**ПРАКТИЧЕСКАЯ РАБОТА 1**

**СОЗДАНИЕ ПЕРВОГО ПРИЛОЖЕНИЯ WINDOWS FORM**

***Цель: получение первичных навыков разработки приложений с использованием Windows Forms.***

Создать проект Windows Forms приложения Hello. Оно должно при нажатии в форме кнопки «Нажми» выводить на экран фразу «Hello, World and Россия <Фамилия студента>».

Варианты к заданию. Индивидуальные данные студента.

Пример. Создать проект Windows Forms приложения Hello. Оно должно при нажатии в форме кнопки «Нажми меня» выводить на экран фразу «Hello, World and Россия от Акчурина!».

Активизировать ИСР. В главном меню выбирается команда File=>New Project. Вызывается окно выбора типа проекта с набором шаблонов. В нем выбираем Приложение WindowsForms. Задаем имя проекта Hello.



Вид среды разработки меняется.



В центре в закладке Form1.cs[Конструктор] отображается окно Конструктора формы. Окно Панель элементов заполняется элементами для выбора. Окна Обозреватель решений содержит описание решения. Конструктор формы отображается по автоматически создаваемому коду (при желании его можно посмотреть двойным щелчком по Form1.cs => Program.cs в Обозревателе решений). Редактор кода модуля формы отображается командой Перейти к коду, которая находится в меню, выпадающем при щелчке по форме в конструкторе правой кнопкой мыши. Редактор отображается в закладке с именем Form1.cs.

Большая часть кода в Редакторе ИСР сделала автоматически. Нужно добавить функциональность.

Окна Конструктора и Редактора можно переключать кнопками в заголовках их закладок.

Теперь приступаем к проектированию в Конструкторе. Из окна Панели элементов перетаскиваем в форму объекты

Button1 - кнопка для запуска обработчика события. Выделяем объект, в окне свойств отображаются свойства кнопки. Свойству Text присваиваем значение - Нажми меня.

Label1 - метка, поле для отображения сообщения. Свойству Text присваиваем значение - Смотри сюда.



Для создания обработчика события щелчка по кнопке дважды щелкаем по кнопке в форме. Автоматически отображается окно Редактора, в котором в

код добавлен шаблон обработчика события button1\_Click, но без функциональности. Курсор устанавливается в место ввода кода, который будет задавать функциональность проекта.

Чтобы обработчик события срабатывал, нужно в окне свойств кнопки button1 в закладке событий выбрать реакцию на щелчок по кнопке из списка:



В шаблон кода, начиная с позиции курсора, нужно ввести инструкции. В примере свойству Text объекта label1 нужно присвоить строку ―Hello, world and Россия от меня!!‖. Чтобы исключить повторный доступ к кнопке, сделаем ее после вывода текста невидимой. Для этого вводим код

label1.Text= "Hello, world и Россия от меня!!"; button1.Visible = false;





Проект готов, проверим его командой Отладка=>Запуск без отладки. Получим окно приложения.



Проект готов, сохраняем его командой Файл=>Сохранить все в папке Hello. В результате проект сохраняется в структуре папок (чтобы увидеть все файлы, нужно активизировать кнопку, показанную на рисунке):



Решение включает:

|  |  |
| --- | --- |
| Решение Hello | Решение |
| Hello | Проект |
| Properties | Свойства |
| Ссылки | Ссылки |
| bin | Двоичные файлы |
| Debug | Файлы отладки |
| Hello.exe | Управляемый исполняемый файл |
| Hello.pdb | База данных для JIT компилятора |
| Hello.vshost.exe | Служебный файл |
| Release | Файлы выпуска |
| obj | Объектные файлы |

Исполняемые (bin) и объектные (obj) файлы образуются при компиляции (построении). Возможны два режима:

Команда Построение=>Построить решение. Построение в режиме отладки, в компонуемые файлы включаются символы отладки и режим оптимизации исключается. Это может увеличить размеры файлов. Файлы размещаются в папках Debug.

Команда Построение=>Перестроить решение. Построение отлаженного проекта, когда в компонуемые файлы символы отладки не включаются и компилятор использует режим оптимизации кода (например, исключает не использованные переменные). Это может уменьшить размеры файлов. Файлы размещаются в папках Release.