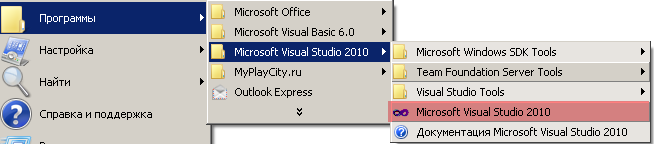
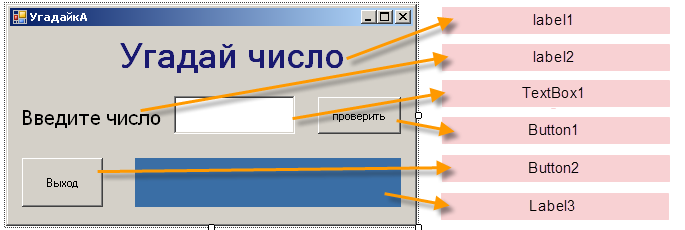
**Самостоятельная работа №5. Visual Basic 2010. Угадай число. Глобальные и локальные переменные и константы.**

Цель: Создадим проект калькулятора.

1. Запустите Microsoft Visual Studio 2010.



1. Создадим новый проект .
2. Создадим новую форму .



1. **Внимательно проверим все названия элементов.**
2. Сегодня мы изучим, такие понятия, как Константы и Переменные.

**Константы** - это величины, значение которых не меняется.

**Переменные** используются в программе для хранения информации. В любой момент времени в переменную можно записать новое значение. Переменные используются для хранения не только введенной пользователем информации, но и результатов вычислений в программе.

В Visual Basic можно использовать константы и переменные множества разных типов. Эти типы встроены в платформу и могут использоваться всеми языками.  
В платформе есть несколько **основных типов переменных**, которые иногда называют базовыми: **String** (текстовые строки), **Integer** (целые числа), **Single** (вещественные числа) и **Boolean** (логические, принимающие значения True или False).

Перед использованием переменной в программе, эту переменную нужно **объявить** (описать). Чтобы объявить переменную, нужно написать строку кода, в которой указывается имя этой переменной и ее тип. Объявление переменных ускоряет компиляцию программы и повышает эффективность ее работы. Кроме того, оно предотвращает ошибки, связанные с неправильным написанием имен переменных в коде.

Чтобы объявить переменную в коде на Visual Basic, используется оператор **Dim**. Синтаксис этого оператора следующий:

**Dim ИмяПеременной As ТипПеременной**

Переменные можно объявлять как **глобальные** или **локальные**. Пока мы писали код только внутри обработчиков событий элементов управления (например, код кнопки). Переменные, которые мы объявим в обработчике, будут **локальными**. Значения локальных переменных можно считывать или записывать только в коде обработчика, в котором они объявлены (это их **область видимости**). К ним нельзя обратиться из другого места в коде формы (например, из другого обработчика). Значения локальных переменных сохраняются только во время выполнения кода обработчика. Когда обработчик заканчивает свое выполнение, значения его локальных переменных теряются.

Чтобы объявить локальные переменные в обработчике, поместите оператор Dim в начало кода обработчика.

До сих пор мы не использовали глобальных переменных. Значения глобальных переменных можно считывать и записывать из любого места кода **формы**. Доступ к ним есть у всех обработчиков событий. Значения глобальных переменных сохраняются, пока форма остается открытой.

Чтобы объявить глобальную переменную, нужно поместить оператор Dim в тело формы. Лучше всего объявлять глобальные переменные в коде формы сразу после строки:

Public Class Form1

1. Внесём в форму следующий код:

В самое начало формы, после Public Class Form1 добавим

**Dim ch As Integer**

Мы объявили программе **Глобальную** переменную **CH**. (Это будет число, которое задумает компьютер).

1. Дважды нажав на форму (в пустом месте), мы откроем код Обработчика открытия формы (т.е. именно здесь, мы описываем действия, которые выполнятся при запуске нашей формы.

**Randomize()**

**ch = Int(Rnd() \* 99 + 1)**

Данный код программы, записывает в переменную **CH** – случайное значение от 0 до 99 и прибавит к значению 1. Т.е. реально задуманное число будет от 1 до 100.

1. В кнопку «Проверить» внесём следующий код.

If TextBox1.Text <> "" And ch <> 0 Then

If CSng(TextBox1.Text) = ch Then

Label3.Text = "ВЫ УГАДАЛИ!"

MsgBox("Задумано число" + Str(ch))

Else

If CSng(TextBox1.Text) > ch Then

Label3.Text = "Задуманное число меньше"

Else

Label3.Text = "Задуманное число больше"

End If

End If

Else

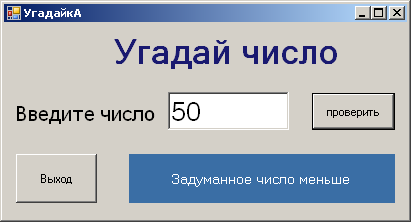
MsgBox("Введите число")

End If

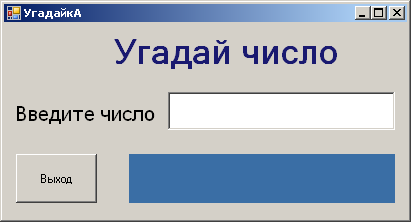
В кнопку «**выход**»:

END

1. **Запустим проект на выполнение**, щелкнув по кнопке **Start** на панели инструментов.

****

1. **Ваша оценка «4».**
2. **На оценку «5».** Внесите в программу изменение, чтобы проверка на «угадывание числа» выполнялась не по нажатию на кнопку «Проверить», а сразу после ввода числа в поле TextBox1.



**Спасибо за работу!**