Siatka (Grid)

Siatka jest prawdopodobnie jednym z najbardziej złożonych paneli. Może ona posiadać wiele wierszy i kolumn, których szerokość lub wysokość może być określona w wartościach bezwzględnych, w postaci procentu dostępnego miejsca lub w sposób automatyczny – gdzie wiersz lub kolumna będą samodzielnie dostosowywać swój rozmiar w zależności od zawartości. Używajcie siatki tam, gdzie inne panele nie są w stanie sobie poradzić – czyli wtedy, gdy potrzeba wielu kolumn często wraz z innymi panelami.

W najprostszej formie siatka po prostu weźmie kontrolki wewnątrz niej i rozciągnie je na całe dostępne miejsca nakładając jedną na drugą.

```
<Grid>
    <Button>Przycisk 1</Button>
    <Button>Przycisk 2</Button>
</Grid>
```



Jak widać, ostatnia kontrolka znajduje się na samej górze, co, w tym przypadku, powoduje, że nie można zobaczyć pierwszego przycisku. Spróbujmy podzielić dostępną przestrzeń. Można to zrobić za pomocą właściwości ColumnDefinitions oraz RowDefinitions, jednak w pierwszym przykładzie ograniczymy się tylko do kolumn.

```
<Grid>
      <Grid.ColumnDefinitions>
          <ColumnDefinition Width="*" />
          <ColumnDefinition Width="*" />

      </Grid.ColumnDefinitions>
      <Button Grid.Column="0">Przycisk 1</Button>
      <Button Grid.Column="1">Przycisk 2</Button>

  </Grid>
```



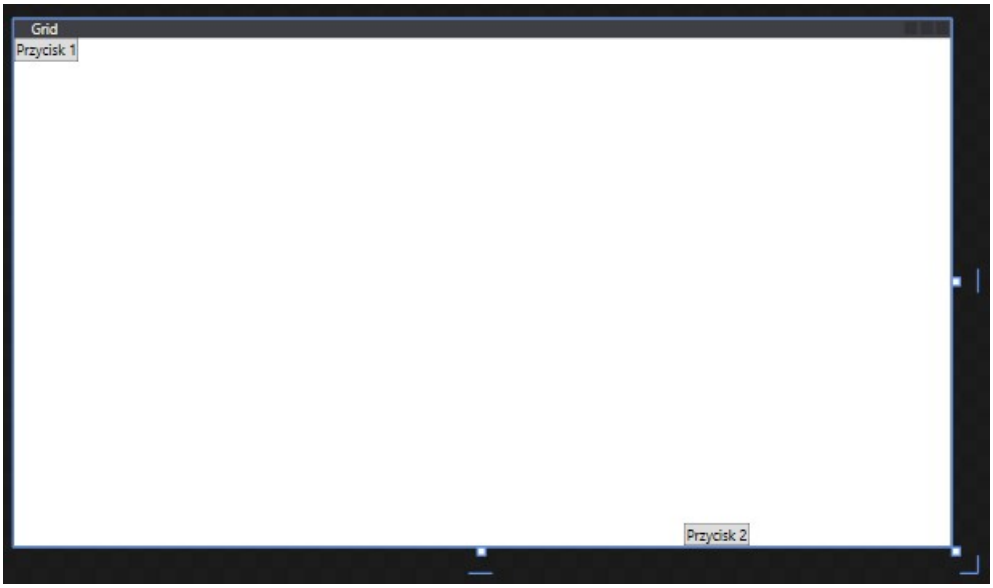
W tym przykładzie po prostu podzieliliśmy dostępne miejsce pomiędzy dwie kolumny o dokładnie takiej samej szerokości z wykorzystaniem „szerokości gwiazdkowej”. 0 odpowiada pierwszej kolumnie, 1 odpowiada drugiej kolumnie, i tak dalej.

Jak widać, przyciski zajęły całą dostępną przestrzeń, co jest domyślnym zachowaniem, kiedy siatka układa podlegające jej kontrolki. Dzieje się tak przez właściwości przycisków HorizontalAlignment (wyrównanie w poziomie) oraz VerticalAlignment (wyrównanie w pionie) ustawione na Rozciągnij (Stretch).

W niektórych sytuacjach możecie chcieć, aby kontrolki zajmowały tylko tyle miejsca ile potrzebują niezależnie od ich ułożenia w siatce. Najłatwiejszym sposobem aby to osiągnąć jest ustawienie właściwości **HorizontalAlignment** oraz **VerticalAlignment** bezpośrednio na kontrolkach, które chcecie zmienić. Sprawdźmy to działanie:

```
  <Grid>
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="*" />
            <ColumnDefinition Width="*" />

        </Grid.ColumnDefinitions>
        <Button Grid.Column="0" VerticalAlignment="Top"
HorizontalAlignment="Left">Przycisk 1</Button>
        <Button Grid.Column="1"  VerticalAlignment="Bottom"
HorizontalAlignment="Center">Przycisk 2</Button>
    </Grid>
```



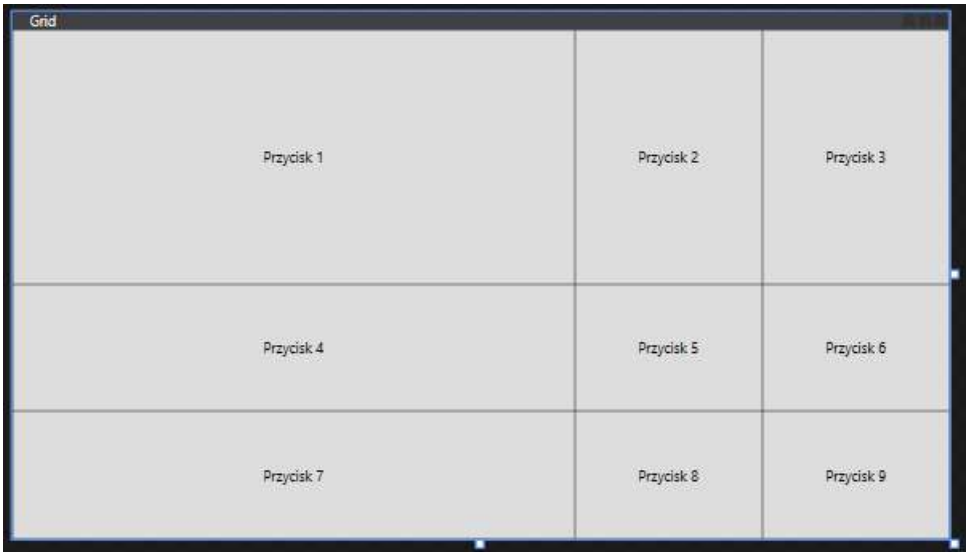
Jak widać, pierwszy przycisk jest teraz wyrównany do góry i wyrównany do lewej strony w poziomie względem kolumny. Drugi przycisk jest wycentrowany w pionie i przyciągnięty do dolnej strony kolumny.

Siatka (Grid) - wiersze i kolumny

Wcześniej przedstawiliśmy panel oraz pokazaliśmy kilka podstawowych przykładów jej użycia. Tym razem zrobimy parę bardziej zaawansowanych układów – czyli to, w czym siatka sprawuje się najlepiej. Najpierw stwórzmy prawdziwie tabelaryczny układ z wykorzystaniem większej ilości kolumn i kilku wierszy:

```
<Grid>
    <Grid.ColumnDefinitions>
        <ColumnDefinition Width="3*" />
        <ColumnDefinition Width="1*" />
        <ColumnDefinition Width="1*" />

    </Grid.ColumnDefinitions>
    <Grid.RowDefinitions>
        <RowDefinition Height="2*"/>
        <RowDefinition Height="1*"/>
        <RowDefinition Height="1*"/>
    </Grid.RowDefinitions>
    <Button Grid.Column="0" Grid.Row="0">Przycisk 1</Button>
    <Button Grid.Column="1" Grid.Row="0">Przycisk 2</Button>
    <Button Grid.Column="2" Grid.Row="0">Przycisk 3</Button>
    <Button Grid.Column="0" Grid.Row="1">Przycisk 4</Button>
    <Button Grid.Column="1" Grid.Row="1">Przycisk 5</Button>
    <Button Grid.Column="2" Grid.Row="1">Przycisk 6</Button>
    <Button Grid.Column="0" Grid.Row="2">Przycisk 7</Button>
    <Button Grid.Column="1" Grid.Row="2">Przycisk 8</Button>
    <Button Grid.Column="2" Grid.Row="2">Przycisk 9</Button>
</Grid>
```



W sumie mamy dziewięć przycisków, każdy w swojej własnej komórce siatki posiadającej trzy wiersze i trzy kolumny. Znowu użyliśmy szerokości opartej na gwiazdce, jednak tym razem dodaliśmy również liczbę – pierwszy wiersz i kolumna mają wysokość (szerokość) równą 3*, co znaczy tyle, że mają użyć trzy razy tyle miejsca ile wiersze i kolumny z wysokością (szerokością) równą 1* (lub po prostu *, bez liczby, co znaczy dokładnie to samo).

Możecie zauważyć, że użyłem właściwości dołączonych Grid.Row i Grid.Column, po to, aby umieścić kontrolki wewnątrz siatki, pamiętajcie, że można pominąć te właściwości przy kontrolkach, co do których chcę aby były w pierwszym wierszu lub kolumnie (lub obu). Pominięcie tych właściwości znaczy tyle samo co przypisanie do nich zera. Pozwala to zaoszczędzić trochę pisania, jednak jeśli chcemy możemy przypisywać te wartości do wszystkich kontrolek.