



# **VISUAL BASIC II**

**CC111  
INTRODUCTION TO COMPUTERS**



# Intended Learning Objectives



- Able to build a simple Visual Basic Application.

# The Sub Statement

```
Private Sub ControlName_eventName(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ControlName.eventName`
```

```
End Sub
```

Where

**Private** is the default procedure type  
**Sub** indicates beginning of procedure  
*controlname* is name of associated control  
\_ (underscore) required separator  
*eventname* is name of corresponding event  
( ) set of parentheses is required  
**End Sub** indicates end of a procedure

## Example on the Sub Statement

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click  
    Const pi = 3.14  
    Dim r As Integer  
    Dim area As Double  
    r = Val(TextBox1.Text)  
    area = pi * r * r  
    Label3.Text = Str(area)  
End Sub
```

# Declarations, Variables, and Constants

**Variable** - a uniquely named storage location that contains data that **changes** during program execution

**Constant** - a uniquely named storage locations that contains data that **does not change** during program execution

# Declarations, Variables, and Constants

## Rules for Naming Variables

- Must begin with an alphabetic character
- Can not contain a period (.) or type-declaration characters such as %, &, !, #, @ or \$
- Must be unique with same scope
- Must be no longer than 255 characters
- Should not be a reserved word

salary



sal ary



x\$



salary4



# Declaring Variables



**Declaration statement** – non executable code that sets aside storage locations for future use

**Local variables** - declared within a procedure or function

**Global variables** - declared in the general section of the application

# Declaring Variables....

- Declare variables using the Dim statements

**Dim statement** - value of variable preserved only until procedure ends

**Dim** *variablename* **As** *datatype*

# The Dim Statement

**Dim** *variablename* **As** *datatype*

Where

**Dim** is required

*variablename* should be a descriptive name

**As** is required

*datatype* is one of the following types:

Boolean, Byte, Date, Integer, Long,  
Single, Double, Currency, String, Object or Variant

Dim x As Integer



# Declaring Variables

## Data Types

- **Boolean** - True or false
- **Date** - From Jan 1, 100 to Dec 31, 9999
- **Integer** - Numbers without a decimal point
- **Long** - Long integer
- **Single** - Numbers with a decimal point
- **Double** - Long Single
- **Currency** - Dollar amounts
- **String** - Character and alphanumeric data
- **Object** - Any object reference such as Word document
- **Variant** - default, can hold any data type

# Assigning Values to Variables

*Variable*name = value

Where

*variablename* is the descriptive name of the variable

= is the assignment operator

*value* is the value the variable will contain

Examples:

```
Number1 = 5
```

```
FirstName = "Steve"
```

```
Length = 17.8
```

Note: Order is important. Variable name always on the left, and value on the right.

Variables can be declared and initialized in the same line:

```
Ex::Dim x As Integer=12
```

# Declaring Constants

```
Const constantname As datatype = value
```

Where

**Const** is required

*constantname* is the descriptive name of the constant

**As** is required

*datatype* is the type of data the constant will contain

= is the assignment operator

*value* is the value of the constant

Examples:

```
Const Pi As Single = 3.14159265358979
```

```
Const MaxNumber As Integer = 100
```

# Functions

- Function - unit of code that returns a value
- Built-in Functions
  - `Math.Sqrt` - square root
  - `Rnd` - random number generator
  - `Int` - returns integer portion of a number
  - `Val` - converts a string to a value
  - `Str` - converts a value to a string

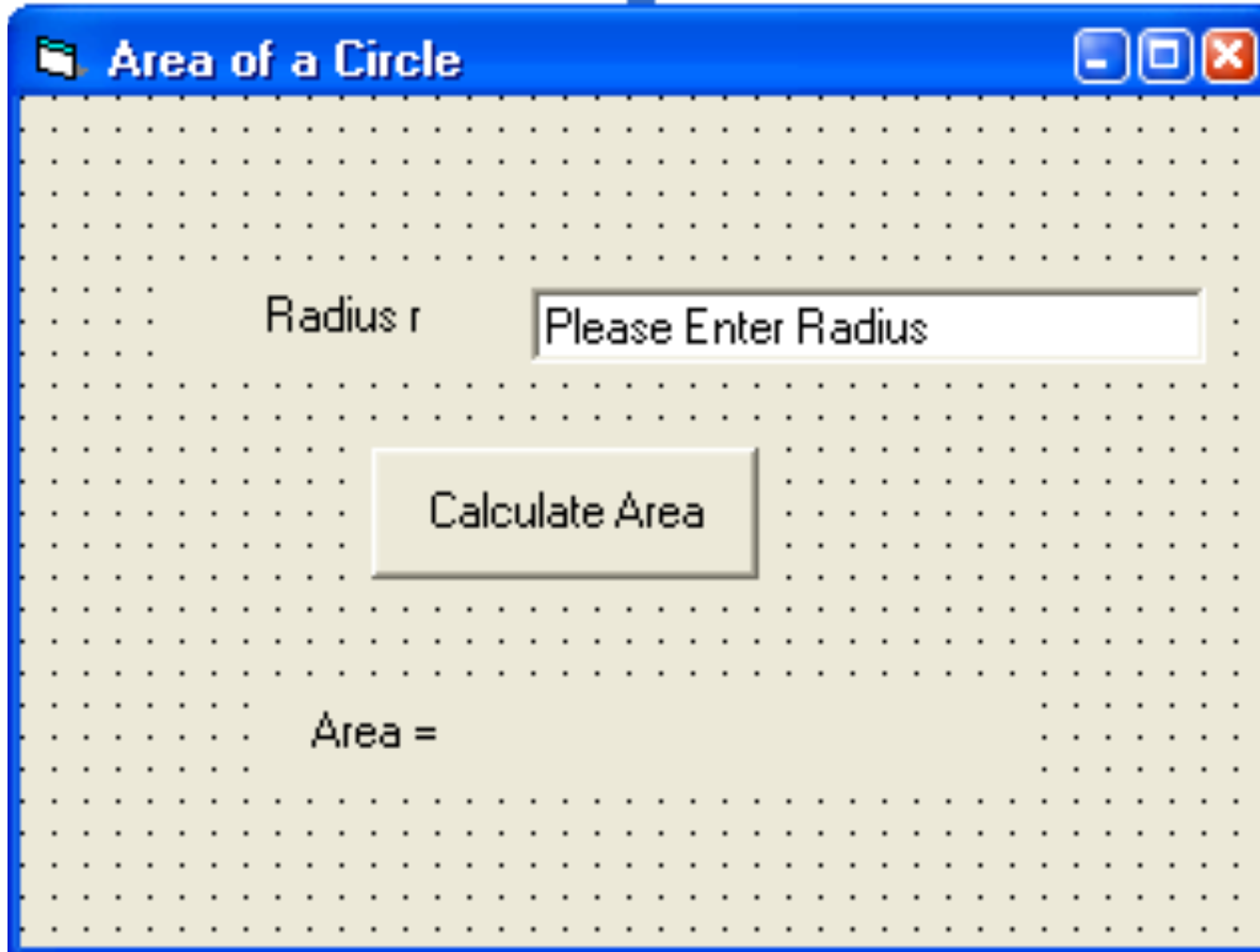
# Application 1: Calculate the area of a circle

- Inputs: radius of the circle  $r$
- Output: area of the circle

- Process: Area =

$$\pi r^2$$

# Application 1: Calculate the area of a circle (Form View)



The image shows a screenshot of a Windows application window titled "Area of a Circle". The window has a blue title bar with standard minimize, maximize, and close buttons. The main content area has a light beige background with a grid of small dots. The form contains the following elements:

- A label "Radius r" positioned to the left of a text input field.
- A text input field containing the placeholder text "Please Enter Radius".
- A button labeled "Calculate Area" centered below the input field.
- A label "Area =" positioned below the button, indicating where the result will be displayed.

# Application 1: Calculate the area of a circle (Code View)

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)  
Handles Button1.Click
```

```
    Const pi As Double = 3.14
```

```
    Dim r As Integer
```

```
    Dim area As Double
```

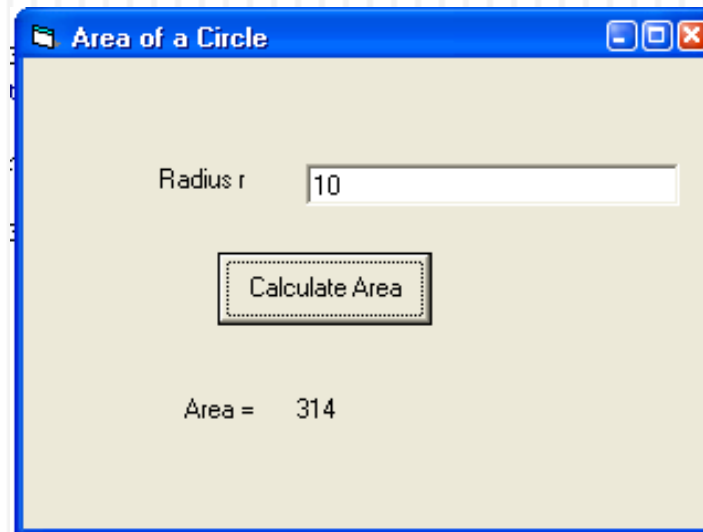
```
    r = Val(TextBox1.Text)
```

```
    area = pi * r * r
```

```
    Label3.Text = Str(area)
```

```
End Sub
```

# Application 1: Calculate the area of a circle (Run)





# Precedence Table

Operator
( )
^
*, /
+, -

If precedence of two following operators is equal, then the evaluation starts from left to right.

# Simple Examples of Equations

□ Equation in **normal** form:

1.  $Y=3X$

2.  $Y=X-10+3(X-Z)$

3.  $Y= \sqrt{X}$

● Equation in **VB** form:

1.  $Y=3*X$

2.  $Y=X-10+3*(X-Z)$

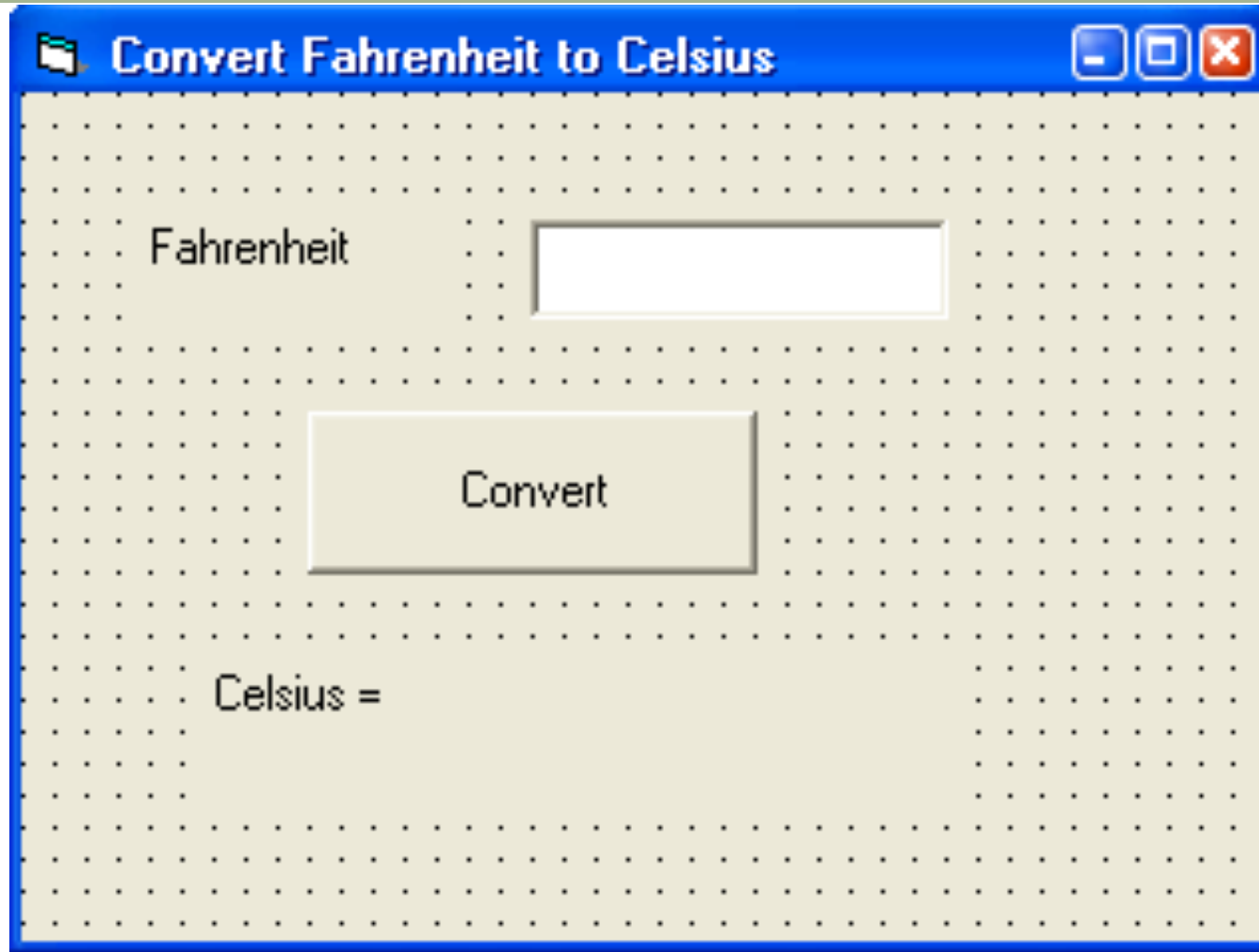
3.  $Y=Math.Sqrt(X)$

## Application 2: Convert from Fahrenheit Degree to Celsius

- Original formula:  $Celsius = \frac{5}{9} (Fahrenheit - 32)$
- Visual Basic formula:  $Celsius = \frac{5}{9} * (Fahrenheit - 32)$
- Input: Fahrenheit
- Output: Celsius
- Process:  $Celsius = \frac{5}{9} * (Fahrenheit - 32)$

# Application 2

## Convert from Fahrenheit Degree to Celsius (Form View)



The image shows a screenshot of a Windows application window titled "Convert Fahrenheit to Celsius". The window has a blue title bar with standard minimize, maximize, and close buttons. The main content area has a light beige background with a dotted grid pattern. It contains the following elements:

- A label "Fahrenheit" followed by an empty text input field.
- A button labeled "Convert" positioned below the input field.
- A label "Celsius =" followed by an empty space for the result.

# Application 2

Convert from Fahrenheit Degree to Celsius (Code View)

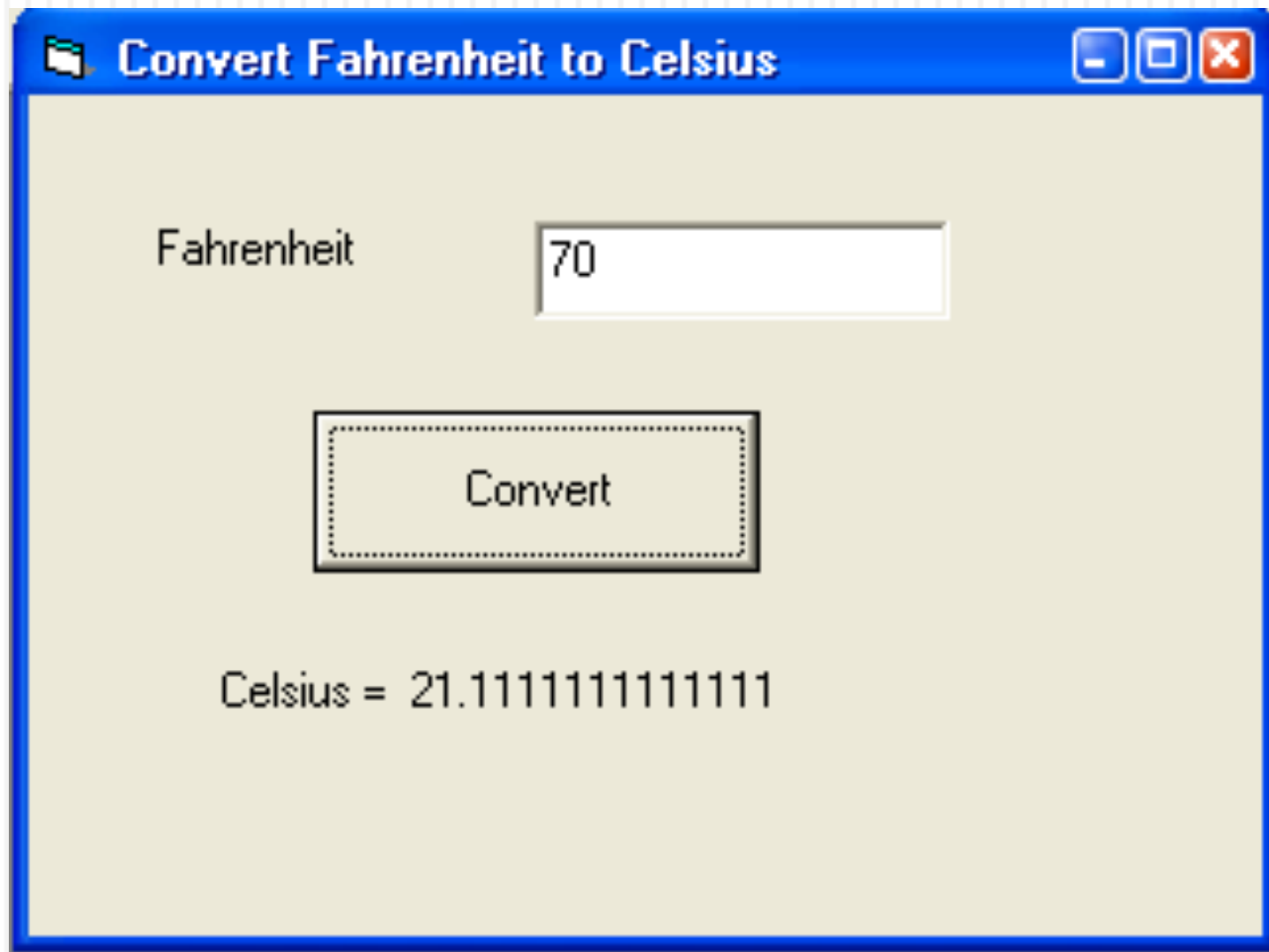
```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button1.Click
    Dim f As Double
    Dim c As Double
    f = Val(TextBox1.Text)
    c = 5 / 9 * (f - 32)
    Label2.Text = "Celsius=" + Str(c)
End Sub
```

Note:

The + operator is used to concatenate strings

## Application 2

### Convert from Fahrenheit Degree to Celsius (Run)



## Application 3

### The Wind Chill Application

- Write a program that calculates the wind chill temperature
- Inputs: Wind Speed and Temperature
- Outputs: Wind Chill Temperature

# Application 3

## The Wind Chill Application

- Original formula

$$WC = 0.0817(3.71(V^{**}0.5) + 5.81 - 0.25V)(T - 91.4) + 91.4$$

- Visual Basic statement

$$WC = 0.0817 * (3.71 * Sqr(V) + 5.81 - (0.25 * V)) * (T - 91.4) + 91.4$$

- Output: Wind Chill (WC)
- Input 1 T: Temperature
- Input 2 V: Wind Speed



# Adding Controls and Changing their Properties

The screenshot displays the Microsoft Visual Basic IDE in design mode. The main window is titled "Project1 - Microsoft Visual Basic [design]". The central area shows a form named "Project1 - Form1 (Form)" with a title bar "Wind Chill Program". The form contains several controls: a label "Wind Speed V" with a text box "Please Enter Wid Speed", a label "Temperature T" with a text box "Please Enetre Temperature", a button "Calculate Wind Chill", and a label "Wind Chill WC" with a text box. A red box highlights the "Calculate Wind Chill" button. A white box labeled "Properties" has an arrow pointing to the "Properties - Form1" window on the right. The "Properties - Form1" window shows the "Form1 Form" selected, with the "Caption" property set to "id Chill Program". The "Form Layout" window shows a small icon of the form.

Project1 - Microsoft Visual Basic [design]

File Edit View Project Format Debug Run Tools Add-Ins Window Help

Project1 - Form1 (Form)

Wind Chill Program

Wind Speed V Please Enter Wid Speed

Temperature T Please Enetre Temperature

Calculate Wind Chill

Wind Chill WC

Properties

Project - Project1

Project1 (Project1.vbp)

Forms

Form1 (Form1.frm)

Properties - Form1

Form1 Form

Alphabetic Categorized

Appearance 1 - 3D

AutoRedraw False

BackColor &H8000000F

BorderStyle 2 - Sizable

Caption id Chill Program

ClientControl True

Caption

Returns/sets the text displayed in an object's title bar or below an object's

Form Layout

start Project1 - Microsoft V... untitled - Paint EN 05:02

# Writing the Code

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
```

```
    Dim V As Integer
```

```
    Dim T As Integer
```

```
    Dim WC As Double
```

```
    v = Val(TextBox1.Text)
```

```
    T = Val(TextBox2.Text)
```

```
    WC = 0.0817 * (3.71 * Math.Sqrt(V) + 5.81 - (0.25 * V)) * (T - 91.4) + 91.4
```

```
    TextBox3.Text = Str(WC)
```

```
End Sub
```

# Running the Application

