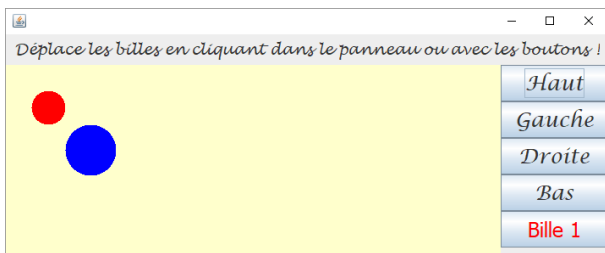


TP6

Exercice 1 : Les billes

Implémenter l'application « TestDessin » étudiée en CM5 dont l'interface est rappelée ci-dessous.



Les boutons de direction permettent de déplacer la bille sélectionnée de 10 pixels dans le sens choisi. Le bouton Bille permet de fixer la bille sélectionnée (Bille 1 ou Bille 2). Le clic dans le panneau de dessin permet de déplacer directement la bille sélectionnée à la position du clic souris.

Exercice 2 : Le jeu de memory

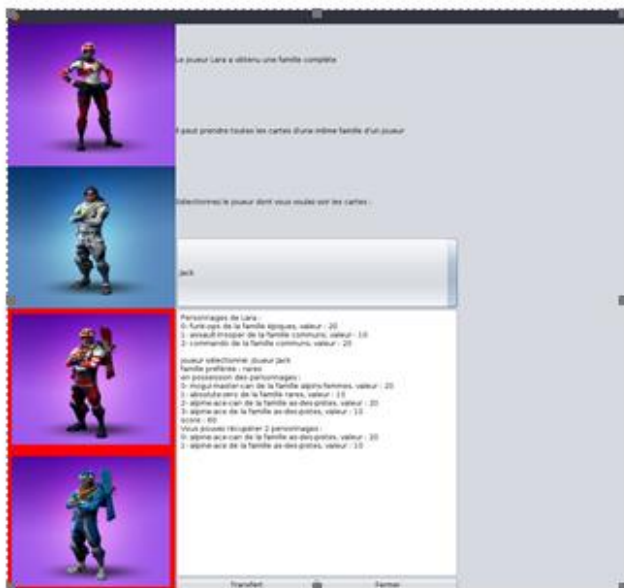
LIRE ATTENTIVEMENT le sujet complet du projet et notamment la partie « Transfert » de cartes.

Au cours du jeu de memory, si le joueur courant obtient tous les personnages de la famille « rares » ou « communs », il peut « prendre » tous les personnages d'une famille, d'un autre joueur.

Lorsque cette situation se produit une boîte de dialogue s'ouvre pour effectuer : la sélection du joueur auquel on souhaite prendre des cartes, puis la famille des cartes que l'on veut prendre à ce joueur. Dans le jeu, la boîte ne s'ouvrira que si au moins un autre joueur a des cartes.

La sélection du joueur à qui on souhaite prendre des cartes, est réalisée dans une liste déroulante (contenant tous les noms des joueurs). Lorsqu'un joueur est sélectionné dans la liste, les photos des personnages qu'il a déjà gagnés sont affichées (à gauche sur l'image écran). Le clic sur la photo d'un personnage permet de visualiser tous les personnages de la même famille possédés par ce joueur (encadrement en rouge des photos des personnages de la famille comme le montre l'image écran ci-dessous). Les informations textuelles sur les personnages possédés par le joueur sélectionné sont également affichées dans la zone d'édition.

Le transfert « effectif » est réalisé lors du clic sur le bouton « Transfert ». Les cartes transférées sont alors affichées à droite de la boîte de dialogue et supprimées de la gauche de cette boîte dialogue.



Après un clic sur l'un des personnages



Après un clic sur le bouton "Transfert"

1. Ajouter dans le projet une classe nommée « TransfertDlg » qui dérive de la classe « JDialog ».
2. Construire l'interface de cette boîte de dialogue en vous aidant des images-écran ci-dessus.

La boîte de dialogue est organisée en GridLayout (1,3). Elle comporte en

- un panneau JPanel nommé PanneauG, organisé en GridLayout(1,1), vide initialement et qui sera rempli dynamiquement par le programme
- un panneau JPanel, contenant des messages indiquant au joueur courant de sélectionner un autre joueur à l'aide de liste déroulante. Cette liste est une JComboBox nommée «ComboJoueurs, une zone d'édition nommée Infos de type JTextArea, et deux boutons Transfert, et Fermer.
- un panneau JPanel nommé PanneauD, organisé en GridLayout(1,1), vide initialement et qui sera rempli dynamiquement par le programme

3. Compléter « partiellement » le code de cette classe « TransfertDlg » en suivant les étapes ci-dessous. Le code complet de la classe sera réalisé après l'étude de la modélisation des actions du jeu.

a. Ajouter les attributs nécessaires à la classe comme décrit ci-dessous.

```
private LesJoueurs lj; //collection des joueurs, pour initialiser la liste déroulante avec les pseudos des joueurs
private int indj; //indice joueur courant
// private Transfert tc; // cette classe sera étudiée ultérieurement – laisser l'attribut en commentaire
private boolean ok; // indicateur pour savoir le transfert a bien été effectué.
private int indjs; //indice du joueur sélectionné dans la liste déroulante
private String fs; //famille du personnage sélectionné en cliquant sur un des personnages du joueur sélectionné
```

b. Ajouter des accesseurs « get » pour le booléen « ok » et pour l'attribut de type transfert, en le mettant en commentaire dans cette version.

c. Etudier précisément le constructeur ci-dessous de la classe, l'implémenter en le commentant et en répondant aux questions suivantes

- i. Quelles sont les informations reçues par la boîte de dialogue de la part de l'application principale ?
- ii. Quelles sont à votre avis, les informations que doit retourner la boîte de dialogue à l'application principale ?

```
public TransfertDlg(java.awt.Frame parent, boolean modal, LesJoueurs lj, int indj) {
    super(parent, modal);
    initComponents();
    this.lj = lj;
    this.indj = indj ;
    // this.tc=null;
    this.ok=false;
    this.fs=null;
    initCombo(); // méthode pour remplir la liste déroulante
    indjs = 0;
    Message.setText("Le joueur "+lj.getJoueur(indj).getPseudo()+" a obtenu une famille complète");
    Infos.setText("Personnages de "+lj.getJoueur(indj).getPseudo()+" : \n"+lj.getJoueur(indj).getPaquet());
}
```

NB : Message est un JLabel de l'interface de la boîte, et Infos, la zone d'édition de type JTextArea

d. Donner le code de la méthode « public void initCombo() » qui permet de remplir la liste déroulante JComboBox nommée « ComboJoueurs » avec les pseudos des joueurs.

e. Implémenter en le commentant précisément le code du gestionnaire de la sélection du pseudo d'un joueur dans la liste déroulante. Ce gestionnaire appelle 2 méthodes « initPanneau () » et « affichePanneau() ».

```
private void ComboJoueursActionPerformed(java.awt.event.ActionEvent evt) {
    this.indjs = ComboJoueurs.getSelectedIndex();
    if (indjs != -1){
        if (this.indjs == this.indj) {
            Infos.setText("Sélectionnez un joueur différent du joueur courant !");
            PanneauG.removeAll();
            PanneauG.repaint();
        }
        else {
            Infos.setText("\nJoueur sélectionné: "+lj.getJoueur(indjs).toString());
            initPanneau();
            affichePanneau();
        }
    }
}
```

- f. La méthode « public void initPanneau() » permet de remplir le JPanel PanneauG avec des boutons pour afficher les personnages possédés par le joueur sélectionné. Cette méthode réalise les actions suivantes :
- Suppression de tous les composants du panneau : PanneauG.removeAll();
 - Ré-affichage de la boîte de dialogue : this.repaint();
 - Récupération dans une variable nommée lcs, de type LesPersonnages, des personnages du paquet du joueur sélectionné
 - Initialisation de la disposition du panneau PanneauG, en fonction du nombre de personnages qu'il possède

```
int t = lcs.getTaille();
int n = 1+(t-1)/4;
PanneauG.setLayout(new java.awt.GridLayout(4, n));
```

- Boucle pour créer "t" boutons. Chaque bouton créé a pour « name », la famille du personnage à qui correspond le bouton : bt.setName(lcs.getPerso(i).getFamille(), est abonné à un écouteur de type action, pour exécuter le gestionnaire « boutonActionPerformed(evt); » et est ajouté au panneau PanneauG. La méthode this.pack() est appelée à la fin de cette méthode.
- La méthode « public void affichePanneau() » permet de récupérer chaque bouton et d'afficher sur le bouton la photo du personnage correspondant, en utilisant une boucle sur les personnages du paquet du joueur.
- Le gestionnaire d'évènement du clic sur un bouton du panneau PanneauG, est donné ci-dessous. Implémenter ce gestionnaire en le commentant précisément.

```
private void boutonActionPerformed(ActionEvent evt)
{
    LesPersonnages lp = lj.getJoueur(indjs).getPaquet();
    int t = lp.getTaille();
    JButton bt=(JButton) evt.getSource();
    fs = bt.getName(); // la propriété Name, contient ici le nom du personnage affiché sur le bouton
    for(int i = 0; i < t; i++) {
        JButton b = (JButton)(PanneauG.getComponent(i));
        if (b.getName().equals(fs))
            b.setBorder(BorderFactory.createMatteBorder(10, 10, 10, 10, new java.awt.Color(255, 0, 0)));
        else
            b.setBorder(null);
    }
    LesPersonnages lps = lp.getPersosFamille(fs);
    Infos.setText("Vous pouvez récupérer "+lps.getTaille()+" personnages : \n"+lps);
}
```

- Donner le code du gestionnaire du clic sur le bouton « Fermer » qui ferme la boîte de dialogue uniquement.

Le code du gestionnaire du clic sur le bouton « Transfert » sera étudié ultérieurement.

4. Le test de la boîte de dialogue pourra être fait en ajoutant dans la classe « JeuMemory » une option de menu nommée « Test » et une sous-option « Transfert_Test » avec le gestionnaire suivant :

```
private void Transfert_TestActionPerformed(java.awt.event.ActionEvent evt) {
    Joueur j1=new Joueur("FanMemory0", "commun");
    j1.initPaquetTest();
    Joueur j2=new Joueur("FanMemory1", "commun");
    j2.getPaquet().ajoutePerso(new Personnage("epiques", "burnout", 20));
    j2.getPaquet().ajoutePerso(new Personnage("epiques", "funk-ops", 30));
    j2.getPaquet().ajoutePerso(new Personnage("alpines-femmes", "mogul-master", 10));
    this.joueurs.ajouteJoueur(j1);
    this.joueurs.ajouteJoueur(j2);
    TransfertDlg diag = new TransfertDlg(this,true,joueurs,0 );
    diag.setVisible(true);
}
```