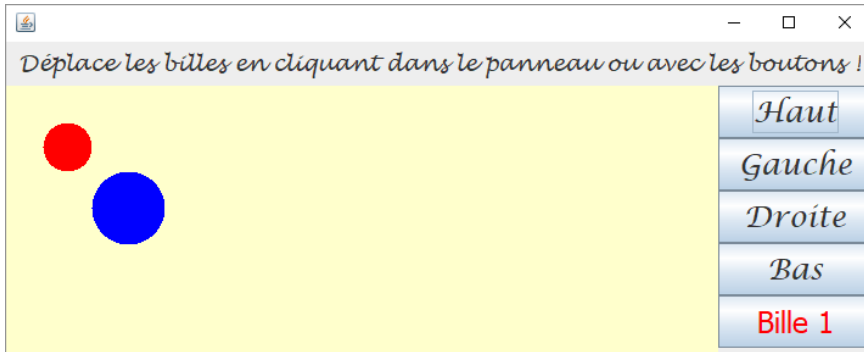


TP 6

Exercice 1 : Les billes

Implémenter l'application « Les Billes » présentée en CM5 et dont l'interface est rappelée ci-dessous.



Le tracé des billes est réalisé dans le code du programme. L'interface est construite par l'EDI.

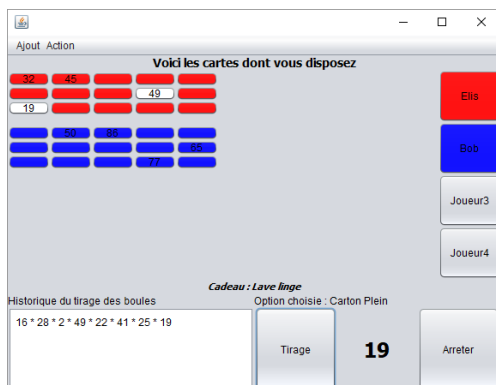
Exercice 2 : Projet Jeu de Loto

On souhaite compléter l'application du Jeu du Loto mise en place au TP4.

On rappelle que l'option de menu « Ajout » comporte deux sous-options « Joueur » et « Lot ». L'option de menu « Action » comporte 5 sous-options (actions) qui sont :

- « Options Jeu » pour paramétrer le mode de jeu
- « Acheter cartes » pour permettre à un joueur d'acheter de nouvelles cartes
- « Démarrer Jeu » pour commencer le jeu
- « Continuer » pour continuer le jeu si le paramétrage était l'option « Queene » ou « Double Queene »
- « Recommencer » pour réinitialiser le jeu et recommencer une partie.

L'interface comporte des boutons pour les joueurs (4 joueurs maximum possibles), pour le tirage, et pour arrêter le jeu.



L'historique des tirages est affiché dans une zone d'édition (de type JTextArea).

Les cartes sont affichées « dynamiquement » par l'application (les cases des cartes sont des boutons, non cliquables). Les autres éléments sont affichés dans des labels (JLabels).

La classe principale (JFrame) nommée « LeJeuLoto » et son interface, les classes Joueur, LesJoueurs, Carte et LesCartes ont été implémentées dans le TP5.

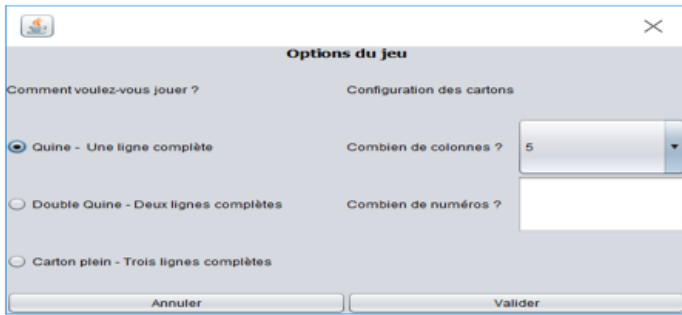
1. Terminer le TP4 exercice 2 (pour le programme du loto) le cas échéant et le TP5.

Mise en place de la boîte de dialogue « OptionDlg ».

Cette boîte de dialogue sert pour choisir les options de jeu. Elle propose de choisir une des 3 options disponibles pour le jeu (Quine, Double Quine, ou Carton plein), de définir le nombre de colonnes par carte dans une JComboBox nommée « ComboNbCol » ainsi que le nombre de numéros sur le carton dans un JTextField nommé « NbNumero ».

Un message peut être affiché en dessous de cette zone d'édition (JLabel nommé « Message »).

Par défaut, c'est l'option « Quine » qui est sélectionnée, 5 colonnes et 5 numéros. L'interface de cette boîte de dialogue est décrite ci-dessous.



2. Ajouter au projet une classe dérivée de la classe JDialog nommée « OptionDlg » (en vous reportant au TP5 si besoin pour l'ajout d'une boîte de dialogue).
3. Construire l'interface de la boîte de dialogue comme décrit ci-dessus.
4. Ajouter les attributs suivants dans la classe en les expliquant en commentaire dans votre code (type et rôle de chaque attribut).

```
private int choixOpt;
private int nbCol;
private int nbNum;
```

5. Ajouter des accesseurs (getNbCol, getChoix et getNbNum) pour ces attributs.

L'appel de la boîte de dialogue peut se faire soit au début du jeu, soit au cours du jeu. Si la boîte de dialogue est ouverte au début du jeu, avant que les joueurs aient acheté des cartes, il est possible de choisir le nombre de colonnes d'une carte et son nombre de numéros. Si la boîte de dialogue est ouverte en cours de jeu et que des cartes ont déjà été créées (achetées), seule l'option de jeu est modifiable.

Il est donc nécessaire de savoir si des cartes ont été achetées, et si oui, les nombres de colonnes et de numéros qui ont été choisis.

La boîte de dialogue reçoit de l'information : un « boolean » qui indique si des cartes existent déjà, le nombre de colonnes, et le nombre de numéros et elle renvoie de l'information : le choix de l'option de jeu, le nombre de colonnes et le nombre de numéros. Le constructeur de cette classe « OptionsDlg » est partiellement décrit ci-dessous. Le compléter en mettant des commentaires y compris pour le code donné, et en expliquant son entête.

```
public OptionDlg(java.awt.Frame parent, boolean modal, boolean ca, int nc, int nn) {
    super(parent, modal);
    initComponents();
    remplirComboCol(); // Méthode qui permet d'initialiser les valeurs de la liste déroulante à 5,6,7,8 et 9.
    // Cocher l'option Quine par défaut – Les boutons doivent être groupés
    ...
    // Fixer le choix de l'option de jeu à 1
    ...
    if (!ca) // test si des cartes ont déjà été achetées
    { // Fixer le nombre de colonnes à 5 et le nombre de numéros à 5
        ...
    }
    else
    { // Fixer le nombre de colonnes et de numéros aux valeurs reçues en paramètres
        ...
        // Afficher la valeur du nombre de numéros dans la zone d'édition (JTextField)
```

```

...
// Sélectionner dans la liste déroulante l'index de la valeur du nombre de colonnes
// en utilisant la méthode (setSelectedIndex(..))
...
// Rendre la liste et la zone d'édition non utilisables, en utilisant la méthode setEnabled(boolean)
...
// Afficher le message : « Nb de colonnes et de numéros non modifiables »
// dans le JLabel nommé « Message »
}
}

```

6. Donner le code la méthode « `private void remplirComboCol();` » qui remplit la liste déroulante avec les valeurs 5,6,7,8 et 9.
7. Donner le code du gestionnaire d'évènement du clic sur le bouton « Valider » qui
 - fixe la valeur du choix de jeu (`choixOpt`) en fonction du bouton coché,
 - récupère le nombre de colonnes sélectionné et l'affecte à `nbCol`
 - si la zone d'édition correspondant au nombre de colonnes est vide, un message d'erreur est affiché
 - sinon
 - o récupère le nombre de numéros choisi (ou fixé) dans la zone d'édition et l'affecte à `nbNum`,
 - o si le nombre de numéros est <5 ou supérieur au nombre de colonnes $*3$, un message d'erreur est affiché
 - o sinon
 - ferme la boîte de dialogue.

private void ValiderActionPerformed(java.awt.event.ActionEvent evt)

Donner le code du gestionnaire d'évènement du clic sur le bouton « Annuler » qui ferme la boîte de dialogue.

private void AnnulerActionPerformed(java.awt.event.ActionEvent evt)

Application principale « LeJeuLoto ».

8. Compléter la classe principale, en ajoutant et en les commentant les attributs nécessaires pour la gestion des options, comme décrit ci-dessous.

```

private int option;
private int nbCol ;
private nbNum;

```

9. Initialiser ces attributs à la valeur 1, pour l'option, 5 pour le nombre de colonnes, et 4 pour le nombres de numéros, dans le constructeur de la classe.
10. Donner le code du gestionnaire d'évènement du clic sur la sous-option de menu « Choisir » qui :
 - Déclare et crée une boîte de dialogue « `OptionDlg` »
 - Affiche la boîte de dialogue (méthode `setVisible(...)`)
 - Puis à la fermeture de la boîte :
 - Récupère le nombre de colonnes et l'affecte à l'attribut « `nbCol` »
 - Récupère le nombre de numéros et l'affecte à l'attribut « `nbNum` »
 - Récupère le choix de jeu fait et l'affecte à l'attribut « `option` »

- Et selon l'option choisie, affiche sur l'interface l'option choisie (comme montré ci-dessus)
 - Option Choisie : Quine
 - Option Choisie : Double Quine
 - Option Choisie : Carton plein