



Select Case block

- A decision-making structure that simplifies choosing among several actions.
- Avoids complex nested If constructs.
- If blocks make decisions based on the truth value of a condition; Select Case choices are determined by the value of an expression called a **selector**.



Select Case Terminology

Each of the possible actions is preceded by a clause of the form

Case *valueList*

where *valueList* itemizes the values of the **selector** for which the action should be taken.



Lab Sheet 5.6: Form

The screenshot shows a Java Swing window titled "Horse Race". Inside the window, there is a label "Finishing position (1, 2, 3, ...)" followed by a small text input field. An arrow points from the label "txtPosition" to this input field. Below the input field is a button labeled "Evaluate Position". At the bottom of the window is a larger text area. An arrow points from the label "txtOutcome" to this text area.



Lab Sheet 5.6: Code

```
Private Sub btnEvaluate_Click(...) _  
                                Handles btnEvaluate.Click  
    Dim position As Integer = CInt(txtPosition.Text)  
    Select Case position  
        Case 1  
            txtOutcome.Text = "Win"  
        Case 2  
            txtOutcome.Text = "Place"  
        Case 3  
            txtOutcome.Text = "Show"  
        Case 4, 5  
            txtOutcome.Text = "You almost placed in the money."  
        Case Else  
            txtBox.Text = "Out of the money."  
    End Select  
End Sub
```

Diagram illustrating the code structure:

- A yellow box labeled **Selector** has an arrow pointing to the `position` variable in the `Select Case` statement.
- A yellow box labeled **Value Lists** is positioned to the right of the `Case` statements, indicating the values being compared.



Lab Sheet 5.6: Output

A screenshot of a Windows-style dialog box titled "Horse Race". The dialog has a blue title bar with standard minimize, maximize, and close buttons. The main area has a light beige background. It contains a label "Finishing position (1, 2, 3, ...):" followed by a text input field containing the number "2". Below this is a button labeled "Evaluate Position". At the bottom is a text input field labeled "Place".

Horse Race

Finishing position (1, 2, 3, ...): 2

Evaluate Position

Place



Lab Sheet 5.6: Code

```
Private Sub btnEvaluate_Click(...) _  
    Handles btnEvaluate.Click  
    Dim position As Integer = CInt(txtPosition.Text)  
    Select Case position  
        Case 1 To 3  
            txtOutcome.Text = "In the money. Congratulations"  
        Case Is >= 4  
            txtOutcome.Text = "Not in the money."  
    End Select  
End Sub
```



Lab Sheet 5.6: Output

A screenshot of a Windows-style application window titled "Horse Race". The window has a blue title bar with standard minimize, maximize, and close buttons. The main area has a light beige background. It contains a label "Finishing position (1, 2, 3, ...):" followed by a text input box containing the number "2". Below the input box is a button labeled "Evaluate Position". At the bottom of the window is a text box containing the message "In the money. Congratulations.".

Horse Race

Finishing position (1, 2, 3, ...):

Evaluate Position

In the money. Congratulations.



Select Case Syntax

The general form of the Select Case block is

Select Case *selector*

Case *valueList1*

action1

Case *valueList2*

action2

Case Else

action of last resort

End Select



Rules for Select Case

- Case Else (and its action) is optional
- Each value list contains one or more of the following types of items separated by commas:
 1. a literal;
 2. a variable;
 3. an expression;
 4. an inequality sign preceded by *Is* and followed by a literal, variable, or expression;
 5. a range expressed in the form *a To b*, where *a* and *b* are literals, variables, or expressions.



Flowchart for Select Case

