

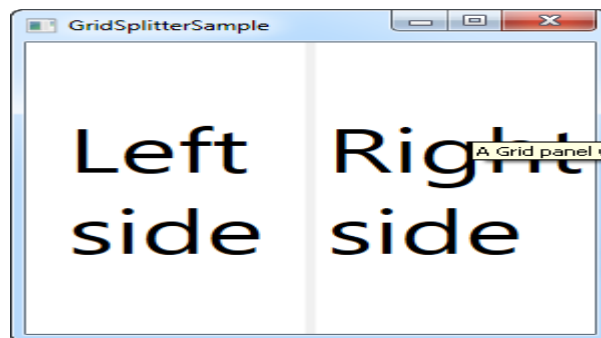
TP1 : WPF de base

I. Les contrôles/Styles

Exercice1 :

Créez un projet WPF et testez les contrôles suivants :

- Grid,StackPanel,warpPanel et DockPanel
- GridSplitter en créant l'interface suivante qui permet de redistribuer l'espace entre les deux colonnes.



- Expander



Exercice2 : Tester le UserControl

Exercice3 : Créer un projet wpf, ouvrir dans Expression Blend et faire les modifications de design dans Expression Blend

Exercice4 :

1. Créer un style qui modifie la couleur d'un bouton puis appeler ce style.
2. Créer un autre style appliqué sur un bouton permettant d'afficher un message dans un textBox lors un événement de click.
3. Créer un fichier ResourceDictionary1.xaml en Expression Blend portant un ensemble de style puis importer ce fichier dans un projet WPF et essayer d'appliquer ses styles.

II.Triggers

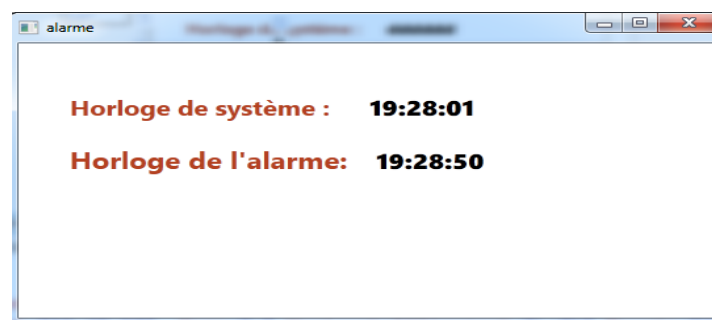
Exercice5 :

-Créer une fenêtre wpf où on a un TextBox ,un Button et un CheckBox :

1. Changer la couleur de Background de TextBox au vert si le texte tapé égale à salut
- 2.Changer la couleur de textBox si on passe la souris au dessus et on fait une sélection par clavier (IsKeyboardFocused=True)
3. Mettre la couleur de background de Button en vert si le curseur de souris est au dessus et en rouge si on sélectionne le bouton
4. Mettre le background de TextBox en rouge si on coche le checkbox.

Exercice6:

- 1.Créer une petite alarme qui déclenche un son lorsque l'horloge de système arrive à celle réglée par l'utilisateur.

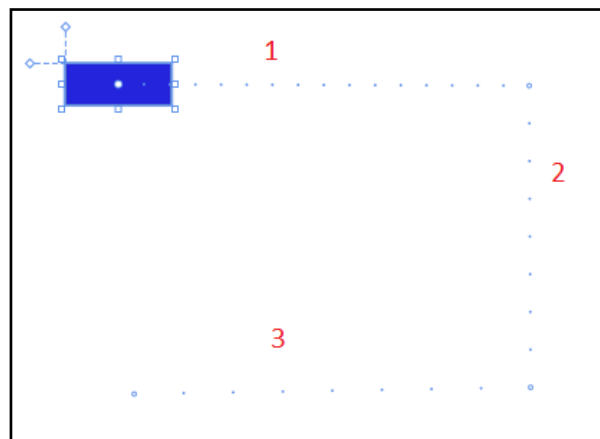


- 2.Fermer l'application après 3 secondes de son démarrage en utilisant une action personnalisée responsable de fermeture de programme.

III.Animation

Exercice7:

- 1.Faire déplacer le rectangle en suivant la trajectoire de figure ci-dessous



- 2.Refaire la question-1 mais cette fois, le rectangle doit faire un retour pour revenir à la position initiale

3. Ajouter un autre rectangle et appliquer une animation de redimensionnement (agrandir puis réduire)

4. Ajouter un autre rectangle et appliquer une animation de rotation.

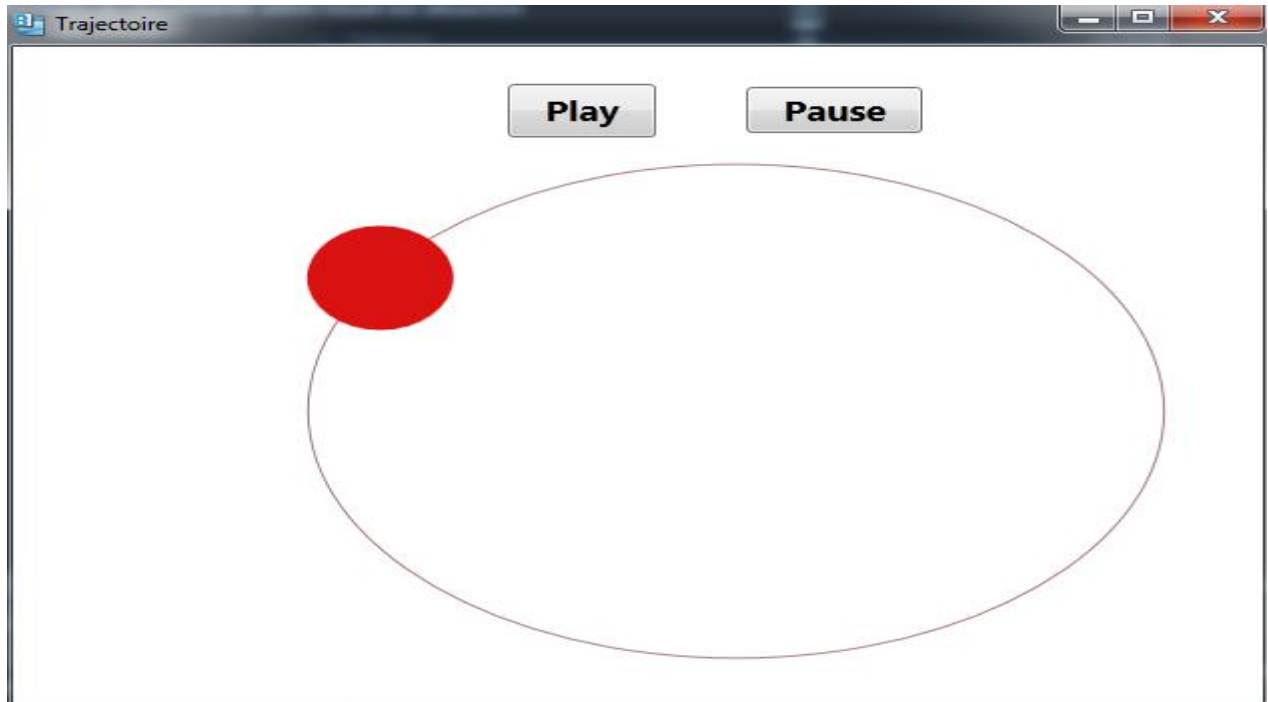
5. Ajouter un autre rectangle et ajouter une animation de couleur.

Exercice8:

Ajouter deux boutons button1 et button2 tel que:

-Button1: Active l'animation

-Button2: Désactive l'animation



Annexe:

1. création de l'horloge

-DispatcherTimer: Minuterie intégrée dans la file d'attente de Dispatcher qui est traitée à un intervalle de temps et selon une priorité spécifiés.

-Evenement.Tick: Se produit lorsque l'intervalle de temporisation est écoulé.

-TimeSpan: Représente un intervalle de temps.

Exemple d'une horloge dont l'heure est mise à jour toutes les secondes

```
private void startclock()
{
    DispatcherTimer timer = new DispatcherTimer();
    timer.Interval = TimeSpan.FromSeconds(2);
    timer.Tick += tickevent;

    timer.Start();
}
```

```
private void tickevent(object sender, EventArgs e)
{
    t1.Text = DateTime.Now.Hour.ToString("00") + ":" +
DateTime.Now.Minute.ToString("00") + ":" +
DateTime.Now.Second.ToString("00");
}
```

2. programme de fermeture d'une application wpf:

```
App.Current.Shutdown();
```