

Ejercicio 3, 4 y 5

Diseño de Interfaces : Diego Santos Bailon
IES Pere Maríó Orts I Bosch
Benidorm, Alicante
Leobardo Salvador Ríos Suárez
15/10/2024
2º de DAM I

Contenido

| | |
|------------------|---|
| Ejercicio 3..... | 1 |
| Maquetación..... | 2 |
| Webgrafía..... | 3 |
| Cuestión..... | 3 |
| Ejercicio 4..... | 4 |
| Maquetación..... | 4 |
| Cuestión..... | 5 |
| Ejercicio 5..... | 6 |
| Maquetación..... | 6 |
| Cuestión..... | 6 |

EJERCICIO 3.

Ten en cuenta los siguientes puntos:

- Cambia el nombre de la ventana y archivo o proyecto a "Ejercicio 3"
- La ventana principal es de 400x500, partiendo de un Grid
- El grid está dividido en 2 filas justo por la mitad (aunque agrandáramos la ventana, la división seguiría estando a la mitad)
- Las áreas roja y amarilla son ambas de tipo StackPanel (cada una con una orientación)
- La anchura de los botones A1..A5 es 60, y la altura de B1..B5 es automática
- Todas las letras están en negrita, tamaño 14 y fuente Consolas

PRÁCTICA 0

Maquetación

Titulo y tamaño de la ventana principal:

```
Title="Ejercicio 3" Height="400" Width="450 ">
```

Grid dividido en 2:

- RowDefinition utilice * que permite distribuir el espacio disponible proporcionalmente entre varias filas.

```
<Grid>
  <Grid.RowDefinitions>
    <RowDefinition Height="*"></RowDefinition>
    <RowDefinition Height="*"></RowDefinition>
  </Grid.RowDefinitions>
</Grid>
```

Orientación y botones

- El primer panel tiene orientación horizontal y el segundo vertical.
- Los botones tienen el tamaño predeterminado por defecto.
- El **FontWeight** es **Bold**. Le da el peso a la letra.
- **FontSize** de 14 determinando el tamaño de la letra.
- **FontFamily** que **Consolas** que determina el estilo.
- El **StackPanel** amarillo tiene la propiedad **Grid.Row** igual a 0 para dejarlo en la primera mitad de la aplicación. El **StackPanel** rojo esta en **Grid.Column** igual a 1 para que se coloque en la segunda parte.

```
<StackPanel Grid.Row="0" Background="Yellow" Orientation="Horizontal">
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas">Boton A1</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas">Boton A2</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas">Boton A3</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas">Boton A4</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas">Boton A5</Button>
</StackPanel>
<StackPanel Grid.Row="1" Background="Blue" Orientation="Vertical">
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas">Boton B1</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas">Boton B2</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas">Boton B3</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas">Boton B4</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas">Boton B5</Button>
</StackPanel>
```

PRÁCTICA 0

Webgrafía

- <https://wpf-tutorial.com/es/26/paneles/el-control-stackpanel/>
- <https://chatgpt.com/>

Cuestión

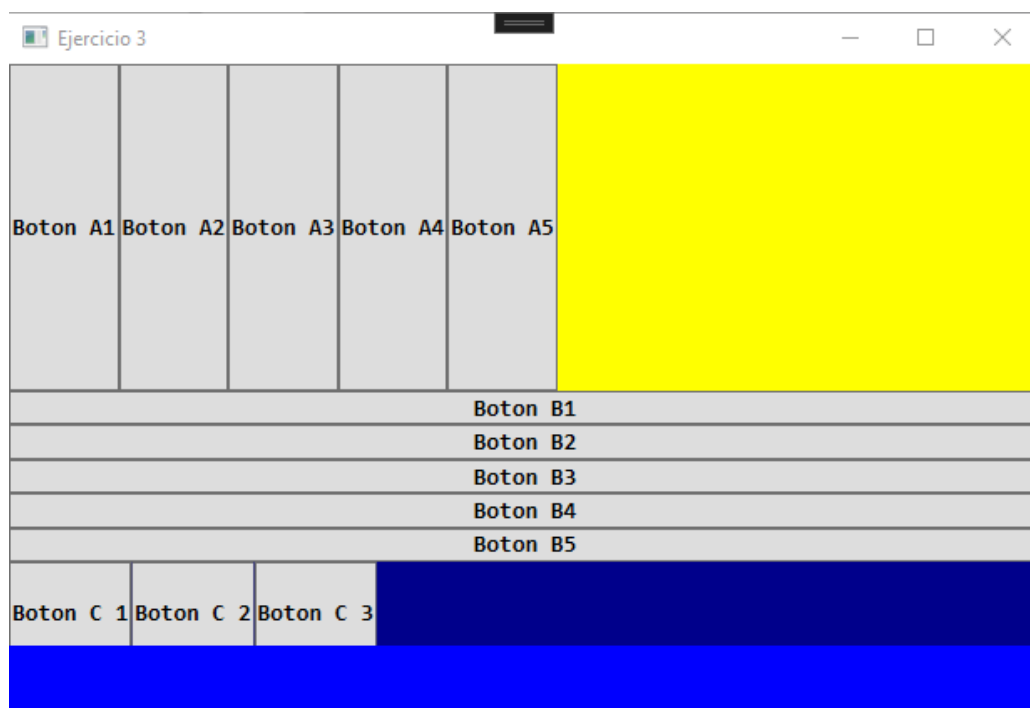
Una vez terminado, añadí el StackPanel inferior (debajo de B5) un StackPanel de altura 50 y Orientación horizontal.

Lo pinté de azul y metí dentro 3 botones, llamados C1...C3· (de altura 60 y mismo tipo de letra que los botones anteriores).

Estos 3 ultimos botones, si miramos el código XAML.

1. ¿Están al mismo “nivel” que los otros? ¿Porque?

Estan un nivel más debajo de los B`s porque estan contenidos dentro de un StackPanel el cual si esta al mismo nivel de los demas botones B`s.



PRÁCTICA 0

EJERCICIO 4

Ten en cuenta los siguientes puntos:

- Al igual que antes partimos de un grid (esta vez de 400x400) dividido en 2 filas justo por la mitad.
- Las áreas roja y amarilla son, en esta ocasión, de tipo WrapPanel.
- La anchura de los botones A1...A4 es 80, y la anchura de B1...B4 es 200.
- Todas las letras tienen la misma fuente que en el ejercicio 3.

Maquetación

Titulo y tamaño de la ventana principal:

```
Title="MainWindow" Height="400" Width="400">
```

Grid dividido en 2:

- RowDefinition en * que permite distribuir el espacio disponible proporcionalmente entre varias filas.

```
<Grid>
  <Grid.RowDefinitions>
    <RowDefinition Height="*"></RowDefinition>
    <RowDefinition Height="*"></RowDefinition>
  </Grid.RowDefinitions>
</Grid>
```

WrapPanel y botones:

- Lo único que cambia aquí es el contenedor, los demás elementos tienen las mismas propiedades.

```
<WrapPanel Grid.Row="0" Background="Red">
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas" Width="80">Botón 1</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas" Width="80">Botón 2</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas" Width="80">Botón 3</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas" Width="80">Botón 4</Button>
</WrapPanel>
<WrapPanel Grid.Row="1" Background="Yellow">
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas" Width="200">Botón 1</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas" Width="200">Botón 2</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas" Width="200">Botón 3</Button>
  <Button FontWeight="Bold" FontSize="14" FontFamily="Consolas" Width="200">Botón 4</Button>
</WrapPanel>
```

PRÁCTICA 0

Cuestión

Una vez terminado, añadi el WrapPanel inferior (debajo de B4) un StackPanel de altura 20 y Orientación horizontal.

Lo pinte de verde y metí dentro 2 botones, llamados BotonC1 y BotonC2 (de altura 80 y mismo tipo de letra que los botones anteriores).

Cuando ejecutamos y reducimos la altura de la ventana.

1. ¿Pasan a la siguiente columna de uno en uno o a la vez?

Pasan a la siguiente columna de uno en uno.

2. ¿A que es debido?

Debido a la propiedad del WrapPanel.



PRÁCTICA 0

EJERCICIO 5

Ten en cuenta los siguientes puntos:

- Al igual que antes partimos de un grid (esta vez de 400x400) dividido en 2 filas justo por la mitad.
- Las áreas roja y amarilla son, en esta ocasión, de tipo WrapPanel.
- La anchura de los botones A1...A4 es 80, y la anchura de B1...B4 es 200.
- Todas las letras tienen la misma fuente que en el ejercicio 3.

Maquetación

Titulo y tamaño de la ventana principal:

```
Title="Ejercicio 3" Height="600" Width="600">
```

Grid dividido en 2 columnas y 3 Filas:

- Al ponerle Height alas columnas y filas abarcan todo el espacio disponible.

```
<Grid.ColumnDefinitions>
  <ColumnDefinition/>
  <ColumnDefinition/>
</Grid.ColumnDefinitions>
<Grid.RowDefinitions>
  <RowDefinition/>
  <RowDefinition/>
  <RowDefinition/>
</Grid.RowDefinitions>
```

Botones:

```
<Button>Boton 1</Button>
<Button Grid.Column="1">Boton 2</Button>
<Button Grid.Row="1">Boton 3</Button>
<Button Grid.Column="1" Grid.Row="1">Boton 4</Button>
<Button Grid.Row="2">Boton 5</Button>
<Button Grid.Column="1" Grid.Row="2">Boton 6</Button>
```

Cuestión

Si miras el código XAML de los & elementos veras que sólo en 3 de ellos aparece tanto Grid.Column como Grid.Row.

En dos de ellos solo aparece una de estas propiedades, y en 1 elemento no aparece ninguna de las dos.

PRÁCTICA 0

1. ¿Por qué ocurre esto?

Porque el valor por default de un elemento dentro de una distribución Grid tanto como en una columna o fila es 0.

2. Si fueras al código XAML de las dos columnas que tienes, y les pusieras 2* y 1* respectivamente. ¿Qué pasaría?

Los elementos situados en Grid.Column 0 su Width tienen 2/3 partes de la ventana.

