

Licence MASS – INF F3 – TP n° 1

2013–2014

Le but de ce TP est de se familiariser avec la création d’interfaces graphiques et la manipulation d’informations. L’esthétique n’étant pas une priorité pour ce premier TP, les composants peuvent être positionnés dans la fenêtre à l’aide d’un layout `null`.

1 Échanger des informations grâce aux composants

1.1 Le champ de texte (`JTextField`)

Le champ de texte est similaire au label en ce sens qu’on peut écrire ou lire du texte grâce à ce composant. La différence étant que l’utilisateur peut aussi écrire dans le champ de texte.

Créez une fenêtre comportant un champ de texte, un label et un bouton. La phrase initiale affichée par le label doit être « Pas de données ». À chaque clic sur le bouton, le texte présent dans le champ de texte doit être lu puis écrit dans le label. Le champ de texte doit ensuite être réinitialisé (il n’affiche plus de texte).

1.2 La boîte de dialogue

La boîte de dialogue est une fenêtre temporaire qui permet d’afficher ou de demander des informations à l’utilisateur. Il est donc possible de remplacer le label ou le champ de texte par une boîte de dialogue.

Pour afficher une information, la fonction `void JOptionPane.showMessageDialog(..)` crée la fenêtre nécessaire à l’affichage de la boîte de dialogue. Cette fonction prend en paramètre :

- La fenêtre parente qui va être mise en pause tant que la boîte de dialogue est affichée ;
- le texte à afficher dans la boîte de dialogue ;
- le titre de la fenêtre nouvellement créée ;
- le type de message (qui va déterminer l’icône par défaut) qui peut être :
 - `JOptionPane.WARNING_MESSAGE` ;
 - `JOptionPane.ERROR_MESSAGE` ;
 - `JOptionPane.PLAIN_MESSAGE` ;
 - `JOptionPane.INFORMATION_MESSAGE` ;
- une image pour remplacer l’icône par défaut.

Les trois derniers paramètres sont optionnels et ont pour valeur par défaut `"Message"`, `JOptionPane.INFORMATION_MESSAGE` et `null`.

Utilisez cette fonction pour remplacer le label en faisant en sorte qu’une boîte de dialogue contenant la valeur du champ de texte s’affiche à chaque clic sur le bouton.

Pour demander une information, la fonction `String JOptionPane.showInputDialog(..)` crée la fenêtre nécessaire à l’affichage de la boîte de dialogue. Cette fonction prend en paramètre :

- La fenêtre parente qui va être mise en pause tant que la boîte de dialogue est affichée ;
- le texte à afficher dans la boîte de dialogue ;
- le titre de la fenêtre nouvellement créée ;
- le type de message (mêmes choix que précédemment) ;

- une image pour remplacer l'icone par défaut ;
- un tableau de choix ;
- une valeur par défaut.

Les cinq derniers paramètres sont optionnels et ont pour valeur par défaut "Message", `JOptionPane.INFORMATION_MESSAGE`, `null`, `null` et `null`. Si le tableau de choix a pour valeur `null`, la fenêtre utilisera un champ de texte pour demander une information ; sinon une liste déroulante est utilisée présentant chacun des éléments du tableau.

En réutilisant la version où le label est présent dans la fenêtre principale, remplacez le champ de texte par l'utilisation d'une boîte de dialogue ayant un effet similaire. Lors d'un clic sur le bouton, la boîte de dialogue doit apparaître et, une fois que celle-ci est fermée, le label doit être mis à jour. Veillez à remettre le message « Pas de données » si l'utilisateur n'a pas renseigné de valeur ou s'il a annulé la saisie.

2 Échanger des informations par des variables

Il est parfois nécessaire de stocker une information avant de pouvoir l'écrire dans un composant graphique. Pour cela, de simples variables ne suffisent pas et il faut utiliser des attributs.

En effet, à chaque clic sur un bouton sa méthode « `actionPerformed` » va être exécutée. Les variables déclarées dans cette méthode vont être créées, initialisées, utilisées... puis détruites à la fin. L'exécution suivante va recommencer ce processus et n'aura, donc, pas connaissance des valeurs précédentes pour les variables de cette fonction. Il y a donc PERTE D'INFORMATION entre chaque clic sur un bouton.

Les attributs sont des variables plus globales et sont créées en dehors des « `actionPerformed` ». Elles ne sont donc pas détruites à la fin d'une exécution et permettent de sauvegarder des informations entre deux clics sur un bouton. Elles permettent même de partager des informations entre différents boutons. Il est d'usage de les déclarer en début de fichier après la ligne `class NomClasse extends JFrame {`.

À l'aide d'un attribut, créez une fenêtre contenant deux boutons. Le premier va déclencher l'apparition d'une boîte de dialogue possédant un champ de texte et le second va déclencher l'apparition d'une boîte de dialogue présentant comme message la valeur du champ de texte précédemment validé (ou le message « Pas de message » le cas échéant). Faites aussi en sorte que la valeur par défaut présentée dans le champ de texte soit la dernière valeur entrée.

3 Compter les clics

À partir d'une fenêtre vierge, créez trois boutons. Chacun de ces boutons va mettre à jour un label (le même pour tout le monde) qui va indiquer le nombre de clics. Le premier bouton va incrémenter le total de 1, le second de 2 et le troisième de 3. Si après cet ajout, le nombre remplit une condition, le total doit être ramené à 0.

La condition pour le bouton 1 est que le nombre soit multiple de 5, celle pour le bouton 2 est que le nombre soit multiple de 11 et celle pour le bouton 3 est que le nombre soit impair.

Quel est le plus haut total possible ?