Министерство образования и науки Украины

Харьковский национальный экономический университет

Кафедра информационных систем

Отчет

о лабораторной работе №1

по дисциплине «Мобильные технологии»

на тему : «**Дослідження інструментальних засобів розробки прикладень**

**для мобільних пристроїв**»

Выполнила:

студентка 2 курса 4 группы

факультета ЭИ

Филоненко Татьяны

Проверил:

доцент кафедры ИС

Поляков А.А.

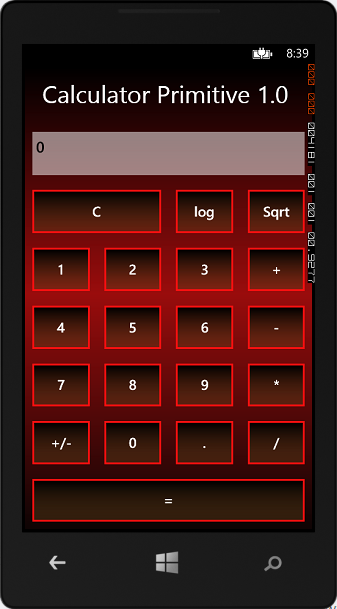
г. Харьков

2013

**Цель:** получить основные практические навыки разработки приложений для мобильных устройств в современных инструментальных средах, исследовать основные особенности инструментальных сред разработки мобильных приложений, изучить архитектуру проектов.

Часть 1

Разработка на Windows Phone 8



<phone:PhoneApplicationPage

x:Class="Lab1\_win\_phone.MainPage"

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:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

mc:Ignorable="d"

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" Margin="5">

<Grid.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#FF9B0C0C" Offset="0.49"/>

<GradientStop Color="#FF0E0404" Offset="1"/>

</LinearGradientBrush>

</Grid.Background>

<Grid.ColumnDefinitions>

<ColumnDefinition />

<ColumnDefinition />

<ColumnDefinition />

<ColumnDefinition />

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition />

<RowDefinition />

<RowDefinition />

<RowDefinition />

<RowDefinition />

<RowDefinition />

<RowDefinition />

<RowDefinition />

</Grid.RowDefinitions>

<!-- LOCALIZATION NOTE:

To localize the displayed strings copy their values to appropriately named

keys in the app's neutral language resource file (AppResources.resx) then

replace the hard-coded text value between the attributes' quotation marks

with the binding clause whose path points to that string name.

For example:

Text="{Binding Path=LocalizedResources.ApplicationTitle, Source={StaticResource LocalizedStrings}}"

This binding points to the template's string resource named "ApplicationTitle".

Adding supported languages in the Project Properties tab will create a

new resx file per language that can carry the translated values of your

UI strings. The binding in these examples will cause the value of the

attributes to be drawn from the .resx file that matches the

CurrentUICulture of the app at run time.

-->

<!--TitlePanel contains the name of the application and page title-->

<TextBlock x:Name="tb1" Text="Сalculator Primitive 1.0 " Grid.Row="0" Grid.ColumnSpan="4" Style="{StaticResource PhoneTextNormalStyle}" VerticalAlignment="Center" HorizontalAlignment="Center" FontFamily="Forte" FontSize="40"/>

<!--ContentPanel - place additional content here-->

<TextBox Grid.Row="1" Grid.ColumnSpan="4" Name="tb" Text="0" FontFamily="Segoe WP Semibold" SelectionBackground="#FFFF3420" BorderBrush="#7FFFFFFF" Background="#7FFFFFFF"></TextBox>

<Button x:Name="bC" Grid.Row="2" Grid.ColumnSpan="2" Content="C" Click="bC\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="b1" Grid.Row="3" Grid.Column="0" Content="1" Click="b1\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="b2" Grid.Row="3" Grid.Column="1" Content="2" Click="b2\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="b3" Grid.Row="3" Grid.Column="2" Content="3" Click="b3\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="bPlus" Grid.Row="3" Grid.Column="3" Content="+" Click="bPlus\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="bLog" Grid.Row="2" Grid.Column="2" Content="log" Click="bLog\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="bSqrt" Grid.Row="2" Grid.Column="3" Content="Sqrt" Click="bSqrt\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="b4" Grid.Row="4" Grid.Column="0" Content="4" Click="b4\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="b5" Grid.Row="4" Grid.Column="1" Content="5" Click="b5\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="b6" Grid.Row="4" Grid.Column="2" Content="6" Click="b6\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="bMinus" Grid.Row="4" Grid.Column="3" Content="-" Click="bMinus\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="b7" Grid.Row="5" Grid.Column="0" Content="7" Click="b7\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="b8" Grid.Row="5" Grid.Column="1" Content="8" Click="b8\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="b9" Grid.Row="5" Grid.Column="2" Content="9" Click="b9\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="bMultiply" Grid.Row="5" Grid.Column="3" Content="\*" Click="bMultiply\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="bPM" Grid.Row="6" Grid.Column="0" Content="+/-" Click="bPM\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="b0" Grid.Row="6" Grid.Column="1" Content="0" Click="b0\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="bDot" Grid.Row="6" Grid.Column="2" Content="." Click="bDot\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="bDivide" Grid.Row="6" Grid.Column="3" Content="/" Click="bDivide\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button x:Name="bEq" Grid.Row="7" Grid.ColumnSpan="4" Content="=" Click="bEq\_Click" BorderBrush="#FFFF1010" Foreground="White">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Color="Black" Offset="0"/>

<GradientStop Color="#685D421A" Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

</Grid>

</phone:PhoneApplicationPage>

Код

using System;

using System.Collections.Generic;

using System.Linq;

using System.Net;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Navigation;

using Microsoft.Phone.Controls;

using Microsoft.Phone.Shell;

using Lab1\_win\_phone.Resources;

namespace Lab1\_win\_phone

{

public partial class MainPage : PhoneApplicationPage

{

Boolean isDouble, isSetDouble, isRezult, isAdd, isSub, isDiv, isMul, isEq, isPow, isEdit;

Double Number;

public MainPage()

{

InitializeComponent();

isDouble = false;

isSetDouble = false;

isRezult = false;

isAdd = false;

isSub = false;

isDiv = false;

isMul = false;

isEq = false;

isPow = false;

isEdit = true;

Number = 0.0;

}

private void bC\_Click(object sender, RoutedEventArgs e)

{

isDouble = false;

isSetDouble = false;

tb.Text = "0";

isDouble = false;

isSetDouble = false;

isRezult = false;

isAdd = false;

isSub = false;

isDiv = false;

isMul = false;

isEq = false;

isPow = false;

Number = 0.0;

}

private void b1\_Click(object sender, RoutedEventArgs e)

{

if (((isAdd) || (isSub) || (isDiv) || (isMul) || (isEq) || (isPow)) && (!(tb.Text=="0") && (!isEdit)))

{

tb.Text="0";

isDouble = false;

isEdit = true;

}

if ((isDouble) && (!isSetDouble))

{

tb.Text+=".";

isSetDouble = true;

}

if (tb.Text == "0")

{

tb.Text = "1";

}

else

tb.Text += "1";

}

private void b2\_Click(object sender, RoutedEventArgs e)

{

if (((isAdd) || (isSub) || (isDiv) || (isMul) || (isEq) || (isPow)) && (!(tb.Text == "0") && (!isEdit)))

{

tb.Text = "0";

isDouble = false;

isEdit = true;

}

if ((isDouble) && (!isSetDouble))

{

tb.Text += ".";

isSetDouble = true;

}

if (tb.Text == "0")

{

tb.Text = "2";

}

else

tb.Text += "2";

}

private void b3\_Click(object sender, RoutedEventArgs e)

{

if (((isAdd) || (isSub) || (isDiv) || (isMul) || (isEq) || (isPow)) && (!(tb.Text == "0") && (!isEdit)))

{

tb.Text = "0";

isDouble = false;

isEdit = true;

}

if ((isDouble) && (!isSetDouble))

{

tb.Text += ".";

isSetDouble = true;

}

if (tb.Text == "0")

{

tb.Text = "3";

}

else

tb.Text += "3";

}

private void b4\_Click(object sender, RoutedEventArgs e)

{

if (((isAdd) || (isSub) || (isDiv) || (isMul) || (isEq) || (isPow)) && (!(tb.Text == "0") && (!isEdit)))

{

tb.Text = "0";

isDouble = false;

isEdit = true;

}

if ((isDouble) && (!isSetDouble))

{

tb.Text += ".";

isSetDouble = true;

}

if (tb.Text == "0")

{

tb.Text = "4";

}

else

tb.Text += "4";

}

private void b5\_Click(object sender, RoutedEventArgs e)

{

if (((isAdd) || (isSub) || (isDiv) || (isMul) || (isEq) || (isPow)) && (!(tb.Text == "0") && (!isEdit)))

{

tb.Text = "0";

isDouble = false;

isEdit = true;

}

if ((isDouble) && (!isSetDouble))

{

tb.Text += ".";

isSetDouble = true;

}

if (tb.Text == "0")

{

tb.Text = "5";

}

else

tb.Text += "5";

}

private void b6\_Click(object sender, RoutedEventArgs e)

{

if (((isAdd) || (isSub) || (isDiv) || (isMul) || (isEq) || (isPow)) && (!(tb.Text == "0") && (!isEdit)))

{

tb.Text = "0";

isDouble = false;

isEdit = true;

}

if ((isDouble) && (!isSetDouble))

{

tb.Text += ".";

isSetDouble = true;

}

if (tb.Text == "0")

{

tb.Text = "6";

}

else

tb.Text += "6";

}

private void b7\_Click(object sender, RoutedEventArgs e)

{

if (((isAdd) || (isSub) || (isDiv) || (isMul) || (isEq) || (isPow)) && (!(tb.Text == "0") && (!isEdit)))

{

tb.Text = "0";

isDouble = false;

isEdit = true;

}

if ((isDouble) && (!isSetDouble))

{

tb.Text += ".";

isSetDouble = true;

}

if (tb.Text == "0")

{

tb.Text = "7";

}

else

tb.Text += "7";

}

private void b8\_Click(object sender, RoutedEventArgs e)

{

if (((isAdd) || (isSub) || (isDiv) || (isMul) || (isEq) || (isPow)) && (!(tb.Text == "0") && (!isEdit)))

{

tb.Text = "0";

isDouble = false;

isEdit = true;

}

if ((isDouble)&&(!isSetDouble))

{

tb.Text += ".";

isSetDouble = true;

}

if (tb.Text == "0")

{

tb.Text = "8";

}

else

tb.Text += "8";

}

private void b9\_Click(object sender, RoutedEventArgs e)

{

if (((isAdd) || (isSub) || (isDiv) || (isMul) || (isEq) || (isPow)) && (!(tb.Text == "0") && (!isEdit)))

{

tb.Text = "0";

isDouble = false;

isEdit = true;

}

if ((isDouble) && (!isSetDouble))

{

tb.Text += ".";

isSetDouble = true;

}

if (tb.Text == "0")

{

tb.Text = "9";

}

else

tb.Text += "9";

}

private void b0\_Click(object sender, RoutedEventArgs e)

{

if (((isAdd) || (isSub) || (isDiv) || (isMul) || (isEq) || (isPow)) && (!(tb.Text == "0") && (!isEdit)))

{

tb.Text = "0";

isDouble = false;

isEdit = true;

}

if ((isDouble) && (!isSetDouble))

{

tb.Text += ".";

isSetDouble = true;

}

if (tb.Text == "0")

{

tb.Text = "0";

}

else

tb.Text += "0";

}

private void bPM\_Click(object sender, RoutedEventArgs e)

{

tb.Text = Convert.ToString(Convert.ToDouble(tb.Text) \* (-1));

}

private void bDot\_Click(object sender, RoutedEventArgs e)

{

if (!isSetDouble)

{

isDouble = true;

}

}

private void bEq\_Click(object sender, RoutedEventArgs e)

{

isDouble=false;

isEdit=false;

isSetDouble=false;

if ((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))

{

if (isAdd) Number=Number+(Convert.ToDouble(tb.Text));

if (isSub) Number=Number-(Convert.ToDouble(tb.Text));

if (isDiv) Number=Number/(Convert.ToDouble(tb.Text));

if (isMul) Number=Number\*(Convert.ToDouble(tb.Text));

if (isPow) Number=Math.Pow(Number,(Convert.ToDouble(tb.Text)));

tb.Text=Number.ToString();

}

isAdd=false;

isSub=false;

isDiv=false;

isMul=false;

isPow=false;

isEq=false;

Number=Convert.ToDouble(tb.Text);

}

private void bPlus\_Click(object sender, RoutedEventArgs e)

{

isDouble=false;

isEdit=false;

isSetDouble=false;

if ((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))

{

if (isAdd) Number=Number+(Convert.ToDouble(tb.Text));

if (isSub) Number=Number-(Convert.ToDouble(tb.Text));

if (isDiv) Number=Number/(Convert.ToDouble(tb.Text));

if (isMul) Number=Number\*(Convert.ToDouble(tb.Text));

if (isPow) Number=Math.Pow(Number,Convert.ToDouble(tb.Text));

tb.Text=Number.ToString();

}

isAdd=true;

isSub=false;

isDiv=false;

isMul=false;

isPow=false;

isEq=false;

Number=Convert.ToDouble(tb.Text);

}

private void bMinus\_Click(object sender, RoutedEventArgs e)

{

isDouble=false;

isEdit=false;

isSetDouble=false;

if ((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))

{

if (isAdd) Number=Number+(Convert.ToDouble(tb.Text));

if (isSub) Number=Number-(Convert.ToDouble(tb.Text));

if (isDiv) Number=Number/(Convert.ToDouble(tb.Text));

if (isMul) Number=Number\*(Convert.ToDouble(tb.Text));

if (isPow) Number=Math.Pow(Number,Convert.ToDouble(tb.Text));

tb.Text=Number.ToString();

}

isAdd=false;

isSub=true;

isDiv=false;

isMul=false;

isPow=false;

isEq=false;

Number=Convert.ToDouble(tb.Text);

}

private void bMultiply\_Click(object sender, RoutedEventArgs e)

{

isDouble=false;

isEdit=false;

isSetDouble=false;

if ((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))

{

if (isAdd) Number=Number+(Convert.ToDouble(tb.Text));

if (isSub) Number=Number-(Convert.ToDouble(tb.Text));

if (isDiv) Number=Number/(Convert.ToDouble(tb.Text));

if (isMul) Number=Number\*(Convert.ToDouble(tb.Text));

if (isPow) Number=Math.Pow(Number,Convert.ToDouble(tb.Text));

tb.Text=Number.ToString();

}

isAdd=false;

isSub=false;

isDiv=false;

isMul=true;

isPow=false;

isEq=false;

Number=Convert.ToDouble(tb.Text);

}

private void bDivide\_Click(object sender, RoutedEventArgs e)

{

isDouble=false;

isEdit=false;

isSetDouble=false;

if ((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))

{

if (isAdd) Number=Number+(Convert.ToDouble(tb.Text));

if (isSub) Number=Number-(Convert.ToDouble(tb.Text));

if (isDiv) Number=Number/(Convert.ToDouble(tb.Text));

if (isMul) Number=Number\*(Convert.ToDouble(tb.Text));

if (isPow) Number=Math.Pow(Number,Convert.ToDouble(tb.Text));

tb.Text=Number.ToString();

}

isAdd=false;

isSub=false;

isDiv=true;

isMul=false;

isPow=false;

isEq=false;

Number=Convert.ToDouble(tb.Text);

}

private void bLog\_Click(object sender, RoutedEventArgs e)

{

if (Convert.ToDouble(tb.Text) <= 0)

MessageBox.Show("Логарифм не найден");

else

{

tb.Text = Convert.ToString(Math.Log(Convert.ToDouble(tb.Text)));

isRezult = true;

}

}

private void bSqrt\_Click(object sender, RoutedEventArgs e)

{

if (Convert.ToDouble(tb.Text) < 0)

MessageBox.Show("Корень невозможно найти");

else

{

tb.Text = Convert.ToString(Math.Sqrt(Convert.ToDouble(tb.Text)));

isRezult = true;

}

}

}

}

**Часть 2**

**Android**

package edu.khneu.mt.calculation;

import android.R.integer;

import android.app.Activity;

import android.os.Bundle;

import android.util.Log;

import android.view.Menu;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.EditText;

public class MainActivity extends Activity {

Button b0,b1,b2,b3,b4,b5,b6,b7,b8,b9;

EditText field;

Boolean isDouble,isSetDouble,isRezult,isAdd,isSub,isDiv,isMul,isEq,isPow,isEdit;

Double Number;

Button b\_plus,b\_minus,b\_divide,b\_multiple,b\_pow,b\_equal,b\_double,b\_sqrt,b\_plus\_minus,b\_clear;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.myscreen);

b0=(Button)findViewById(R.id.button10);

b1=(Button)findViewById(R.id.button7);

b2=(Button)findViewById(R.id.button8);

b3=(Button)findViewById(R.id.button9);

b4=(Button)findViewById(R.id.button4);

b5=(Button)findViewById(R.id.button5);

b6=(Button)findViewById(R.id.button6);

b7=(Button)findViewById(R.id.button1);

b8=(Button)findViewById(R.id.button2);

b9=(Button)findViewById(R.id.button3);

b\_double=(Button)findViewById(R.id.button11);

b\_clear=(Button)findViewById(R.id.button\_c);

b\_sqrt=(Button)findViewById(R.id.button\_sqrt);

b\_plus\_minus=(Button)findViewById(R.id.button\_plus\_minus);

field=(EditText)findViewById(R.id.field1);

// operation buttons

b\_plus=(Button)findViewById(R.id.button12);

b\_minus=(Button)findViewById(R.id.button\_minus);

b\_divide=(Button)findViewById(R.id.button\_devide);

b\_multiple=(Button)findViewById(R.id.button\_multiple);

b\_pow=(Button)findViewById(R.id.button\_pow);

b\_equal=(Button)findViewById(R.id.button13);

isDouble=false;

isSetDouble=false;

isRezult=false;

isAdd=false;

isSub=false;

isDiv=false;

isMul=false;

isEq=false;

isPow=false;

isEdit=true;

Number=0.0;

OnClickListener Click1=new OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

if (((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))&&(!field.getText().toString().equals("0"))&&(!isEdit))

{

field.setText(null);

field.append("0");

isDouble=false;

isEdit=true;

}

if (isDouble)

{field.append(".");

isSetDouble=true;}

if (field.getText().toString().equals("0"))

{

field.setText(null);

field.append("1");

}

else

field.append("1");

}

};

b1.setOnClickListener(Click1);

OnClickListener Click2=new OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

if (((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))&&(!field.getText().toString().equals("0"))&&(!isEdit))

{

field.setText(null);

field.append("0");

isDouble=false;

isEdit=true;

}

if (isDouble)

{field.append(".");

isSetDouble=true;}

if (field.getText().toString().equals("0"))

{

field.setText(null);

field.append("2");

}

else

field.append("2");

}

};

b2.setOnClickListener(Click2);

OnClickListener Click3=new OnClickListener() {

@Override

public void onClick(View v) {

if (((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))&&(!field.getText().toString().equals("0"))&&(!isEdit))

{

field.setText(null);

field.append("0");

isDouble=false;

isEdit=true;

}

if (isDouble)

{field.append(".");

isSetDouble=true;}

// TODO Auto-generated method stub

if (field.getText().toString().equals("0"))

{

field.setText(null);

field.append("3");

}

else

field.append("3");

}

};

b3.setOnClickListener(Click3);

OnClickListener Click4=new OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

if (((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))&&(!field.getText().toString().equals("0"))&&(!isEdit))

{

field.setText(null);

field.append("0");

isDouble=false;

isEdit=true;

}

if (isDouble)

{field.append(".");

isSetDouble=true;}

if (field.getText().toString().equals("0"))

{

field.setText(null);

field.append("4");

}

else

field.append("4");

}

};

b4.setOnClickListener(Click4);

OnClickListener Click5=new OnClickListener() {

@Override

public void onClick(View v) {

if (((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))&&(!field.getText().toString().equals("0"))&&(!isEdit))

{

field.setText(null);

field.append("0");

isDouble=false;

isEdit=true;

}

if (isDouble)

{field.append(".");

isSetDouble=true;}

// TODO Auto-generated method stub

if (field.getText().toString().equals("0"))

{

field.setText(null);

field.append("5");

}

else

field.append("5");

}

};

b5.setOnClickListener(Click5);

OnClickListener Click6=new OnClickListener() {

@Override

public void onClick(View v) {

if (((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))&&(!field.getText().toString().equals("0"))&&(!isEdit))

{

field.setText(null);

field.append("0");

isDouble=false;

isEdit=true;

}

if (isDouble)

{field.append(".");

isSetDouble=true;}

// TODO Auto-generated method stub

if (field.getText().toString().equals("0"))

{

field.setText(null);

field.append("6");

}

else

field.append("6");

}

};

b6.setOnClickListener(Click6);

OnClickListener Click7=new OnClickListener() {

@Override

public void onClick(View v) {

if (((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))&&(!field.getText().toString().equals("0"))&&(!isEdit))

{

field.setText(null);

field.append("0");

isDouble=false;

isEdit=true;

}

if (isDouble)

{field.append(".");

isSetDouble=true;}

// TODO Auto-generated method stub

if (field.getText().toString().equals("0"))

{

field.setText(null);

field.append("7");

}

else

field.append("7");

}

};

b7.setOnClickListener(Click7);

OnClickListener Click8=new OnClickListener() {

@Override

public void onClick(View v) {

if (((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))&&(!field.getText().toString().equals("0"))&&(!isEdit))

{

field.setText(null);

field.append("0");

isDouble=false;

isEdit=true;

}

if (isDouble)

{field.append(".");

isSetDouble=true;}

// TODO Auto-generated method stub

if (field.getText().toString().equals("0"))

{

field.setText(null);

field.append("8");

}

else

field.append("8");

}

};

b8.setOnClickListener(Click8);

OnClickListener Click9=new OnClickListener() {

@Override

public void onClick(View v) {

if (((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))&&(!field.getText().toString().equals("0"))&&(!isEdit))

{

field.setText(null);

field.append("0");

isDouble=false;

isEdit=true;

}

if (isDouble)

{field.append(".");

isSetDouble=true;}

// TODO Auto-generated method stub

if (field.getText().toString().equals("0"))

{

field.setText(null);

field.append("9");

}

else

field.append("9");

}

};

b9.setOnClickListener(Click9);

OnClickListener Click0=new OnClickListener() {

@Override

public void onClick(View v) {

if (((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))&&(!field.getText().toString().equals("0"))&&(!isEdit))

{

field.setText(null);

field.append("0");

isDouble=false;

isEdit=true;

}

if (isDouble)

{field.append(".");

isSetDouble=true;}

// TODO Auto-generated method stub

if (field.getText().toString().equals("0"))

{

field.setText(null);

field.append("0");

}

else

field.append("0");

}

};

b0.setOnClickListener(Click0);

OnClickListener Click\_Clear=new OnClickListener() {

@Override

public void onClick(View v) {

isDouble=false;

isSetDouble=false;

field.setText(null);

field.append("0");

isDouble=false;

isSetDouble=false;

isRezult=false;

isAdd=false;

isSub=false;

isDiv=false;

isMul=false;

isEq=false;

isPow=false;

Number=0.0;

}

};

b\_clear.setOnClickListener(Click\_Clear);

OnClickListener Click\_Double=new OnClickListener() {

@Override

public void onClick(View v) {

if (!isSetDouble)

//TODO Auto-generated method stub

{

isDouble=true;

}

}

};

b\_double.setOnClickListener(Click\_Double);

OnClickListener Click\_Sqrt=new OnClickListener() {

@Override

public void onClick(View v) {

Double d=new Double(field.getText().toString());

d=Math.sqrt(d);

double rez =Math.round(d);

if (rez==d)

{field.setText(null);

int x = d.intValue();

field.append(Integer.toString(x));}

else{field.setText(null);

field.append(d.toString());}

isRezult=true;

}

};

b\_sqrt.setOnClickListener(Click\_Sqrt);

OnClickListener Click\_Plus\_Minus=new OnClickListener() {

@Override

public void onClick(View v) {

Double d=new Double(field.getText().toString());

d=d\*(-1);

double rez =Math.round(d);

if (rez==d)

{field.setText(null);

int x = d.intValue();

field.append(Integer.toString(x));

}

else

{

field.setText(null);

field.append(d.toString());

}

}

};

b\_plus\_minus.setOnClickListener(Click\_Plus\_Minus);

OnClickListener Click\_Plus=new OnClickListener() {

@Override

public void onClick(View v) {

isDouble=false;

isEdit=false;

isSetDouble=false;

if ((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))

{

if (isAdd) Number=Number+(new Double(field.getText().toString()));

if (isSub) Number=Number-(new Double(field.getText().toString()));

if (isDiv) Number=Number/(new Double(field.getText().toString()));

if (isMul) Number=Number\*(new Double(field.getText().toString()));

if (isPow) Number=Math.pow(Number,(new Double(field.getText().toString())));

double rez =Math.round(Number);

if (rez==Number)

{

field.setText(null);

int x = Number.intValue();

field.append(Integer.toString(x));

}

else

{

field.setText(null);

field.append(Number.toString());

}

}

isAdd=true;

isSub=false;

isDiv=false;

isMul=false;

isPow=false;

isEq=false;

Number=new Double(field.getText().toString());

}

};

b\_plus.setOnClickListener(Click\_Plus);

OnClickListener Click\_Minus=new OnClickListener() {

@Override

public void onClick(View v) {

isDouble=false;

isEdit=false;

isSetDouble=false;

if ((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))

{

if (isAdd) Number=Number+(new Double(field.getText().toString()));

if (isSub) Number=Number-(new Double(field.getText().toString()));

if (isDiv) Number=Number/(new Double(field.getText().toString()));

if (isMul) Number=Number\*(new Double(field.getText().toString()));

if (isPow) Number=Math.pow(Number,(new Double(field.getText().toString())));

double rez =Math.round(Number);

if (rez==Number)

{

field.setText(null);

int x = Number.intValue();

field.append(Integer.toString(x));

}

else

{

field.setText(null);

field.append(Number.toString());

}

}

isAdd=false;

isSub=true;

isDiv=false;

isMul=false;

isPow=false;

isEq=false;

Number=new Double(field.getText().toString());

}

};

b\_minus.setOnClickListener(Click\_Minus);

OnClickListener Click\_Maltiple=new OnClickListener() {

@Override

public void onClick(View v) {

isDouble=false;

isEdit=false;

isSetDouble=false;

if ((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))

{

if (isAdd) Number=Number+(new Double(field.getText().toString()));

if (isSub) Number=Number-(new Double(field.getText().toString()));

if (isDiv) Number=Number/(new Double(field.getText().toString()));

if (isMul) Number=Number\*(new Double(field.getText().toString()));

if (isPow) Number=Math.pow(Number,(new Double(field.getText().toString())));

double rez =Math.round(Number);

if (rez==Number)

{

field.setText(null);

int x = Number.intValue();

field.append(Integer.toString(x));

}

else

{

field.setText(null);

field.append(Number.toString());

}

}

isAdd=false;

isSub=false;

isDiv=false;

isMul=true;

isPow=false;

isEq=false;

Number=new Double(field.getText().toString());

}

};

b\_multiple.setOnClickListener(Click\_Maltiple);

OnClickListener Click\_Divide=new OnClickListener() {

@Override

public void onClick(View v) {

isDouble=false;

isEdit=false;

isSetDouble=false;

if ((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))

{

if (isAdd) Number=Number+(new Double(field.getText().toString()));

if (isSub) Number=Number-(new Double(field.getText().toString()));

if (isDiv) Number=Number/(new Double(field.getText().toString()));

if (isMul) Number=Number\*(new Double(field.getText().toString()));

if (isPow) Number=Math.pow(Number,(new Double(field.getText().toString())));

double rez =Math.round(Number);

if (rez==Number)

{

field.setText(null);

int x = Number.intValue();

field.append(Integer.toString(x));

}

else

{

field.setText(null);

field.append(Number.toString());

}

}

isAdd=false;

isSub=false;

isDiv=true;

isMul=false;

isPow=false;

isEq=false;

Number=new Double(field.getText().toString());

}

};

b\_divide.setOnClickListener(Click\_Divide);

OnClickListener Click\_Pow=new OnClickListener() {

@Override

public void onClick(View v) {

isDouble=false;

isEdit=false;

isSetDouble=false;

if ((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))

{

if (isAdd) Number=Number+(new Double(field.getText().toString()));

if (isSub) Number=Number-(new Double(field.getText().toString()));

if (isDiv) Number=Number/(new Double(field.getText().toString()));

if (isMul) Number=Number\*(new Double(field.getText().toString()));

if (isPow) Number=Math.pow(Number,(new Double(field.getText().toString())));

double rez =Math.round(Number);

if (rez==Number)

{

field.setText(null);

int x = Number.intValue();

field.append(Integer.toString(x));

}

else

{

field.setText(null);

field.append(Number.toString());

}

}

isAdd=false;

isSub=false;

isDiv=false;

isMul=false;

isPow=true;

isEq=false;

Number=new Double(field.getText().toString());

}

};

b\_pow.setOnClickListener(Click\_Pow);

OnClickListener Click\_Eq=new OnClickListener() {

@Override

public void onClick(View v) {

isDouble=false;

isEdit=false;

isSetDouble=false;

if ((isAdd)||(isSub)||(isDiv)||(isMul)||(isEq)||(isPow))

{

if (isAdd) Number=Number+(new Double(field.getText().toString()));

if (isSub) Number=Number-(new Double(field.getText().toString()));

if (isDiv) Number=Number/(new Double(field.getText().toString()));

if (isMul) Number=Number\*(new Double(field.getText().toString()));

if (isPow) Number=Math.pow(Number,(new Double(field.getText().toString())));

double rez =Math.round(Number);

if (rez==Number)

{

field.setText(null);

int x = Number.intValue();

field.append(Integer.toString(x));

}

else

{

field.setText(null);

field.append(Number.toString());

}

}

isAdd=false;

isSub=false;

isDiv=false;

isMul=false;

isPow=false;

isEq=false;

Number=new Double(field.getText().toString());

}

};

b\_equal.setOnClickListener(Click\_Eq);

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

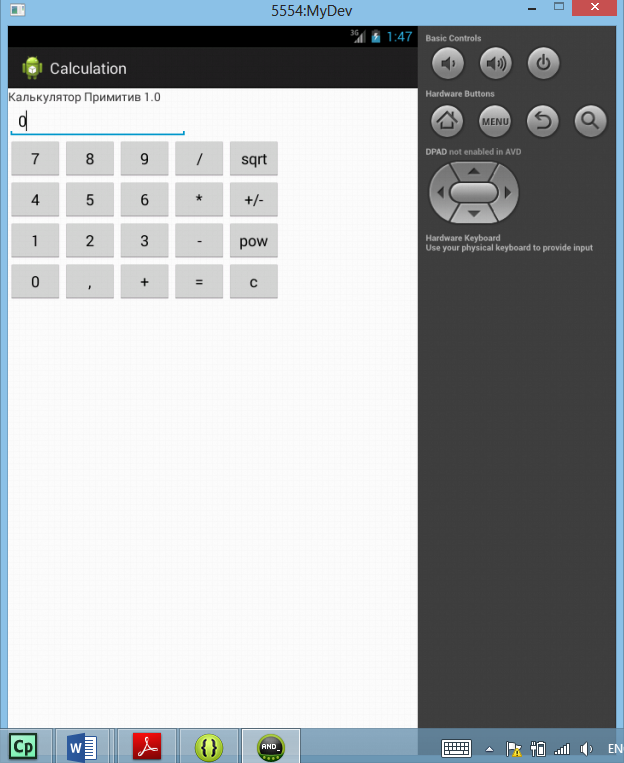
// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.main, menu);

return true;

}

}



<?xml version=*"1.0"* encoding=*"utf-8"*?>

<LinearLayout

xmlns:android=*"http://schemas.android.com/apk/res/android"*

android:orientation=*"vertical"*

android:layout\_width=*"match\_parent"*

android:layout\_height=*"match\_parent"*>

<TextView

android:id=*"@+id/textView1"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

android:text=*"@string/Program\_Name"*>

</TextView>

<EditText

android:id=*"@+id/field1"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

android:ems=*"10"*

android:text=*"@string/field1"*

android:inputType=*"number|numberSigned|numberDecimal"* >

<requestFocus />

</EditText>

<TableLayout

android:layout\_height=*"wrap\_content"*

android:layout\_width=*"match\_parent"*

android:id=*"@+id/tableLayout1"*>

<TableRow

android:id=*"@+id/tableRow1"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*>

<Button

android:text=*"@string/button1"*

android:id=*"@+id/button1"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button2"*

android:id=*"@+id/button2"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button3"*

android:id=*"@+id/button3"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button\_divide"*

android:id=*"@+id/button\_devide"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button\_sqrt"*

android:id=*"@+id/button\_sqrt"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

</TableRow>

<TableRow

android:id=*"@+id/tableRow2"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*>

<Button

android:text=*"@string/button4"*

android:id=*"@+id/button4"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button5"*

android:id=*"@+id/button5"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button6"*

android:id=*"@+id/button6"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button\_multiple"*

android:id=*"@+id/button\_multiple"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button\_plus\_minus"*

android:id=*"@+id/button\_plus\_minus"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

</TableRow>

<TableRow

android:id=*"@+id/tableRow3"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*>

<Button

android:text=*"@string/button7"*

android:id=*"@+id/button7"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button8"*

android:id=*"@+id/button8"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button9"*

android:id=*"@+id/button9"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button\_minus"*

android:id=*"@+id/button\_minus"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button\_pow"*

android:id=*"@+id/button\_pow"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

</TableRow>

<TableRow

android:id=*"@+id/tableRow4"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*>

<Button

android:text=*"@string/button10"*

android:id=*"@+id/button10"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button11"*

android:id=*"@+id/button11"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button12"*

android:id=*"@+id/button12"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button13"*

android:id=*"@+id/button13"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

<Button

android:text=*"@string/button\_clear"*

android:id=*"@+id/button\_c"*

android:layout\_width=*"wrap\_content"*

android:layout\_height=*"wrap\_content"*

>

</Button>

</TableRow>

</TableLayout>

</LinearLayout>

**Вывод:** на практической работы я исследовала инструментальные средства разработки приложений для мобильных устройств и создала приложения калькуляторы для Windows Phone и Android