

Today's Agenda

- Schedule, announcements
- Exam Study guide
- VB Assignment
- Review of VB concepts
- VB Cont'd – Ch 5 and 6

Exam 1

- **WHEN:** Sept 25, 6:30-7:25, total duration 45 min
- **WHERE:** BUSI 116
- **REMINDER OF IMPORTANT EXAM POLICIES:**
 - Closed books, closed notes
 - No electronic devices
 - You are required to present a UNT ID
- **WHAT TO STUDY**
 - Amer book, Ch 1-3
 - Power Point, Ch. 1, 2
- **WHAT TO EXPECT**
 - Total 50-60 questions, of which 30 on PP and 20-30 on Amer

4

Announcements

- Student recommended links:
 - From Chris:
 - If you've got students that would like to purchase the MS Office Suite, here's a link. they just have to make sure to use their unt.edu email address.
 - <http://www.microsoft.com/student/discounts/theultimatesteal-us/default.aspx>
 - From Cody
 - Microsoft recently partnered with many Universities (UNT included) and is giving their new development software (Visual Studio 2008 and Expression Studio 2008 included) to registered students through a program called Dreamspark. The only requirement is that they have a Windows Live account.
 - <https://downloads.channel8.msdn.com>

Exam study info on the Web

- Practice questions, etc. - http://www.coba.unt.edu/itds/faculty/sidorova/BCIS2610/Exam1Guide_B.pdf
- Topical overview - http://www.coba.unt.edu/itds/faculty/sidorova/BCIS2610/Exam1Guide_A.pdf

5

Schedule Weeks 1-5

	Topics	Reading	Assignments Due
Week 1	Aug 28 Course Overview PowerPoint 2007 Tutorials 1 & 2	Amer 1 PowerPoint 1, 2	
Week 2	Sep 4 Computers in Business Lecture from Ch. 1, 2 of VB 2008 PowerPoint 2007 Tutorials Q&A	Amer 1 VB 1, 2	
Week 3	Sept 11 Lecture from Ch. 3 of VB 2008 Computers in Government Computers in Science	VB 3 Amer 2 Amer 3	PowerPoint Assign. due by 5:00 pm FRIDAY
Week 4	Sept 18 Lecture from Ch. 5 of VB 2008 Lecture from Ch. 6 of VB 2008	VB Ch. 5 VB Ch. 6	
Week 5	Sept 25 EXAM 1 (45 min) covers PP and Amer 1, 2, 3 Computers in Entertainment Computers in Transportation Computers in Education	Amer 4 Amer 5 Amer 6	Visual Basic Assign. due by 5:00pm FRIDAY.

VB assignment

- Download Visual Basic tutorial files to a storage medium of your choice, as described in General Assignment Guidelines.
- Complete Chapter 2 and 3 tutorials (highlighted on pages 83-93 and 160-177).
- Complete Chapter 4 tutorial (highlighted on pages 254-272).
- Complete Chapter 5 tutorial (highlighted on pages 351-362).
- Complete Chapter 6 tutorial (highlighted on pages 442-461).

Last lecture

- What is VB
- Objects and properties
 - Form
 - Label
 - Button
 - PictureBox
 - TextBox
- Variables, their data types and scope

Assignment Statements

```

1 Public Class Form1
2
3
4     Private Sub btnCalculateCost_Click(ByVal sender As System.Object, ByVal e As EventArgs) Handles btnCalculateCost.Click
5         Dim strNumberOfSongs As String
6         strNumberOfSongs = Me.txtNumberOfSongs.Text
7     End Sub
8 End Class
9

```

What we learned about variables

- A program can have
 - Variables (can assume different values)
 - Constants (the same value)
- Variables and constants have to be declared
 - **Dim intNumberOfSongs as Integer**
 - **Const cdecPricePerSong as Decimal**
- Variables and constants can be:
 - Global **_cdecPricePerSong**
 - Local **intNumberOfSongs**

Data types and their prefixes

Data Type	Prefix
String	str
Integer	int
Decimal	dec
Double	dbl
Char	chr
Boolean	bln
Byte	byt
Date	dtn
Long	lng
Short	shr
Single	sng

Assignment Statements

- One method to place data in the variable is to use an **assignment statement**

```

strNumberOfSongs = Me.txtNumberOfSongs.Text
strNumberOfSongs = "15"

```

- Uses equal sign operator
- The value which is assigned is on the right
- The variable or property, to which the value is assigned is on the left

Converting Variable Data

- Visual Basic includes several procedures that allow you to convert one data type to another data type
- A procedure to convert a String data type to an Integer data type is named ToInt32
- The procedure is found in the Convert class, which is available in a Visual Studio 2005 class library

```

intNSongs=Convert.ToInt32(strNSongs)

```

Displaying Numeric Output Data

Format: ToString Function

```
decimalvariable.ToString()
```

Clear Procedure

- The Clear procedure clears any data currently placed in the Text property of a TextBox object

General Format: Clear Procedure

```
txtTextBoxName.Clear()
```

EXAMPLE: `Me.txtNumberOfSongs.Clear()`

Format Specifications for the ToString Function

- Use the **format specifier** to identify the format for the numeric data to be returned by the ToString function

Format Specifier	Format	Description	Output from the Function
General (G)	To-String("G")	Displays the number as is	8976.43561
Currency (C)	To-String("C")	Displays the number with a dollar sign, a thousands separator (comma), two digits to the right of the decimal and negative numbers in parentheses	\$8,976.44
Fixed (F)	To-String("F")	Displays the number with 2 digits to the right of the decimal and a minus sign for negative numbers	8976.44
Number (N)	To-String("N")	Displays a number with a thousands separator, 2 digits to the right of the decimal and a minus sign for negative numbers	8,976.44
Percent (P)	To-String("P")	Displays the number multiplied by 100 with a % sign, a thousands separator, 2 digits to the right of the decimal and a minus sign for negative numbers	897,643.56%
Scientific (E)	To-String("E")	Displays the number in E-notation and a minus sign for negative numbers	8.9764356E+03

Arithmetic Operations

```
Dim strNumberOfSongs As String
Dim intNumberOfSongs As Integer
Dim decTotalCostOfDownloads As Decimal
Const cdecPricePerDownload As Decimal = 0.99D

strNumberOfSongs = Me.txtNumberOfDownloads.Text
intNumberOfSongs = Convert.ToInt32(strNumberOfSongs)
decTotalCostOfDownloads = intNumberOfSongs * cdecPricePerDownload
```



```
decTotalCostOfDownloads = intNumberOfSongs * cdecPricePerDownload
```

Precision Specifier

- The **precision specifier** is a number that is included within the quotation marks in the function call to identify the number of positions to the right of the decimal point that should be returned

Statement	Copied to Text Property of lblOutput Label Object
lblOutput = decNumericValue.ToString("C2")	\$8,976.44
lblOutput = decNumericValue.ToString("C3")	\$8,976.436
lblOutput = decNumericValue.ToString("F1")	8976.4
lblOutput = decNumericValue.ToString("N4")	8,976.4356
lblOutput = decNumericValue.ToString("P0")	897,644%

Arithmetic Operators

Arithmetic Operator	Use	Assignment Statement Showing Their Use
+	Addition	decTotal = decPrice + decTax
-	Subtraction	decCost = decRegularPrice - decDiscount
*	Multiplication	decTax = decItemPrice * decTaxRate
/	Division	decClassAverage = decTotalScores / intNumberOfStudents
^	Exponentiation	intSquareArea = intSquareSide ^ 2
\	Integer Division	intResult = 13 \ 5
Mod	Modulus Arithmetic (remainder)	intRemainder = 13 Mod 5

Multiple Operations

- A single assignment statement can contain multiple arithmetic operations
- Hierarchy of Operations
 - Exponentiation (^) is performed first
 - Multiplication (*) and division (/) are performed next
 - Integer division (\) is next
 - MOD then occurs
 - Addition (+) and subtraction (-) are performed last
 - Within these five steps, calculations are performed left to right

Concatenation

- The process of joining two different values into a single string is called **concatenation**
- The values being concatenated must be String data types
 - Use the concatenation sign &

```
Me.lblCostHeading.Text = cdecPricePerDownload.ToString("C") _
& " Per Download"
```

Clearing the Text Property of a Label

- The Clear procedure cannot be used with a Label object
- You must write an assignment statement that assigns a null length string to the Text property of a Label object

```
Me.lblTotalCostOfDownloads.Text = ""
```

Visual Basic Ch.5

Form Load Event

- A form load event occurs when the program starts and the Digital Downloads form is loaded
- This event handler completes the following tasks:
 - Display the cost per download heading
 - Clear the placeholder from the lblTotalCostOfDownloads Text property
 - Set the focus on the txtNumberOfSongs text box

Objectives

- Write programs for devices other than a personal computer
- Understand the use of handheld technology
- Write handheld applications for a Personal Digital Assistant
- Use the Panel object
- Place RadioButton objects in applications

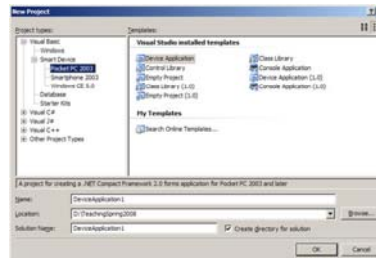
Objectives

- Display a message box
- Make decisions using If...Then statements
- Make decisions using If...Then...Else statements
- Make decisions using nested If statements

25 Chapter 5: Mobile Applications Using Decision Structures

Create a Smart Device Application

- Select Smart Devices/Pocket PC 2003



28 Chapter 5: Mobile Applications Using Decision Structures

Objectives

- Make decisions using logical operators
- Make decisions using Case statements
- Test input to ensure a value is numeric

26 Chapter 5: Mobile Applications Using Decision Structures

Create a Smart Device Application



29 Chapter 5: Mobile Applications Using Decision Structures

Chapter Project



27 Chapter 5: Mobile Applications Using Decision Structures

Placing Objects on the PocketPC Form Object

- Many of the same objects as in a Windows application
- Form object
 - Cannot resize the Form object
 - Can be named using the (Name) property
 - Text property used in the same way as in the Windows Form object

30 Chapter 5: Mobile Applications Using Decision Structures

Placing Objects on the PocketPC Form Object



31 Chapter 5: Mobile Applications Using Decision Structures

Windows Application Container Objects

Option	GroupBox Object	Panel Object
Have a caption	Yes	No
Have scroll bars	Yes	No
Display a labeled border	Yes	No

34 Chapter 5: Mobile Applications Using Decision Structures

Using the Panel Object

- Panel object is used as a container for other objects, such as radio buttons or check boxes
 - Can be added from the toolbox
 - Name should start with *pnI*, e.g. *pnIWoodType*
 - Can be resized using sizing handles and size property

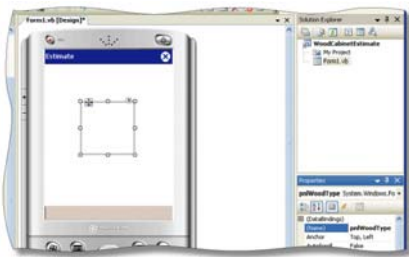
32 Chapter 5: Mobile Applications Using Decision Structures

Adding the RadioButton Objects

- RadioButton object
 - Name should start with *rad* prefix, e.g. *radCherry*
 - Text property determines what is written next to the radio button, e.g. Cherry
 - Checked property
 - True if the radio button is checked
 - False if the radio button is not checked

35 Chapter 5: Mobile Applications Using Decision Structures

Using the Panel Object



33 Chapter 5: Mobile Applications Using Decision Structures

Adding the RadioButton Objects



36 Chapter 5: Mobile Applications Using Decision Structures

Messages

- Used for warning or informing the user

Input Error ok

Enter the Linear Feet of the Cabinets.

Enter the Linear Feet of the Cabinets

OK

37 Chapter 5: Mobile Applications Using Decision Structures

Displaying a Message Box

- Adding a buttons option – the third argument
 - MessageBoxButtons.OKCancel

EXAMPLE

Windows Application:
 MessageBox.Show("User name is missing", "User Name Text Box Error", MessageBoxButtons.OKCancel)

Windows Application:
 MessageBox.Show("You have been disconnected from the Internet", "ISP", MessageBoxButtons.RetryCancel)

RESULT

User Name Text Box Error

User name is missing

OK Cancel

ISP

You have been disconnected from the Internet

Retry Cancel

40 Chapter 5: Mobile Applications Using Decision Structures

Displaying a Message Box

- General syntax
 - MessageBox.Show("Message text")

EXAMPLE

Mobile Application:
 MessageBox.Show("Enter the Linear Feet of the Cabinets.")

Input Error ok

Enter the Linear Feet of the Cabinets.

38 Chapter 5: Mobile Applications Using Decision Structures

Displaying a Message Box

MessageBoxButton Arguments	Use
MessageBoxButtons.AbortRetryIgnore	After a failing situation, the user can choose to Abort, Retry, or Ignore
MessageBoxButtons.OK	Displays an OK button — default setting
MessageBoxButtons.OKCancel	Displays an OK and cancel button
MessageBoxButtons.RetryCancel	After an error occurs, the user can choose to Retry or Cancel
MessageBoxButtons.YesNo	Displays Yes and No buttons
MessageBoxButtons.YesNoCancel	Displays Yes, No, and Cancel buttons

41 Chapter 5: Mobile Applications Using Decision Structures

Displaying a Message Box

- To display a message with caption
 - MessageBox.Show("Message", "Caption")

EXAMPLE

Mobile Application:
 MessageBox.Show("Enter the Linear Feet of the Cabinets.", "Error: Missing a Number")

Windows Application:
 MessageBox.Show("User name is missing", "User Name Text Box Error")

RESULT

Error: Missing a Number

Enter the Linear Feet of the Cabinets.

User Name Text Box Error

User name is missing

OK

39 Chapter 5: Mobile Applications Using Decision Structures

Displaying a Message Box

General Format: Message Box Statement with Caption and Button

MessageBox.Show("Message", "Caption", Button Entry, Icon Picture)

EXAMPLE

Windows Application: MessageBox.Show("User name is missing", "User Name Text Box Error", MessageBoxButtons.OKCancel, MessageBoxIcon.Error)

Windows Application:
 MessageBox.Show("You have been disconnected from the Internet", "ISP", MessageBoxButtons.RetryCancel, MessageBoxIcon.Warning)

RESULT

User Name Text Box Error

User name is missing

OK Cancel





ISP

You have been disconnected from the Internet

Retry Cancel

42 Chapter 5: Mobile Applications Using Decision Structures

Displaying a Message Box

MessageBoxIcon Argument Constants	Icon	Use
MessageBoxIcon.Asterisk MessageBoxIcon.Information		Displays an information icon
MessageBoxIcon.Error MessageBoxIcon.Hand MessageBoxIcon.Stop		Alerts the user to an error
MessageBoxIcon.Exclamation MessageBoxIcon.Warning		Alerts the user to a possible problem
MessageBoxIcon.Question		Displays a question mark

43 Chapter 5: Mobile Applications Using Decision Structures

Making Decisions with Conditional Statements: Using an If...Then Statement

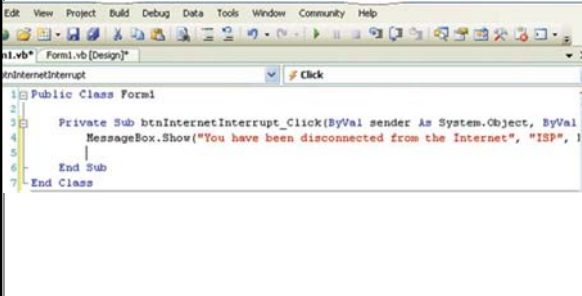
- A **decision structure** is one of the three fundamental control structures used in computer programming
- When a condition is tested in a Visual Basic program, the condition either is true or false

```

If condition Then
    Statement(s) executed when condition is true
End If
    
```

46 Chapter 5: Mobile Applications Using Decision Structures

Displaying a Message Box



```

1 Public Class Form1
2
3     Private Sub btnInternetInterrupt_Click(ByVal sender As System.Object, ByVal
4         e As EventArgs) Handles btnInternetInterrupt.Click
5         MessageBox.Show("You have been disconnected from the Internet", "ISP", 1)
6     End Sub
7 End Class
    
```

44 Chapter 5: Mobile Applications Using Decision Structures

General Syntax

```

If <cond> Then
    Statements
ElseIf <cond> Then
    Statements
Else
    Statements
EndIf
    
```


Optional

47 Chapter 5: Mobile Applications Using Decision Structures

Displaying a Message Box

```

MessageBox.Show("You entered " & declinearFeet.ToString() &
    ". Enter a Number Greater Than Zero.", "Input Error")
    
```



45 Chapter 5: Mobile Applications Using Decision Structures

Relational Operators

Relational Operator	Meaning	Example	Resulting Condition
1 =	Equal to	8 = 8	True
2 <>	Not equal to	6 <> 6	False
3 >	Greater than	7 > 9	False
4 <	Less than	4 < 6	True
5 >=	Greater than or equal to	3 >= 3	True
6 <=	Less than or equal to	7 <= 5	False

```

If intAge >= 21 Then
    Me.lblVotingEligibility.Text = "You are old enough to vote"
End If
    
```

48 Chapter 5: Mobile Applications Using Decision Structures

Comparing Different Data Types

- Every type of data available in Visual Basic can be compared
 - Different numeric types can be compared to each other
 - A single string character can be compared to a Char data type

49

Chapter 5: Mobile Applications Using Decision Structures

Using the If...Then...Elseif Statement

```
If decOrderAmount > 500D Then
    Statement(s) executed if condition is true
ElseIf decOrderAmount > 400D Then
    Statement(s) executed if condition is true
ElseIf decOrderAmount > 200D Then
    Statement(s) executed if condition is true
ElseIf decOrderAmount > 0D Then
    Statement(s) executed if condition is true
End If
```

52

Chapter 5: Mobile Applications Using Decision Structures

Comparing Strings

- A string value comparison compares each character in two strings, starting with the first character in each string

```
Dim String1 As String = "abc"
Dim String2 As String = "abc"

If String1 = String2 Then
    Me.lblStringTest.Text = "Equal"
End If
```

50

Chapter 5: Mobile Applications Using Decision Structures

Using the If...Then...Elseif Statement

```
If intAge >= 65 Then
    Me.lblSocialSecurity.Text = "Full Benefits"
ElseIf intAge > 0 Then
    Me.lblSocialSecurity.Text = "Not Eligible for Benefits"
Else
    Me.lblSocialSecurity.Text = "Invalid Age"
End If
```

53

Chapter 5: Mobile Applications Using Decision Structures

Using the If...Then...Else Statement

```
27 If strStudentStatus = "Graduate" Then
28     decStudentFees = decGraduateFee * intNumberOfUnits
29 Else
30     decStudentFees = decUndergraduateFee * intNumberOfUnits
31 End If
```

```
27 If strStudentStatus = "Graduate" Then
28     decStudentFees = decGraduateFee * intNumberOfUnits
29 Else
30     decStudentFees = decUndergraduateFee * intNumberOfUnits
31 End If
```

```
41 If decWithdrawals > decCurrentBalance - decDeposits - decAccountCharges Then
42     Me.lblAccountStatus.Text = "Overdrawn"
43 Else
44     Me.lblAccountStatus.Text = "Balance is Positive"
45 End If
```

51

Chapter 5: Mobile Applications Using Decision Structures

Nested If Statements

```
If decGPA > 3.5D Then
    If intSatScore > 1100 Then
        lblAdmissionStatus.Text = "You have earned admission"
    Else
        lblAdmissionStatus.Text = "Retake the SAT exam"
    End If
Else
    lblAdmissionStatus.Text = "You have been denied admission"
End If
```

54

Chapter 5: Mobile Applications Using Decision Structures

Nested If Statements

```

123 If first condition Then
124     If second condition Then
125         Statement(s) executed if condition 1 is true and condition 2 is true
126     Else
127         Statement(s) executed if condition 1 is true and condition 2 is false
128     End If
129 Else
130     If third condition Then
131         Statement(s) executed if condition 1 is false and condition 3 is true
132     Else
133         Statement(s) executed if condition 1 is false and condition 3 is false
134     End If
135 End If

```

```

140 If decGPA > 3.5D Then
141     If intSatScore > 1000 Then
142         lblAdmissionStatus.Text = "You have earned admission"
143     Else
144         lblAdmissionStatus.Text = "Retake the SAT exam"
145     End If
146 Else
147     If intSatScore > 1200 Then
148         lblAdmissionStatus.Text = "You have earned probationary admission"
149     Else
150         lblAdmissionStatus.Text = "You have been denied admission"
151     End If
152 End If

```

55

Chapter 5: Mobile Applications Using Decision Structures

Block-Level Scope

- Scope is defined by where the variable is declared within a program
- Within an event handler, an If...Then...Else statement is considered a block of code
- Variables can be declared within a block of code
 - The variable can be referenced only within the block of code where it is declared

58

Chapter 5: Mobile Applications Using Decision Structures

Matching If, Else, and End If Entries

- If statements must be fully contained within the outer If statement
- Place the correct statements with the correct If and Else statements within the nested If statement
 - This illustration shows incorrect logic

56

Chapter 5: Mobile Applications Using Decision Structures

Using Logical Operators

- When more than one condition is included in an If...Then...Else statement, the conditions are called a **compound condition**

Logical Operator	Meaning
And	All conditions tested in the If statement must be true
Or	One condition tested in the If statement must be true
Not	Negates a condition

59

Chapter 5: Mobile Applications Using Decision Structures

Testing the Status of a RadioButton Object in Code

```

If Me.radPine.Checked Then
    decCostPerFoot = decPineCost
ElseIf Me.radOak.Checked Then
    decCostPerFoot = decOakCost
ElseIf Me.radCherry.Checked Then
    decCostPerFoot = decCherryCost
End If

```

57

Chapter 5: Mobile Applications Using Decision Structures

Using the And/OR Logical Operator

```

If decFlightCost < 300D And decHotelCost < 120D Then
    Me.lblTripMessage.Text = "Your business trip is approved"
Else
    Me.lblTripMessage.Text = "Your business trip is denied"
End If

If decGPA >= 3.5D Or intSATScore >= 1080 Then
    Me.lblAcceptance.Text = "You have been accepted"
Else
    Me.lblAcceptance.Text = "You are not accepted"
End If

```

60

Chapter 5: Mobile Applications Using Decision Structures

Other Logical Operators

Logical Operator	Meaning
Xor	When one condition in the compound condition is true, but not both, the compound condition is true
AndAlso	As soon as a condition is found to be false, no further conditions are tested and the compound condition is false
OrElse	As soon as a condition is found to be true, no further conditions are tested and the compound condition is true

61 Chapter 5: Mobile Applications Using Decision Structures

Select Case Statement

```
Select Case Test Expression
Case First Expression
  Statement(s) for First Case
Case Second Expression
  Statement(s) for Second Case
Case Third Expression
  Statement(s) for Third Case
Case Else
  Statement(s) for when the Case Conditions do not match the
  test expressions above
End Select
```

64 Chapter 5: Mobile Applications Using Decision Structures

Order of Operations for Logical Operators

Logical Operator	Order
Not	Highest Precedence
And, AndAlso	Next Precedence
Or, OrElse, Xor	Last Precedence

62 Chapter 5: Mobile Applications Using Decision Structures

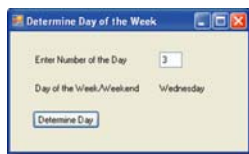
Select Case Statement

```
Select Case intDayNumber
Case 1
  Me.lblDayOfWeek.Text = "Monday"
Case 2
  Me.lblDayOfWeek.Text = "Tuesday"
Case 3
  Me.lblDayOfWeek.Text = "Wednesday"
Case 4
  Me.lblDayOfWeek.Text = "Thursday"
Case 5
  Me.lblDayOfWeek.Text = "Friday"
Case 6
  Me.lblDayOfWeek.Text = "Weekend"
Case 7
  Me.lblDayOfWeek.Text = "Weekend"
Case Else
  Me.lblDayOfWeek.Text = "Enter 1 through 7"
End Select
```

65 Chapter 5: Mobile Applications Using Decision Structures

Select Case Statement

- In some programming applications, different operations can occur based upon the value in a single field



63 Chapter 5: Mobile Applications Using Decision Structures

Select Case Test Expressions

```
Select Case Me.txtStudentMajor.Text
Case "Accounting"
  Me.lblDepartment.Text = "Business"
Case "Marketing"
  Me.lblDepartment.Text = "Business"
Case "Electrical Engineering"
  Me.lblDepartment.Text = "Engineering"
Case "Biochemistry"
  Me.lblDepartment.Text = "Chemistry"
Case "Shakespearean Literature"
  Me.lblDepartment.Text = "English"
Case "Web Design and E-Commerce"
  Me.lblDepartment.Text = "CIS"
Case Else
  Me.lblDepartment.Text = "Other"
End Select
```

66 Chapter 5: Mobile Applications Using Decision Structures

Using Relational Operators in a Select Case Statement

```

191 Select Case decOrderAmount
192     Case Is > 5000
193         decShippingCost = 300
194     Case Is > 4000
195         decShippingCost = 250
196     Case Is > 3000
197         decShippingCost = 200
198     Case Is > 200
199         decShippingCost = 150
200     Case Else
201         decShippingCost = 00
202 End Select
203

```

67

Chapter 5: Mobile Applications Using Decision Structures

Code Snippets



70

Chapter 5: Mobile Applications Using Decision Structures

Using Ranges in Select Case Statements

```

224 Select Case intGradeLevel
225     Case 1 To 3
226         Me.lblGradeLevelExam.Text = "Early elementary"
227     Case 4 To 6
228         Me.lblGradeLevelExam.Text = "Late elementary"
229     Case 7 To 8
230         Me.lblGradeLevelExam.Text = "Middle school"
231     Case 9 To 10
232         Me.lblGradeLevelExam.Text = "Early high school"
233     Case 11
234         Me.lblGradeLevelExam.Text = "Late high school"
235     Case 12
236         Me.lblGradeLevelExam.Text = "Final exam"
237     Case Else
238         Me.lblGradeLevelExam.Text = "Invalid grade level"
239 End Select

```

```

230 Select Case intDepartmentNumber
231     Case 1, 3, 8, 11, 17
232

```

```

234 Select Case intDepartmentNumber
235     Case 2, 4, 7, 12 To 14, 22

```

68

Chapter 5: Mobile Applications Using Decision Structures

Code Snippets

- Right-click the line in the code editing window where you want to insert the snippet
- Click Insert Snippet on the shortcut menu
- Double-click Common Code Patterns
- Double-click the Conditionals and Loops
- Double-click the If...Else...End If Statement code snippet

71

Chapter 5: Mobile Applications Using Decision Structures

Selecting Which Decision Structure to Use

- Select Case statement is most useful when more than two or three values must be tested for a given variable
- The If...Then...ElseIf statement is more flexible
 - More than one variable can be used in the comparison
 - Compound conditions with the And, Or, and Not logical operators can be used

69

Chapter 5: Mobile Applications Using Decision Structures

Validating Data

- Developers should anticipate that users will enter invalid data
- Developers must write code that will prevent the invalid data from being used in the program to produce invalid output

72

Chapter 5: Mobile Applications Using Decision Structures

Testing Input to Determine If the Value Is Numeric


- The Visual Basic **IsNumeric** function can check the input value to determine if the value can be converted into a numeric value such as an Integer or Decimal data type

```

If IsNumeric(Me.txtLinearFeet.Text) Then
    Statement(s) executed when condition is true
Else
    'Display MessageBox if user entered nonnumeric value
    MessageBox.Show("Enter the Linear Feet of the Cabinets.", _
        "Input Error")
    Me.txtLinearFeet.Text = ""
    Me.txtLinearFeet.Focus()
End If
    
```

73 Chapter 5: Mobile Applications Using Decision Structures

Deploying the Application



76 Chapter 5: Mobile Applications Using Decision Structures

Checking for a Positive Number

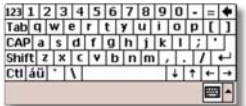
```

24 If IsNumeric(Me.txtLinearFeet.Text) Then
25     decLinearFeet = Convert.ToDecimal(Me.txtLinearFeet.Text)
26
27     ' Is linear feet greater than zero
28     If decLinearFeet > 0 Then
29         Statement(s) executed when condition is true
30     Else
31         ' Display MessageBox if a negative value is entered
32         MessageBox.Show("You entered " & decLinearFeet.ToString() & _
33             " - Since a Number Greater Than Zero.", "Input Error")
34         Me.txtLinearFeet.Text = ""
35         Me.txtLinearFeet.Focus()
36     End If
37 Else
38     'Display MessageBox if user entered nonnumeric value
39     MessageBox.Show("Enter the Linear Feet of the Cabinets.", _
40         "Input Error")
41     Me.txtLinearFeet.Text = ""
42     Me.txtLinearFeet.Focus()
43 End If
    
```

74 Chapter 5: Mobile Applications Using Decision Structures

Using the Input Panel

- When you use the emulator, you can enter data directly from the keyboard
- The Pocket PC has the **input panel** to enter data into applications
 - You can use a stylus to select the characters from the input panel.
 - When you press the stylus on a character in the input panel, the character is entered into the focused object on the form



77 Chapter 5: Mobile Applications Using Decision Structures

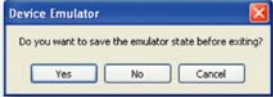
Deploying the Application

- Click the Start Debugging button on the Standard toolbar
- If necessary, select Pocket PC 2003 SE Emulator in the Device list. Click the Deploy button

75 Chapter 5: Mobile Applications Using Decision Structures

Closing the Emulator

- When you are finished with the application, close the emulator by clicking the Close button (X) in the upper-right corner of the Pocket PC emulator
- It is **critical** that you click the No button



78 Chapter 5: Mobile Applications Using Decision Structures

Summary

- Write mobile applications
- Using new objects
 - Panel
 - RadioButton
- Messages
- Conditional statements
- Case selection

79

Chapter 5: Mobile Applications Using Decision Structures

CHAPTER 5 COMPLETE

Mobile Applications
Using Decision Structures