An Introduction to Programming Using
Visual Basic 2017

Chapter 1, 2

An Introduction to Computers and Problem Solving

Hardware and Software

- Hardware
  - physical components of a computer
- Software
  - set of instructions that tell computer what to do

Programming Languages

- Low level (machine language)
  - Hard for humans to understand (0s and 1s)
- High level (more English-like)
  - More easily understood by humans
  - Instructions like If … Then, Do While …, arithmetic statements

---

What is Programming?

- Problem solving
  - break problem down into steps
- Algorithm
  - set of step by step instructions
- Program
  - set of step by step instructions that tell computer what to do
  - written in a programming language
  - must be written so computer understands them
  - must follow set of syntax rules

Basic Programming Tasks

- <u>Declarations</u>
  - commands to tell computer what data is used in a program
- <u>Input</u> (raw data)
  - data entered into computer from user
- <u>Processing</u> (calculations, formatting, sorting …)
  - work done by the program
- <u>Output</u> (information)
  - results or answers

Following Directions

Computer
- does only what you tell it to do
- follows directions exactly
- not bored by repetition
- almost never makes a mistake

Programmer/User

- Analyst
  - person(s) that develops specifications, designs program
- Programmer
  - person(s) that creates algorithm, writes code, tests program
- User
  - person(s) that uses finished program
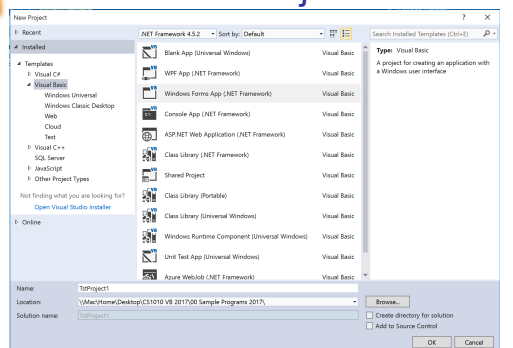- You will be both programmer and user

Visual Basic Programming

- Access Visual Basic
- Create a Visual Basic project
- Save a Visual Basic project
- Develop a Visual Basic project
- Run a Visual Basic program
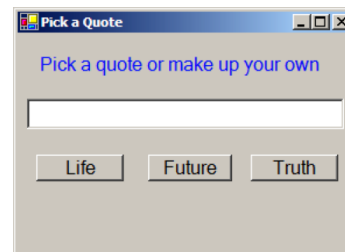- Review
- Debug a Program (correct errors)

Create a New Project

Sample VB Program

Running a Visual Basic Program

1.

2.

3.

Saving a Visual Basic Project

1.

2.

3.

Opening a Visual Basic Project

1.

2.

3.

Turning in a Visual Basic Project Electronically

1.

2.

3.

4.

Developing a Visual Basic Project

1. Design User Interface
   - Add controls
   - Set Properties
2. Determine events each control should recognize
3. Write Event Procedure Code
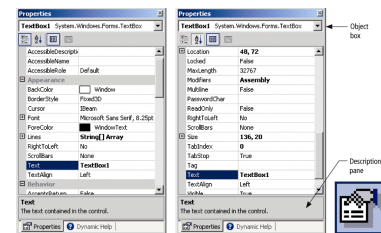
Set Properties

Categorized view   Alphabetical view
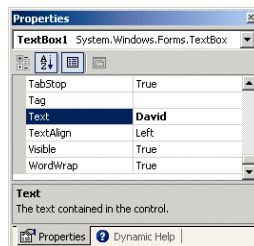
Set Properties

Set Name Property

- Name used to refer to a control in code
- Name must begin with a letter
- Must be less than 215 characters long
- May include numbers and the underscore
- Use appropriate 3 character naming prefix

3

Control Name Prefixes

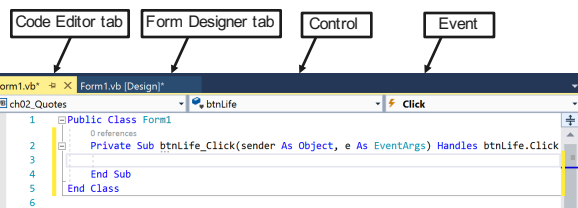| Control	Prefix	Example Setting
button	btn	btnCalcTotal
label	lbl	lblEnterAmount
text box	txt	txtAmount
list box	lst	lstOutput

Running a Visual Basic Program

- What happens when you run a Visual Basic program?
- What is an <u>event</u>?
- What is an <u>event procedure</u>?
- How do you create an event procedure?

Code Editor

Code Editor tab   Form Designer tab   Control   Event

```
Form1.vb*  ⊕ ×  Form1.vb [Design]*
VB ch02_Quotes              btnLife              Click
    1   Public Class Form1
            0 references
    2       Private Sub btnLife_Click(sender As Object, e As EventArgs) Handles btnLife.Click
    3
    4       End Sub
    5   End Class
    6
```

Changing Properties of Controls in Code

controlName.property = setting

Examples:

```
txtQuote.Text      = "I like life, it's something to do."

txtQuote.ForeColor = Color.DarkRed

txtQuote.TextAlign = HorizontalAlignment.Center
```
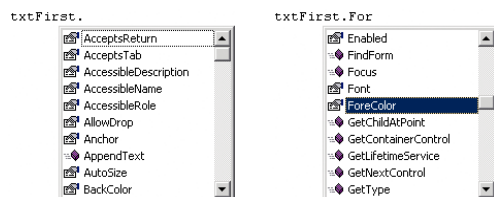
IntelliSense

Automatically pops up to give the programmer help.

```
txtFirst.                        txtFirst.For
    AcceptsReturn                    Enabled
    AcceptsTab                       FindForm
    AccessibleDescription            Focus
    AccessibleName                   Font
    AccessibleRole                   ForeColor
    AllowDrop                        GetChildAtPoint
    Anchor                           GetContainerControl
    AppendText                       GetLifetimeService
    AutoSize                         GetNextControl
    BackColor                        GetType
```

Review: Form

- Contains one or more controls
- <u>Text</u> property default is **Form1**
- Enter a more meaningful name
- Where does form's text property appear?
- DO NOT change (Name) property

Review: Button

- <u>Name</u> property prefix _____

- Where does <u>Text</u> property appear?

- What is most common event?

- Purpose

Review: Label

- <u>Name</u> property prefix _____

- What is most used property?

- Purpose?

- Usually placed to left of _____

Review: Text Box

- <u>Name</u> property prefix _____

- What is most used property?

- Purpose?

- Common events?

Program Errors (Bugs)

1. Syntax (Compile-time) Errors
2. Run-time Errors
3. Logic Errors

Debugging - process of removing errors
     (bugs) from your code

1. Syntax Errors

- Spelling, punctuation, grammar errors detected by:
  - Code editor as you type
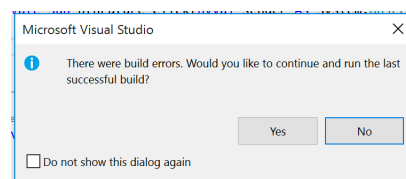 - Compiler when you compile (build)
- What are syntax (compiler) errors below?

```
txtHello.Text = Hello World
tstHello.Text = "Hello World"
txtHello,Text = "Hello World"
```

Syntax Errors Detected by Compiler

Microsoft Visual Studio
There were build errors. Would you like to continue and run the last successful build?

Yes   No

☐ Do not show this dialog again

5

Syntax Errors Detected by Compiler

Error List							
Entire Solution ▾	⊗ 2 Errors	⚠ 0 Warnings	ⓘ 0 of 3 Messages		Build + IntelliSense ▾		

Search Error List

	Code	Description	Project	File	Line	Suppression S...	▼
⊗	BC30451	'tstQuote' is not declared. It may be inaccessible due to its protection level.	ch02_Quotes	Form1.vb	6	Active	
⊗	BC30456	'for' is not a member of 'TextBox'.	ch02_Quotes	Form1.vb	13	Active	

Visual Basic 2017   31

---

2. Run-time Errors

- Caused by running program containing statement that computer cannot execute
- E.g., Attempting impossible arithmetic operations like divide by zero, calculations with nonnumeric data
- Stops execution (running) of program

Visual Basic 2017   32

---

3. Logic Errors

- Program compiles and runs without errors
- But produces incorrect results

Visual Basic 2017   33

Event Procedures

```
Private Sub objectName_event(...)
    Handles objectName.event
  statements  to handle event
End Sub

Private Sub btnTruth_Click(...)
    Handles  btnTruth.Click
  txtQuote.Text = "Tell the truth and run."
  txtQuote.ForeColor = Color.Black
End Sub
```

Visual Basic 2017   34

Changing Properties in Code

- Program statement to set a property:
  controlName.property = setting
- Examples:
  ```
  txtQuote.ForeColor = Color.Black
  txtQuote.Text = "Hey now!"
  btnFuture.BackColor = Color.Blue
  ```

- These are assignment statements

Visual Basic 2017   35

Sample Code

```
Private Sub btnLife_Click(…)Handles btnLife.Click
  txtQuote.Text = "I like life, it's something to do."
  txtQuote.ForeColor = Color.Black
End Sub
      …
Private Sub btnTruth_Click(…)Handles btnTruth.Click
  txtQuote.Text = "Tell the truth and run."
  txtQuote.ForeColor = Color.Black
End Sub

Private Sub txtQuote_TextChanged(…)Handles
  txtQuote.TextChanged
  txtQuote.ForeColor = Color.Red
End Sub
```

Visual Basic 2017   36

Review: Visual Basic Project

- Creating your project
  - Name
  - Location
- Locating your project
- Turning in your project