

Ejercicios resueltos en Xamarin Forms

Armando Roman Gallardo

aroman@ucol.mx

Contenido

Suma de dos números que utiliza los componentes Label, Entry y Button.....	2
Calculadora utilizando radio buttons para seleccionar las operaciones.....	5
Calculadora que utiliza el componente CheckBox Nativo incluido en Xamarin Forms 4.....	8

Suma de dos números que utiliza los componentes Label, Entry y Button

- **Diseño de la interfaz**

XML

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    xmlns:d="http://xamarin.com/schemas/2014/forms/design"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    mc:Ignorable="d"
    x:Class="PMSuma.MainPage">

    <StackLayout>
        <!-- Place new controls here -->
        <Label Text="Primer valor"
            Margin="5"
            />
        <Entry x:Name="valor1"
            Margin="5"/>
        <Label Text="Segundo valor"
            Margin="5"
            />
        <Entry x:Name="valor2"
            Margin="5"/>

        <Button Text="Sumar" Margin="5" Clicked="Button_Clicked"/>
        <Label x:Name="resultado"
            Text="Resultado"
            Margin="5" FontAttributes="Bold"/>

    </StackLayout>

</ContentPage>
```

- Así se ve la app



- Código fuente

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Xamarin.Forms;

namespace PMSuma
{
    // Learn more about making custom code visible in the Xamarin.Forms previewer
    // by visiting https://aka.ms/xamarinforms-previewer
    [DesignTimeVisible(false)]
    public partial class MainPage : ContentPage
    {
        public MainPage()
        {
            InitializeComponent();
        }

        private void Button_Clicked(object sender, EventArgs e)
        {
            float a = float.Parse(valor1.Text);
        }
    }
}
```

```
float b = float.Parse(valor2.Text);  
float c= a + b;  
resultado.Text = a + "+" + b + "=" + c;  
    }  
}
```

Calculadora utilizando radio buttons para seleccionar las operaciones.

Xamarin forms trata de utilizar mayormente un numero de controles nativos, en este caso en la plataforma de IOS no existen los radios buttons por lo que se pueden implementar de manera manual o bajar un Nuget que nos permita utilizarlos en todas las plataformas.

- Instala el nuguet Xamarin.Forms.InputKit
- Diseña la interfaz

XAML

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             xmlns:d="http://xamarin.com/schemas/2014/forms/design"
             xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
             xmlns:input="clr-namespace:Plugin.InputKit.Shared.Controls;assembly=Plugin.InputKit"
             mc:Ignorable="d"
             x:Class="PMCalculadoraRadioButton.MainPage">

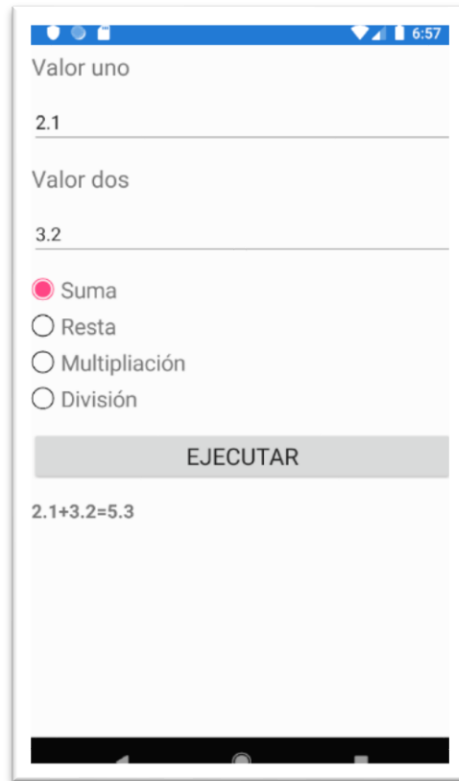
    <StackLayout>
        <!-- Place new controls here -->
        <Label Text="Valor uno"
              Margin="5"
              FontSize="Large"/>
        <Entry x:Name="valor1"
              Margin="5"
              FontSize="Medium"
              Keyboard="Numeric"/>
        <Label Text="Valor dos"
              Margin="5"
              FontSize="Large"/>
        <Entry x:Name="valor2"
              Margin="5"
              FontSize="Medium"
              Keyboard="Numeric"/>

        <input:RadioButtonGroupView Margin="5" x:Name="operacion">
            <input:RadioButton Text="Suma"/>
            <input:RadioButton Text="Resta" />
            <input:RadioButton Text="Multipliación" />
            <input:RadioButton Text="División" />
        </input:RadioButtonGroupView>
        <Button Text="Ejecutar" Margin="5" FontSize="Large"
              Clicked="Button_Clicked"/>
        <Label x:Name="resultado" Margin="5" FontSize="Medium"
              FontAttributes="Bold"/>

    </StackLayout>

</ContentPage>
```

- Así se ve la app



- Código fuente

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Xamarin.Forms;

namespace PMCalculadoraRadioButton
{
    // Learn more about making custom code visible in the Xamarin.Forms previewer
    // by visiting https://aka.ms/xamarinforms-previewer
    [DesignTimeVisible(false)]
    public partial class MainPage : ContentPage
    {
        public MainPage()
        {
            InitializeComponent();
        }

        private void Button_Clicked(object sender, EventArgs e)
        {

```

```

float a = float.Parse(valor1.Text);
float b = float.Parse(valor2.Text);
float c;

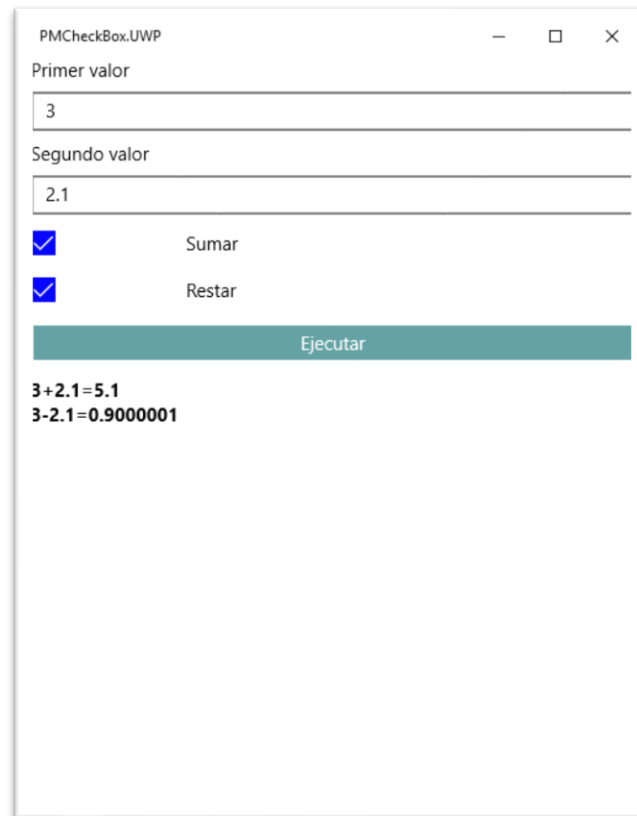
switch (operacion.SelectedIndex)
{
    case -1:
    {
        resultado.Text = "No selecciono ninguna operacion";
        break;
    }
    case 0:
    {
        c = a + b;
        resultado.Text = a+" "+b+"="+c;
        break;
    }
    case 1:
    {
        c = a - b;
        resultado.Text = a + "-" + b + "=" + c;
        break;
    }
    case 2:
    {
        c = a * b;
        resultado.Text = a + "X" + b + "=" + c;
        break;
    }
    case 3:
    {
        c = a / b;
        resultado.Text = a + "/" + b + "=" + c;
        break;
    }
}
}
}
}
}

```

Calculadora que utiliza el componente CheckBox Nativo incluido en Xamarin Forms 4

Este Ejercicio nos muestra una calculadora de dos operaciones que utiliza el control Xamarin Forms CheckBox nativo a las aplicaciones tanto IOS, como Android por lo que no hay que incluir algún componente externo.

- **Así se ve la App**



- **XML**

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    xmlns:d="http://xamarin.com/schemas/2014/forms/design"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    mc:Ignorable="d"
    x:Class="PMCheckBox.MainPage">

    <StackLayout>
        <!-- Place new controls here -->
        <Label Text="Primer valor"
            Margin="5,1"
            />
        <Entry x:Name="valor1"
            Margin="5,1"/>
        <Label Text="Segundo valor"
```



```

        Margin="5,1"
    />
    <Entry x:Name="valor2"
        Margin="5,1"/>
    <StackLayout Orientation="Horizontal" Margin="5,0">
        <CheckBox x:Name="c1" />
        <Label Text="Sumar" VerticalOptions="Center"/>
    </StackLayout>
    <StackLayout Orientation="Horizontal" Margin="5,0">
        <CheckBox x:Name="c2" />
        <Label Text="Restar" VerticalOptions="Center"/>
    </StackLayout>
    <Button Text="Ejecutar" Margin="5"
        Clicked="Button_Clicked" BackgroundColor="CadetBlue"
        TextColor="White"/>
    <Label x:Name="resultado"
        Text="Resultado"
        FontAttributes="Bold"
        Margin="5,1"/>
</StackLayout>

</ContentPage>

```

- **Código Fuente**

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Xamarin.Forms;

namespace PMCheckBox
{
    // Learn more about making custom code visible in the Xamarin.Forms previewer
    // by visiting https://aka.ms/xamarinforms-previewer
    [DesignTimeVisible(false)]
    public partial class MainPage : ContentPage
    {
        public MainPage()
        {
            InitializeComponent();
        }

        private void Button_Clicked(object sender, EventArgs e)
        {
            try
            {
                String s = "";
                float a = float.Parse(valor1.Text);
                float b = float.Parse(valor2.Text);

                if (c1.IsChecked == true)
                {
                    s = s + a + "+" + b + "=" + (a + b) + "\n";
                }
            }
        }
    }
}

```

```

    }

    if (c2.IsChecked == true)
    {
        s = s + a + "-" + b + "=" + (a - b) + "\n";
    }
    resultado.Text = s;
} catch
{
    resultado.Text = "Valores novalidos!!";
}
}
}
}
}

```