Vizuális programozás gyakorlat

Képnézegető alkalmazás WPF alapú felülettel

Készítsen egy WPF képnézegető alkalmazást, ami a mellékelt ábrának megfelelően a bal oldali oszlopban (Grid) egy könyvtárban található jpg képek bélyegképeit jeleníti meg, a jobb oldali oszlopban az éppen kiválasztott kép nagyban látható.

L:\Gyakorlat\05\Kepek\Kepek			
Könyvtárválasztás			
		4	SHATTAKANK
die Ale			副脚為前期
		•	file:///L:/Gyakorlat/05/Kepek/Kepek/HPIM2486.JPG

A két oszlop között egy elválasztó sáv legyen (GridSplitter), ami lehetővé teszi, hogy a felhasználó változtathassa az oszlopok szélességét.

Megoldás

Készítsük egy WPF projektet Kepek néven. Az ablak állományának neve legyen wndFoablak.xaml és osztályának neve legyen wndFőablak.

1. A felület elkészítése

Az ablakban két oszlopot alakítsunk ki: első a bélyegképek és a mappaválasztó nyomógomb, valamint az elválasztó vonal számára, míg a második a nagyméretű kép számára. Az első szélessége "*" legyen, míg a másodiké Auto.

A baloldali oszlopban két sort hozunk létre egy Grid beépítésével. Az első sorban egy nyomógomb található, amin kattintva egy könyvtárválasztó párbeszédablak jelenik majd meg. A mappakiválasztó párbeszédablak nem beépített, az őt tartalmazó kódot (FolderPickerLib.dll) a t:\info\Johanyák Csaba\Kepek\FolderPickerLib.dll útvonalon érhetjük el, és be kell másolni a projekt könyvtárába, majd fel kell venni a projekt referenciái közé.



A bélyegképeket egy ScrollViewer vezérlőre helyezett WrapPanel vezérlőn helyezzük el, ez biztosítja, hogy szükség esetén jelenjen meg a görgető sáv, és annyi oszlopban jelenjenek meg, amennyinek a vízszintes megjelenítésére lehetőség van.



Ha a bélyegképek felé visszük az egeret, akkor a kép meg kell növekedjen kissé egy animáció segítségével, majd az egér eltávolítását követően vissza kell zsugorodjon az eredeti méretére. Ehhez a megfelelő méretű helyet előre le kell foglalni, így a kép nem közvetlenül a WrapPanelre kerül, hanem egy keretre (Border), és a keretet tesszük a WrapPanelre. A keret mérete nagyobb a bélyegkép méreténél, és az animáció során a bélyegkép ki fogja tölteni a keretet.

A jobb oldalon a nagyméretű kép és a kép teljes elérési útvonala+állományneve jelenik meg egymás alatt. Ehhez a jobb oldalon is egy Gridet építünk be, ami kétsoros lesz. Az első sor magassága 25 lesz.

A felületet leíró XAML kód az alábbi:

```
<Window x:Class="Kepek.wndFőablak"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    Title="Válasszon könyvtárat!" Height="350" Width="525">
    <Grid Name="grRács">
```

```
<Grid.ColumnDefinitions>
        <ColumnDefinition />
        <ColumnDefinition Width="Auto"/>
     </Grid.ColumnDefinitions>
        <Grid Grid.Column="0" Margin="0,0,5,0" >
            <Grid.RowDefinitions>
                <RowDefinition Height="25" />
                <RowDefinition />
            </Grid.RowDefinitions>
                 <Button Name="btKönyvtárVálasztó"
                                   Content="Könyvtárválasztás ..."
                                   Click="btKönyvtárVálasztó Click" />
                 <ScrollViewer Name="swKépek"</pre>
  VerticalScrollBarVisibility="Auto"
                                                         Grid.Row="1">
                    <WrapPanel Name="wpKépek"</pre>
                                                   HorizontalAlignment="Right" />
                 </ScrollViewer>
        </Grid>
        <GridSplitter ResizeDirection="Columns"
                                           Grid.Column="0"
                                           Width="5" />
         <Grid Grid.Column="1">
            <Grid.RowDefinitions>
                <RowDefinition />
                <RowDefinition Height="25" />
            </Grid.RowDefinitions>
            <Image Name="imNagyKép" MinWidth="100"</pre>
                                   MinHeight="100" Grid.Row="0"
                                    Stretch="Uniform" />
         <TextBlock Name="tlKépnév" Height="15"
                                   VerticalAlignment="Center" Grid.Row="1"
                                   Margin="5,0,0,0"/>
      </Grid>
   </Grid>
</Window>
```

A feladathozt felhasználható mintaképek a t:\info\Johanyák Csaba\Kepek\ könyvtárban találhatóak meg.

A feladatot megvalósító kód

```
1. Adattagok
```

```
/// <summary>
/// A bélyegkép vezérlő alap szélessége.
/// </summary>
private double KépSzélesség;
/// <summary>
/// A bélyegkép vezérlő alap magassága.
/// </summary>
private double KépMagasság;
/// <summary>
/// Ennyi ideig tart az animáció.
/// </summary>
private TimeSpan tsAnimációIdő;
/// <summary>
/// Az animáció során ennyivel nő a kép vezérlő szélessége.
/// </summary>
```

```
private double dSz;
/// <summary>
/// Az animáció során ennyivel nő a kép vezérlő magassága.
/// </summary>
private double dM;
```

2. Konstruktor

A konstruktorban megadjuk a bélyegképet megjelenítő vezérlő méreteit. A feladat egyszerűsítése érdekében itt úgy dolgozunk, hogy előre tudjuk, hogy a képek 1024x766-os felbontásúak lesznek. Az animációhoz fél másodperces időtartamot adunk meg. Az aktuális könyvtár (amiből a képeket meg fogja jeleníteni a program) elérési útvonalát az ablak fejlécében tároljuk, illetve jelenítjük meg. Ennek kezdőértékét (C:\) is a konstruktorban adjuk meg. Végül beolvassuk a memóriába az adott könyvtárban levő képeket. Ez utóbbi feladatot egy külön metódussal oldjuk meg (KépeketBetölt()). A metódus vázát automatikusan generáltatjuk a Visual Studioval.

```
/// <summary>
/// A főablak konstruktora. Adattagok inicializálása és kezdőképek betöltése.
/// </summary>
public wndFőablak()
{
  InitializeComponent();
  // A kép vezérlő eredeti szélessége.
  KépSzélesség = 70;
  // A kép vezérlő eredeti magassága. A felhasznált mintaképek mind 1024x766-os
  // méretűek.
  KépMagasság = KépSzélesség*766/1024;
  // Az animáció során ennyivel nő a kép vezérlő szélessége.
  dSz = 30;
  // Az animáció során ennyivel nő a kép vezérlő magassága.
  dM = dSz * 766 / 1024;
  // Az animáció időigényének megadása.
  tsAnimációIdő = TimeSpan.FromMilliseconds(500);
  // A kezdőkönyvtár (képeket tároló könyvtár) megadása.
  Title = @"C:\";
  // Beolvassuk a képeket a könyvtárból.
  KépeketBetölt();
}
```

2. Képek betöltése

```
/// <summary>
/// Betölti a képeket a kiválasztott mappából, és kép
/// vezérlők formájában elhelyezi őket a WrapPanel-en.
/// </summary>
private void KépeketBetölt()
{
    // A képeket tartalmazó könyvtárt leíró objektum létrehozása.
    DirectoryInfo Di = new DirectoryInfo(Title);
    // Töröljük a WrapPanel-en levő vezérlők listáját.
    wpKépek.Children.Clear();
    try
```

```
{
  // Lekérdezzük a jpg kiterjesztésű állományokat a könyvtárból.
  FileInfo[] Fi = Di.GetFiles("*.jpg");
  // Minden képet beolvasunk.
  foreach (FileInfo Fájl in Fi)
  {
     // A helyörző létrehozása. Ez nagyobb kell legyen, mint a kép vezérlő.
     // Amikor növeljük a kép vezérlő méretét, a helyörzőt fogja kitölteni.
     Border bdHelyörző = new Border();
     bdHelyörző.Width = KépSzélesség + dSz;
     bdHelyörző.Height = KépMagasság + dM;
     // Felvesszük a helyörzőt a panelre.
     wpKépek.Children.Add(bdHelyörző);
     // Létrehozunk egy kép objektumot, és betöltjük a fájlból a képet.
     var imKép = new Image
     {
        // Kép forrás megadása.
        Source = new BitmapImage(new Uri(Fájl.FullName, UriKind.Absolute)),
        Width = KépSzélesség,
        Height = KépMagasság
     };
     // A kép a kép vezérlőn töltse ki a rendelkezésre álló helyet
     // az eredeti képarány megtartásával.
     imKép.Stretch = Stretch.Uniform;
     // A kép vezérlő a helyörző közepére kerüljön.
     imKép.VerticalAlignment = VerticalAlignment.Center;
     imKép.HorizontalAlignment = HorizontalAlignment.Center;
     // Eseménykezelő rendelése az egérgomb lenyomáshoz.
     imKép.MouseDown += Kép_MouseDown;
     // Eseménykezelő rendelése az egér vezérlő fölé érkezéséhez.
     imKép.MouseEnter += imKép_MouseEnter;
     // Eseménykezelő rendelése az egér vezérlő fölüli távozásához.
     imKép.MouseLeave += Kép MouseLeave;
     // Kép elhelyezése a helyörzőben.
     bdHelyörző.Child = imKép;
  }
}
catch (Exception exc)
{
  // Hibaüzenet, ha nem sikerült valamelyik művelet.
  MessageBox.Show(exc.Message);
}
// Ha vannak képek a panelen.
if (wpKépek.Children.Count > 0)
  // Beállítjuk a legelső képet nagynak.
  KépBeállít((Image)((Border)wpKépek.Children[0]).Child);
```

3. Az animációt megvalósító metódusok

}

```
/// <summary>
/// A felhasználó az egeret a kép vezérlő felé mozgatta.
/// </summary>
/// <param name="sender">A kép vezérlő objektum.</param>
/// <param name="e"></param>
void imKép_MouseEnter(object sender, MouseEventArgs e)
{
```

```
// Az aktuális kép objektum.
  var imKép = (Image)sender;
  // A vízszintes méretváltoztatást leíró animáció objektum.
  DoubleAnimation da = new DoubleAnimation();
  // Kezdőméret.
  da.From = KépSzélesség;
  // Végső méret.
  da.To = KépSzélesség + dSz;
  // Az animáció időtartama.
  da.Duration = new Duration(tsAnimációIdő);
  // A függőleges méretváltoztatást leíró animáció objektum.
  DoubleAnimation db = new DoubleAnimation();
  // Kezdőméret.
  db.From = KépMagasság;
  // Végső méret.
  db.To = KépMagasság + dM;
  // Az animáció időtartama.
  db.Duration = new Duration(tsAnimációIdő);
  // A két animáció elindítása.
  imKép.BeginAnimation(WidthProperty, da);
  imKép.BeginAnimation(HeightProperty, db);
}
/// <summary>
/// A felhasználó az egeret elmozgatta a képről.
/// </summary>
/// <param name="sender">A kép vezérlő objektum.</param>
/// <param name="e"></param></param>
void Kép_MouseLeave(object sender, MouseEventArgs e)
{
  var imKép = (Image)sender;
  // A vízszintes méretváltoztatást leíró animáció objektum.
  DoubleAnimation da = new DoubleAnimation();
  // Kezdőméret.
  da.From = KépSzélesség + dSz;
  // Végső méret.
  da.To = KépSzélesség;
  // Az animáció időtartama.
  da.Duration = new Duration(tsAnimációIdő);
  // A függőleges méretváltoztatást leíró animáció objektum.
  DoubleAnimation db = new DoubleAnimation();
  // Kezdőméret.
  db.From = KépMagasság + dM;
  // Végső méret.
  db.To = KépMagasság;
  // Az animáció időtartama.
  db.Duration = new Duration(tsAnimációIdő);
  // A két animáció elindítása.
  imKép.BeginAnimation(WidthProperty, da);
  imKép.BeginAnimation(HeightProperty, db);
}
```

4. A nagyméretű kép megjelenítése

```
/// <summary>
/// Beállítjuk a képet a nagy vezérlőben láthatónak.
/// </summary>
/// <param name="imKép">A megjelenítendő kép.</param>
private void KépBeállít(Image imKép)
{
    // A kép forrása.
    imNagyKép.Source = imKép.Source;
    // A kép alatt megjelenítjük a kép állomány nevét.
    tlKépnév.Text = imNagyKép.Source.ToString();
}
```

5. A bélyegképen történő egérgomb lenyomás eseménykezelője

```
/// <summary>
/// Egérgomb lenyomása egy kép vezérlőn.
/// </summary>
/// <param name="sender">A kép vezérlő.</param>
/// <param name="e">Kiegészítő paraméterek.</param>
void Kép_MouseDown(object sender, MouseButtonEventArgs e)
{
    // Beállítjuk a képet a nagy vezérlőben láthatónak.
    KépBeállít((Image)sender);
}
```

2. Eseménykezelő a mappaválasztáshoz

A mappaválasztó gombon történt kattintást követően létrehozunk egy példányt a könyvtárválasztó párbeszédablakból (FolderPickerDialog), beállítjuk a kezdő könyvtárat az ablak fejlécében tárolt útvonalnak megfelelően, majd megjelenítjük a párbeszédablakot.

Amennyiben a felhasználó az OK gombbal zárja be az ablakot, akkor a kiválasztott útvonalat átmásoljuk az ablak fejlécébe, és beolvassuk a memóriába az adott mappában található képeket.

```
/// <summary>
/// A képeket tartalmazó könyvtár kiválasztása.
/// </summary>
/// <param name="sender">Az eseményt előidéző nyomógomb.</param>
/// <param name="e">Kiegészítő paraméterek.</param>
private void btKönyvtárVálasztó_Click(object sender, RoutedEventArgs e)
{
  // Mappaválasztó párbeszédablak objektum létrehozása.
  var dlg = new FolderPickerDialog();
  // Kezdőkönyvtár beállítása.
  dlg.InitialPath = Title;
  // Párbeszédablak megjelenítése.
  if (dlg.ShowDialog() == true)
  {
     // A mappa elérési útvonalának átmásolása az ablak fejlécébe.
     Title = dlg.SelectedPath;
     // A képek betöltése a kiválasztott mappából, és elhelyezésük kép
     // vezérlők formájában a WrapPanel-en.
     KépeketBetölt();
```

} }

Házi feladat

Alakítsa át úgy a programot, hogy a nagyméretű kép alatt csak a képállomány neve jelenjen meg az elérési útfonal és a file:/// felirat nélkül.

Helyezzen el egy nyomógombot a nagyméretű kép alatt "Vágólapra másol" felirattal, és készítse el hozzá az eseménykezelőt.

Ellenőrizze le, hogy az eseménykezelő által a vágólapra másolt kép beilleszthető-e egy Word dokumentumba.