxml` décrivant les vues.