Licence Sciences et Technologies – 1ère année
Module Info2B
Interfaces Visuelles

TP4

Exercice 1 : La Palette des Couleurs

Implémenter l'application la palette des couleurs **présentée en cours** dont une partie a été décrite en cours (CM3) et **dont le code complet a été écrit au tableau**. On rappelle que les boutons sont construits dynamiquement dans le programme par une méthode initPanneau() et non par l'EDI de Netbeans. Il y a 400 boutons de couleurs différentes. Le clic sur un bouton colorie tous les boutons du panneau de la couleur de ce bouton et affiche en information la couleur sélectionnée (rouge, vert, bleu).

Exercice 2 : Jeu de memory

Mise en place de boîte de dialogue « VisuJoueursDlg » pour la visualisation des joueurs (voir CM4)

1. Ajouter dans le projet « JeuMemory » une classe nommée « VisuJoueursDlg » qui dérive de la classe JDialog (JDialog Form) – *(si besoin choisir l'option other puis Swing GUI Forms et JDialog)*
2. Construire l'interface de cette classe sans les boutons qui sont gérés dynamiquement.
3. Compléter le code de cette classe « VisuJoueursDlg » en ajoutant :
   a. Un attribut de type « LesJoueurs » nommée « lj » et initialisé dans le constructeur avec le paramètre de ce type passé à ce constructeur (voir CM4)
   b. Le constructeur appelle également une méthode nommée « initTrombi() » qui permet de créer dynamiquement les boutons dans le panneau de type « JPanel » nommé « Panneau » dont le code est rappelé ci-dessous.
   Implémenter cette méthode et commenter ligne à ligne ses instructions, puis en une phrase de synthèse pour expliquer son rôle

```
private void initTrombi()
{   int nb = this.lj.getNbJoueurs();
   Panneau.setLayout(new GridLayout(1, nb));
   for (int i=0; i< nb; i++)
   {
      JButton jb= new JButton();
      Joueur j= lj.getJoueur(i);
      jb.setName(""+i);
      jb.addActionListener(new java.awt.event.ActionListener() {
         public void actionPerformed(java.awt.event.ActionEvent evt) {
            AfficheInfosJoueur(evt);
         }
      });
      Panneau.add(jb);
   }
   this.pack(); // pour ajuster correctement la taille des composants après les ajouts si besoin
```

4. Lors du clic sur un bouton le gestionnaire « AfficheInfosJoueur » est exécutée. Il affiche les informations textuelles du joueur dans la zone d'édition de type « JTextArea » nommée « Edition ».
   private void AfficheInfosJoueur(java.awt.event.ActionEvent evt) { . . . }

5. Lors du clic sur le bouton « Afficher », les photos des joueurs sont affichées sur les boutons. Donner le code de ce gestionnaire. Il récupère chaque bouton du Panneau et affiche la photo du joueur sur le bouton (la méthode getComponent devra être utilisée).
   private void AfficherActionPerformed(java.awt.event.ActionEvent evt)

6. Ajouter dans la classe principale « JeuMemory » qui dérive de la JFrame, les attributs nécessaires
   private Les Personnages persos ;
   private LesJoueurs joueurs ;
7. Compléter le constructeur de la classe « JeuMemory » pour initialiser ces attributs.
8. Ajouter 2 joueurs « Lara » et « Jack » également dans le constructeur pour pouvoir tester la boîte de visualisation.

```
Joueur j=new Joueur("Lara", "epiques");
j.setPhoto(new ImageIcon(getClass().getResource("/JeuMemory/img/lara.jpg")));
joueurs.ajouteJoueur(j);
j=new Joueur("Jack", "rares");
j.setPhoto(new ImageIcon(getClass().getResource("JeuMemory/img/jack.png")));
joueurs.ajouteJoueur(j);
```

9. Ajouter le gestionnaire de la sous-option « Joueurs » de l'option « Visualiser » du menu (comme présenté en CM4).

Exemple d'interface avec les 2 joueurs Lara et Jack.

<u>Exercice 3 : Jeu de memory</u>
<u>Mise en place de boîte de dialogue « InitDlg » pour la sélection des options du jeu de memory (voir CM4)</u>

La boîte de dialogue « InitDlg » a été ajoutée et construite dans le projet « JeuMemory » lors du TP3. Il s'agit ici de compléter le code de cette classe pour la rendre fonctionnelle.

1. Compléter le code de la classe « InitDlg » comme présenté au CM4.
    a. Ajouter les attributs nécessaires à la classe (pour la liste des joueurs, le niveau et le mode de fermeture)
    b. Ajouter les accesseurs d'accès en lecture (get) pour ces attributs.
    c. Compléter le constructeur de la classe pour initialiser les attributs.
    d. Ajouter les gestionnaires de clic sur les boutons radio qui permettent de fixer le niveau du jeu.
    e. Ajouter le gestionnaire du clic sur le bouton « Valider »
    f. Ajouter le gestionnaire du clic sur le bouton « Annuler ».

    Le clic sur une case à cocher, par exemple pour la case à cocher nommée « CaseLara » de type « JCheckBox » permet de créer de façon temporaire, le joueur « Lara », d'afficher ses informations textuelles dans la zone d'édition nommée « Edition » de type « JTextArea » et sa photo sur le bouton « BPhoto » comme le montre le code du gestionnaire ci-dessous. Implémenter ce gestionnaire en commentant les instructions.

```
private void CaseLaraActionPerformed(java.awt.event.ActionEvent evt) {
    Joueur j=new Joueur("Lara", "epiques");
    j.setPhoto(new ImageIcon(getClass().getResource("/memory/img/lara.jpg")));
    Edition.setText(j.toString());
    Image img = j.getPhoto().getImage().getScaledInstance(BPhoto.getWidth(), BPhoto.getHeight(), Image.SCALE_SMOOTH);
    BPhoto.setIcon(new ImageIcon(img));
```

    g. Implémenter les autres gestionnaires pour les joueurs :
        i. pseudo : Jack, famille préférée « rares », photo « jack.png »
        ii. pseudo : Jean-Sébastien, famille préférée « alpins-femmes », photo « bach.jpg
        iii. pseudo : Amadeus, famille préférée « communs », photo « amadeus.jpg »
2. Ajouter le gestionnaire de la sous-option « Options » de l'option « paramètres » du menu (comme présenté en CM4) qui permet l'ouverture de la boîte de dialogue et la prise en compte des informations sélectionnées, le cas échéant. Chaque joueur de la liste sélectionnée est ajouté à l'ensemble des joueurs (attribut joueurs de la classe JeuMemory). Le niveau est utilisé pour appeler le constructeur avec paramètre de la classe LesPersonnages, pour l'ensemble des personnages (attribut persos de la classe JeuMemory).
   private void OptionsActionPerformed(java.awt.event.ActionEvent evt) { …}