

TD n°1

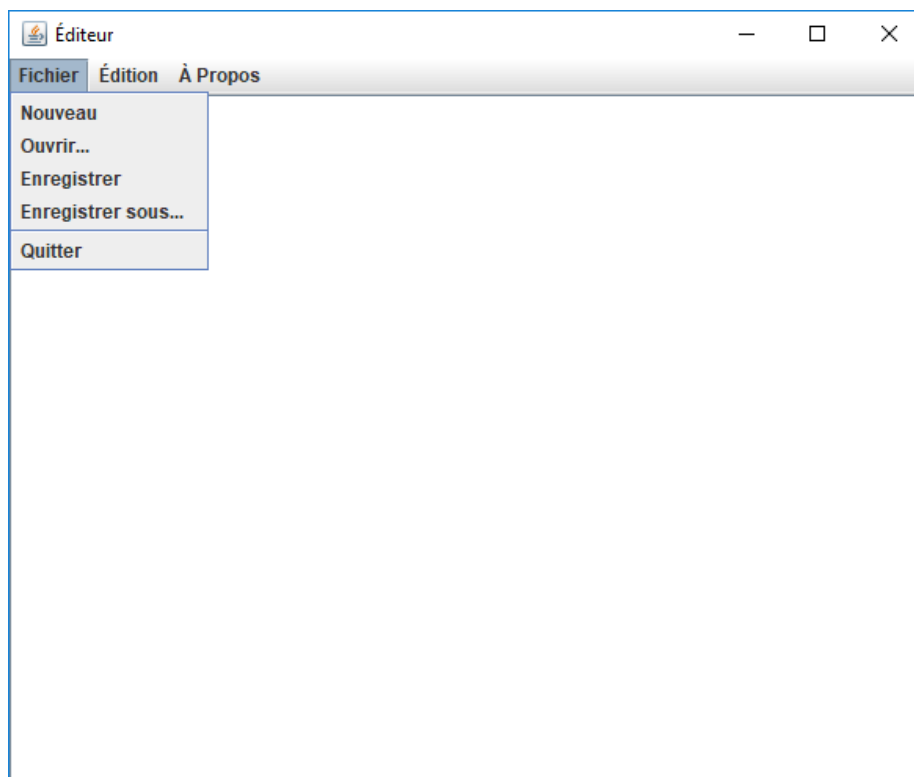
Éditeur de texte simplifié

Créer un éditeur de texte simplifié permettant d'effectuer les actions les plus courantes du traitement de texte :

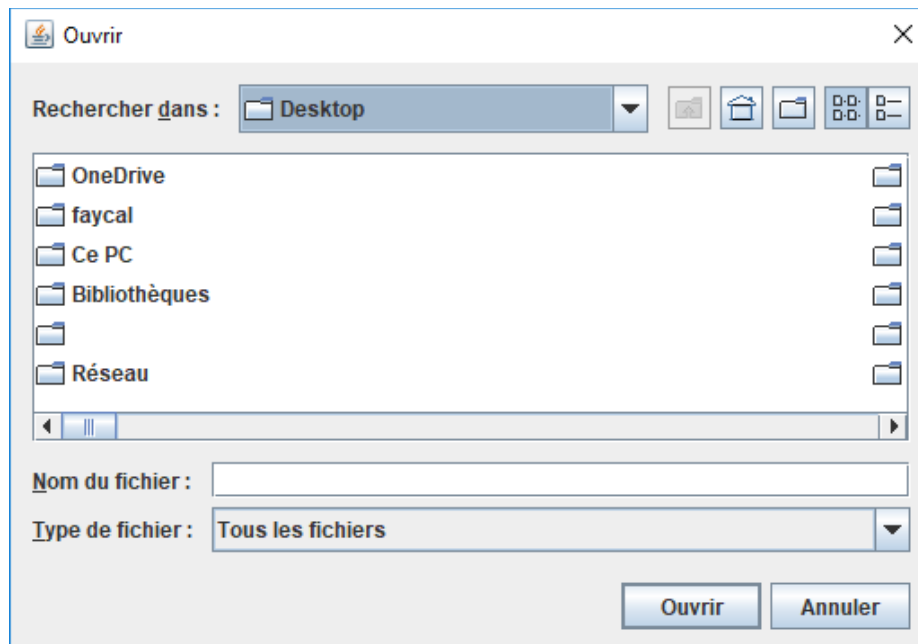
- Créer/ouvrir un fichier
- Écrire dans un fichier
- Copier/Couper/sélectionner/Coller dans un fichier
- Enregistrer un fichier

Cahier des Charges :

1. Le Menu Fichier



- a. La sous-rubrique « Nouveau »
Elle permet d'écrire de nettoyer l'écran afin de la rendre propre à l'utilisation.
N.B : l'enregistrement du contenu tapé n'interviendra que si l'utilisateur utilise les sous-rubriques « Enregistrer » ou « Enregistrer sous ».
- b. La sous-rubrique « Ouvrir »
Elle permet de lancer un composant de type «JFileChooser »



Le résultat du choix d'ouvrir un fichier ou d'annuler l'opération sera traité par la méthode *showOpenDialog* du composant.

L'affichage du fichier sera traité par le biais d'une lecture bufferisée d'un fichier (vue l'année dernière) dont le contenu sera inséré dans la zone de texte.

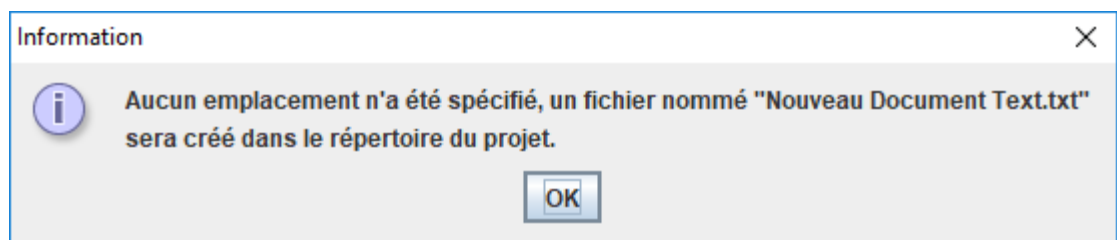
Il faudra, pour ce faire, définir une méthode privée de lecture de fichier (`private void lireFichier(File)`).

c. La sous-rubrique « Enregistrer »

Comme son nom l'indique, elle permet d'enregistrer les fichiers nouvellement créés ou modifiés.

Dans le 1^{er} cas, le nom de fichier n'est pas connu, l'enregistrement se fait dans le répertoire par défaut (dans netbeans, le répertoire racine du projet) sous le nom « Nouveau Document.txt ».

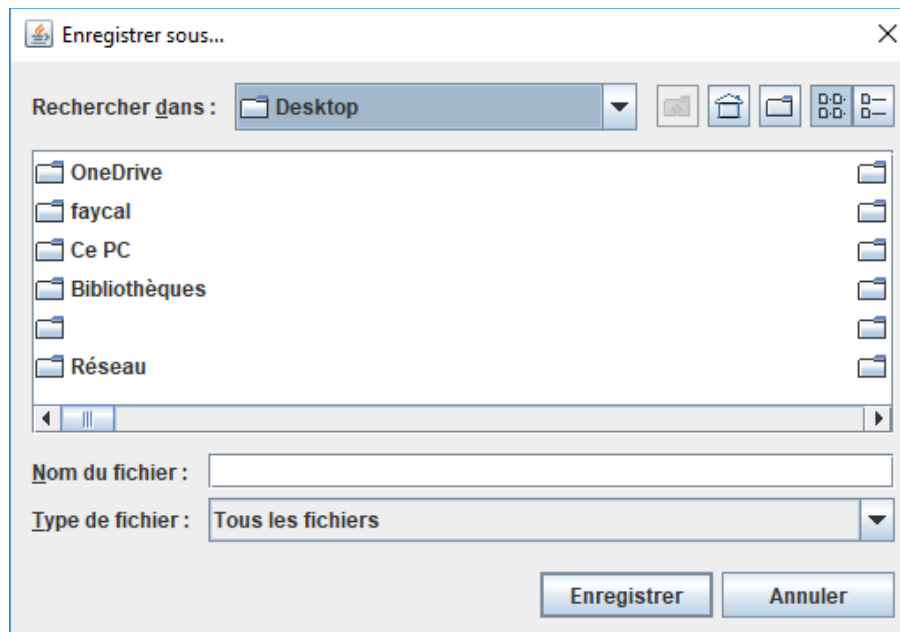
Il faudra en avvertir l'utilisateur par un message informatif de type `JOptionPane`



Dans le 2nd cas, il s'agira d'écrire le contenu modifié de la zone de texte dans le fichier en question. Il faudra, pour ce faire, définir une méthode privée d'écriture dans un fichier (`private void ecrireFichier(File)`) qui gèrera les exceptions afférentes à ce cas.

d. La sous-rubrique « Enregistrer sous »

Elle permet de lancer un composant de type « `JFileChooser` » s'intitulant « Enregistrer sous... ».

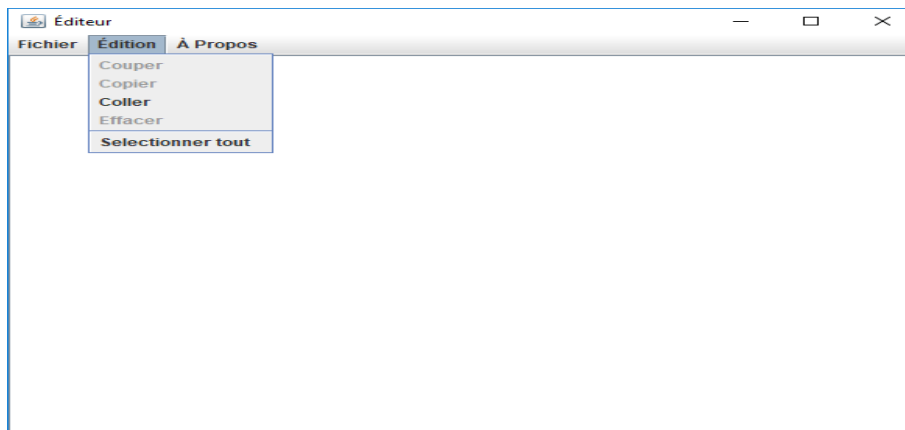


Si le bouton « Enregistrer » est pressé, la méthode d'écriture de fichier est invoquée avec pour paramètre le nom du fichier saisi par l'utilisateur.

e. La sous-rubrique « Quitter »

Elle permet de quitter l'éditeur à l'aide de la méthode `dispose()`.
à noter le séparateur de la sous-rubrique, il est obtenu en utilisant la méthode `addSeparator()` du composant `JMenu` juste avant la création de la sous-rubrique que l'on souhaite séparer.

2. Le menu « Édition »



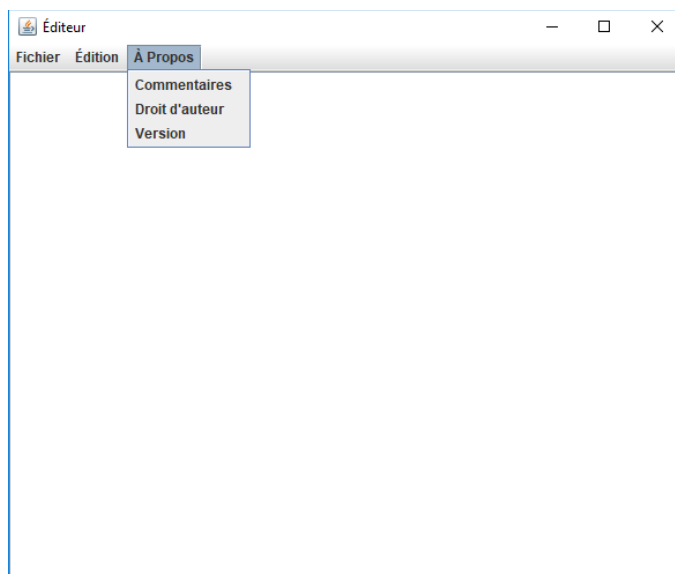
Au lancement, s'il n'y a pas de texte sélectionné les sous-rubriques « Couper », « Copier » et « Effacer » sont inactives. Elles deviennent actives lorsque du texte est sélectionné et pour ce faire il faudra gérer un événement de type clic-souris sur le menu « Édition » (interface `MouseListener` à implémenter).

a. La sous-rubrique « Couper »

Elle permet de couper le texte sélectionné dans une zone de texte.

- b. La sous-rubrique « Copier »
Elle permet de copier le texte sélectionné dans une zone de texte.
- c. La sous-rubrique « Coller »
Elle permet de coller du texte dans une zone de texte.
- d. La sous-rubrique « Effacer »
Elle permet d'effacer le texte sélectionné dans une zone de texte.
Il s'agira en fait du remplacement du texte sélectionné par « »
- e. La sous-rubrique « Sélectionner tout »
Elle permet de sélectionner l'ensemble du contenu d'une zone de texte.

3. Le menu « à propos »



- a. La sous-rubrique « Commentaires »
Elle permet de lancer une popup de type information indiquant « Éditeur de texte java rudimentaire ».
Pour lancer une popup, il vous faudra utiliser la méthode `JOptionPane.showMessageDialog(Component parentComponent, Object message, String title, int messageType)`.
- b. La sous-rubrique « Droit d'auteur »
Elle permet de lancer une popup de type information indiquant "auteur: FB \ndate : 05/11/2017".
- c. La sous-rubrique « Version »
Elle permet de lancer une popup de type information indiquant "Version 1.0".

