

HCIM SUMMER WORKSHOP

Introduction to C#

.NET

.NET is:

- Microsoft's Platform for Windows Development
- CLR (Common Language Runtime) – the Virtual Machine that runs MSIL (Microsoft Intermediate Language Code)
- A set of compilers that can generate MSIL
 - C#, Visual Basic, C++ etc..
- This provides inter-operability

C#

Originated by Microsoft as a response to Java

Software robustness

- strong type checking, array bounds checking, detection of use of uninitialized variables, source code portability, automatic garbage collection
- allows rapid application development

Unsafe shortcuts for speedy

- Pointers and stuff

Developed specifically for .NET

Very easy to migrate from C++ and Java

HELLO WORLD TIME!

Console Version

EVENTS

An event in C# is a way for a class to provide notifications to clients of that class when some interesting thing happens to an object.

The most familiar use for events is in graphical user interfaces; typically, the classes that represent controls in the interface have events that are notified when the user does something to the control (for example, click a button).

Events, however, need not be used only for graphical interfaces. Events provide a generally useful way for objects to signal state changes that may be useful to clients of that object.

DELEGATES

Events are declared using delegates.

A delegate object encapsulates a method so that it can be called anonymously.

An event is a way for a class to allow clients to give it delegates to methods that should be called when the event occurs.

When the event occurs, the delegate(s) given to it by its clients are invoked.

HELLO WORLD TIME! MKII

Windows Forms Application

WINDOWS PRESENTATION FOUNDATION (WPF)

Windows Presentation Foundation (or **WPF**) is a graphical subsystem for rendering user interfaces in Windows-based applications

WPF attempts to provide a consistent programming model for building applications and provides a separation between the user interface and the inner logic.

WPF employs XAML, a derivative of XML, to define and link various UI elements.

WPF utilizes DirectX.

GRAPHICAL COMPONENTS

Designer View



Code View

```
HyperlinkButton t1 = new HyperlinkButton();
t1.FontSize = 28;
t1.Content = ImportantValues.questionsLocal[rand.Next(0,5)];
t1.Width = Double.NaN;
t1.HorizontalAlignment = System.Windows.HorizontalAlignment.Left;
t1.Click += new RoutedEventHandler(t1_Click);
scrollView1.Content = t1;

HyperlinkButton t2 = new HyperlinkButton();
t2.FontSize = 28;
t2.Content = ImportantValues.questionsTime[rand.Next(0, 5)];
t2.Width = Double.NaN;
t2.HorizontalAlignment = System.Windows.HorizontalAlignment.Left;
t2.Click += new RoutedEventHandler(t1_Click);
scrollView2.Content = t2;

HyperlinkButton t3 = new HyperlinkButton();
t3.FontSize = 28;
t3.Content = ImportantValues.questionsGeneral[rand.Next(0, 3)];
t3.Width = Double.NaN;
t3.HorizontalAlignment = System.Windows.HorizontalAlignment.Left;
t3.Click += new RoutedEventHandler(t1_Click);
scrollView3.Content = t3;

HyperlinkButton t4 = new HyperlinkButton();
t4.FontSize = 28;
t4.Content = ImportantValues.questionsWho[rand.Next(0, 5)];
t4.Width = Double.NaN;
t4.HorizontalAlignment = System.Windows.HorizontalAlignment.Left;
t4.Click += new RoutedEventHandler(t1_Click);
scrollView4.Content = t4;
```

XAML View

```
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
xmlns:phone="clr-namespace:Microsoft.Phone.Controls;assembly=Microsoft.Phone"
xmlns:shell="clr-namespace:Microsoft.Phone.Shell;assembly=Microsoft.Phone"
xmlns:id="http://schemas.microsoft.com/expression/blend/2008"
xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
mc:Ignorable="d" d:DesignWidth="480" d:DesignHeight="768"
FontFamily="{StaticResource PhoneFontFamilyNormal}"
FontSize="{StaticResource PhoneFontSizeNormal}"
Foreground="{StaticResource PhoneForegroundBrush}"
SupportedOrientations="Portrait" Orientation="Portrait"
shell:SystemTray.IsVisible="True"/>

<!--LayoutRoot is the root grid where all page content is placed-->
<Grid x:Name="LayoutRoot" Background="Transparent">
    <Grid.RowDefinitions>
        <RowDefinition Height="Auto"/>
        <RowDefinition Height="*" />
    </Grid.RowDefinitions>

    <!--TitlePanel contains the name of the application and page title-->
    <StackPanel x:Name="TitlePanel" Grid.Row="0" Margin="12,17,0,28">
        <TextBlock x:Name="ApplicationTitle" Text="XisQuê" Style="{StaticResource PhoneTextNormal}" />
        <TextBlock x:Name="PageTitle" Text="Insira uma questão" Margin="9,-7,0,0" Style="{StaticResource PhoneTextNormal}" />
    </StackPanel>

    <!--ContentPanel - place additional content here-->
    <Grid x:Name="ContentPanel" Grid.Row="1" Margin="12,0,12,0">
        <Grid.RowDefinitions>
            <RowDefinition Height="174*" />
            <RowDefinition Height="Auto" />
        </Grid.RowDefinitions>
        <TextBox Height="96" HorizontalAlignment="Left" Margin="0,10,0,0" Name="textBox1" Text="" />
        <Button Background="#009BFFFF" Content="Perguntar" Height="86" HorizontalAlignment="Left" Margin="0,0,0,0" Name="button1" />
        <Button Background="#009BFFFF" Content="Limpar" Height="86" HorizontalAlignment="Left" Margin="250,124,0,0" Name="button2" />
        <Grid Height="301" Margin="0,52,0,0" Name="grid1" VerticalAlignment="Top" ShowGridLines="False">
            <Grid.RowDefinitions>
                <RowDefinition Height="Auto" />
                <RowDefinition Height="72*" />
                <RowDefinition Height="73*" />
                <RowDefinition Height="74*" />
                <RowDefinition Height="82*" />
            </Grid.RowDefinitions>
            <ScrollView Grid.Row="1" Height="72" HorizontalAlignment="Left" Name="scrollView1" />
            <ScrollView Grid.Row="2" Height="73" HorizontalAlignment="Left" Name="scrollView2" />
            <ScrollView Grid.Row="3" Height="74" HorizontalAlignment="Left" Name="scrollView3" />
            <ScrollView Grid.Row="4" Height="76" HorizontalAlignment="Left" Name="scrollView4" />
        </Grid>
    </Grid>
</Grid>
```

VARIABLES

Primitive types pretty much like Java.

Safe type conversion provided by class `Convert`.

Getters and Setters:

```
public int c;  
  
public int C  
{  
    get { return c; }  
    set { c = value; }  
}
```

```
public int A { get; set; }  
public int B { get; set; }
```

GUI COMPONENTS & EVENTS

Try out time!

USEFUL LIBRARIES I

System

- namespace contains fundamental classes and base classes that define commonly-used value and reference data types, events and event handlers, interfaces, attributes, and processing exceptions.

System.Collections.Generic

- contains interfaces and classes that define generic collections (list, dictionary etc.), which allow users to create strongly typed collections that provide better type safety and performance than non-generic strongly typed collections.

System.IO

- contain types that support input and output, including the ability to read and write data to streams, to compress data in streams, to create and use isolated stores, to map files to an application's logical address space, to store multiple data objects in a single container, to communicate using anonymous or named pipes, to implement custom logging, and to handle the flow of data to and from serial ports.

USEFUL LIBRARIES II

System.Windows

- namespaces contain types used in Windows Presentation Foundation (WPF) applications, including animation clients, user interface controls, data binding, and type conversion.
- System.Windows.Forms and its child namespaces are used for developing Windows Forms applications.

System.Threading

- contain types that enable multithreaded programming. A child namespace provides types that simplify the work of writing concurrent and asynchronous code.

System.Data

- contain classes for accessing and managing data from diverse sources. The top-level namespace and a number of the child namespaces together form the ADO.NET architecture and ADO.NET data providers. For example, providers are available for SQL Server, Oracle, ODBC, and OleDb.