

## Module 1 – Informatique – Algorithmique et Programmation Objet

### Travaux Pratiques (15), Licence 1ère Année

### Swing

#### Exercice 1

Écrire la classe **FenetreLitEntier** qui :

- affiche une fenêtre contenant le texte "**Entrez un nombre entier**" (un composant de type **JLabel**)
- affiche un champ textuel (un composant de type **JTextField**), qui permet de saisir un texte au clavier.

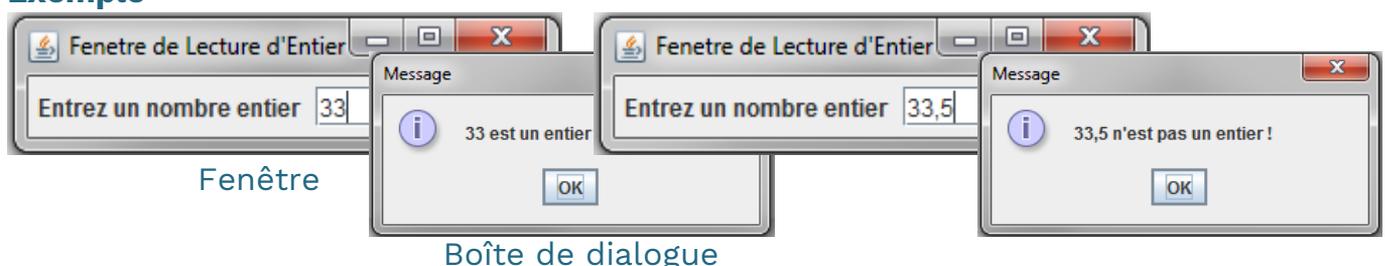
Dès qu'un texte a été rentré dans le champ textuel, la fenêtre doit se fermer puis une boîte de dialogue affichant l'entier doit être lancée.

Voici comment fonctionne un composant de type **JTextField** : lorsque la touche "entrée" du clavier valide le texte rentré, un événement de type **ActionEvent** parvient au composant. On peut récupérer le texte rentré grâce à la méthode **String getText()** définie dans la classe **TextField**. Un des constructeurs de **JTextField** prend en paramètre la taille du champ en nombre de colonnes.

Par ailleurs, la méthode de classe **int parseInt(String)** de la classe **Integer** retourne un entier à partir de sa forme textuelle.

La méthode **void dispose()** définie dans la classe **JFrame** permet de fermer une fenêtre (sans terminer le programme). On peut lancer une boîte de dialogue grâce à l'instruction **JOptionPane.showMessageDialog(null, s)**; où **s** est le texte (de type **String**) à afficher.

#### Exemple



#### Exercice 2

Écrire une application graphique en Swing qui affiche une fenêtre contenant deux boutons (**JButton**). Le premier bouton intitulé "**Encore ?**" lance et affiche une autre fenêtre identique lorsqu'on clique dessus (sans effacer la fenêtre précédente). Le second bouton intitulé "**Stop ?**" ferme la fenêtre contenant ce bouton (et pas les autres fenêtres) quand on clique dessus.

#### Exemple



### Exercice 3

Un carré magique est une grille carrée de  $n \times n$  cases ( $n$  impair) remplie des entiers de  $1$  à  $n^2$  telle que la somme des entiers de toute ligne, de toute colonne et de toute diagonale est égale à  $\frac{n(n^2+1)}{2}$ .

Il vous est demandé de réaliser une application graphique en Swing qui permet à un utilisateur de rentrer des entiers dans une grille, qui met à jour et affiche, au fur et à mesure des entrées, les sommes des lignes et des colonnes, ainsi que celles de la diagonale et de l'anti diagonale.

#### Exemple

Carré en cours de remplissage

0	0	0	0	3	3
4	0	0	0	0	4
0	5	0	0	0	5
0	0	1	0	0	1
0	6	0	2	0	8
4	11	1	2	3	0
3					

Carré rempli

11	24	7	20	3	65
24	12	25	8	16	85
17	5	13	21	9	65
10	18	1	14	22	65
23	6	19	2	15	65
85	65	65	65	65	65
65					65

Sommes des lignes (indicated by a red bracket on the right side of the grid)  
 Somme de l'anti diagonale (indicated by a blue arrow pointing to the bottom-left cell)  
 Sommes des colonnes (indicated by a red arrow pointing to the bottom row)  
 Somme de la diagonale (indicated by an orange arrow pointing to the bottom-right cell)

Chaque case sera définie par un composant de type **TextField**. Les sommes seront stockées dans des composants de type **JLabel**. Les champs textuels et les sommes seront disposés dans une grille de dimension  $(n + 2) \times (n + 1)$ . En dernière ligne, il n'y a que la somme de l'anti diagonale.

C'est le **Layout** de type **GridLayout** qui permet le placement automatique de composants dans une grille (grâce à la méthode habituelle **void add(Component)** de la classe **Container**). Le constructeur de **GridLayout** prend le nombre de lignes et de colonnes en paramètres.

Avant de créer la grille, le programme demandera la longueur du côté du carré grâce à une fenêtre similaire à celle de type **FenetreLitEntier** définie dans la question 1, qui elle créera la grille si le nombre introduit est impair.