LE LENGTH CHOIX, L'EXCELLER.

PROMO 15-16, Année académique: 2017-2018

Licence 3 en Informatique (GL et RT) INF 3531 : Programmation Orientée Objet 2

Responsables CM: Pr THIAM - TD/TP: M. DIOP

Unité de Formation et de Recherche en Sciences Et Technologies Département Informatique

Fiche n° 4: Interfaces graphiques

On vous demande de réaliser une application pour compter (des étoiles, des passagers embarqués, des bactéries, etc.). Cela se présente (voir la figure 1) comme un panneau comportant un titre, un nombre entier et un bouton. Chaque fois que l'utilisateur appuie sur le bouton, le nombre augmente de une unité.

Exercice TPE:
Compteurs

A. Pour commencer, réalisez un programme très minimaliste : une classe Compteur avec la méthode main et deux variables statiques (la valeur du nombre et le JLabel chargé de son affichage).

Il vous faudra aussi une classe auxiliaire **AuditeurBouton** pour représenter l'objet qui détecte et dispatche les pressions sur le bouton ; faitesen une classe interne à **Compteur**, ce qui lui permettra d'accéder aux variables **valeur** et **affichage** (Pour des raisons techniques - mais compréhensibles - cette classe devra elle aussi être qualifiée **static**).



Fig. 1

- B. [Légère amélioration du code] Remplacez la classe interne AuditeurBouton par une classe anonyme.
- C. [Légère amélioration de l'aspect] Faites en sorte que le bouton « ++ » ait sa largeur préférée (voyez la figure 2), au lieu d'occuper toute la largeur du cadre. Pour cela, intercalez un panneau entre le bouton et le cadre.
- D. [Grosse amélioration du code] Faites les choses comme il faut les faire : réorganisez le code précédent afin de définir une classe Compteur, sous-classe de JPanel. Elle est munie d'un constructeur prenant le titre pour argument, et chacune de ses instances représente un panneau supportant un triplet (titre, nombre affiché, bouton).



Fig. 2

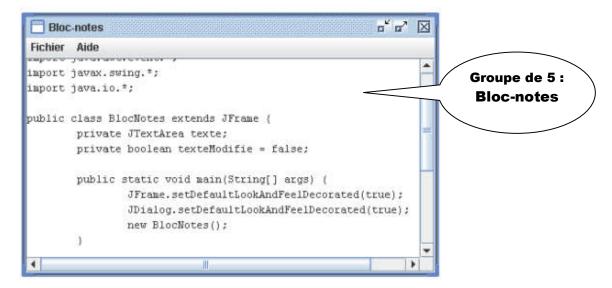
Pour essayer cette classe, écrivez une méthode **main** (soit dans la classe **Compteur**, soit dans une autre classe définie à cet effet) qui crée un cadre et y place un compteur.

E. Pour vous convaincre du bien fondé de la classe précédente, modifiez la méthode main précédente afin qu'elle crée un cadre avec, par exemple, quatre compteurs indépendants :



Fig. 3

On vous demande d'écrire un éditeur de texte simple, analogue au bloc-notes de Windows :

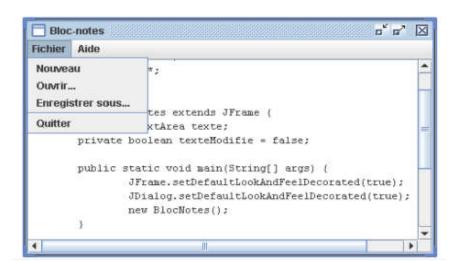


Fondamentalement, le bloc notes est un cadre (classe **JFrame**) avec une barre de menus (classe **JMenuBar**) et un panneau de contenu dans lequel on aura mis un panneau de défilement (classe **JScrollPane**) contenant une zone de texte (classe **JTextArea**).

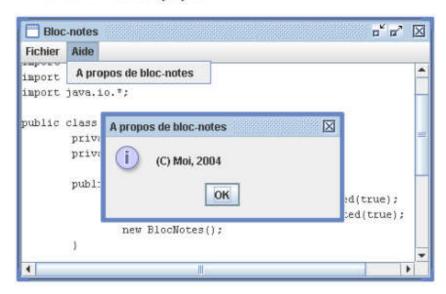
On veillera à ce que toute commande produisant la perte du texte édité soit précédée d'une confirmation :



Le menu Fichier est formé des commandes les plus classiques :

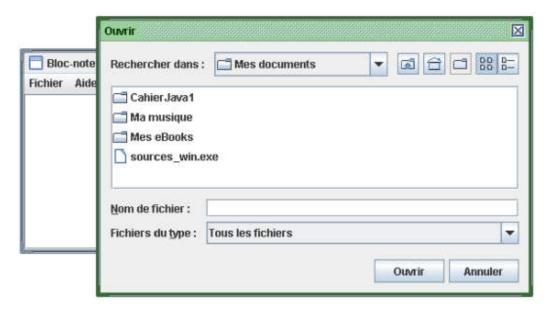


Le menu Aide se réduit à la boîte à propos :



Note 1. Pour déterminer s'il y a lieu de demander une confirmation avant de détruire le texte définissez une variable d'instance **texteModifié** que certaines opérations mettent à **false**, et qui est mise à **true** chaque fois qu'un caractère est tapé au clavier en visant la zone de texte (événement **Key**).

Note 2. La sauvegarde du texte dans un fichier ne pose pas de problème : il suffit d'écrire tout le contenu de la zone de texte dans un **FileWriter** ouvert à partir d'un objet **File** obtenu à l'aide d'un dialogue **FileChooser** :



La lecture du texte est un peu plus alambiquée, car les objets **FileReader** ne peuvent lire que dans un tampon de caractères, qu'il faudra allouer à partir de la taille donnée par une expression de la forme unFichier.length() (où unFichier est un objet **File** obtenu à l'aide d'un objet **FileChooser**).