

ความรู้พื้นฐานของ Visual Basic 2005

การใช้ไดอะล็อกบ็อกซ์

- ไดอะล็อกบ็อกซ์ (Dialog Box) เป็นกล่องข้อความที่ใช้ในการโต้ตอบหรือแสดงข้อความให้ผู้ใช้งานทราบ
- ไดอะล็อกบ็อกซ์เบื้องต้นมี 2 แบบคือ
 1. MessageBox
 2. InputBox

MessageBox

- MessageBox ใช้สำหรับแสดงข้อมูลให้ผู้ใช้ทราบ แล้วให้ผู้ใช้เลือกกระทำอย่างใดอย่างหนึ่ง เช่น การยืนยันการปิดโปรแกรม หรือเตือนให้ผู้ใช้ทราบเมื่อเกิดเหตุการณ์ผิดพลาด เป็นต้น
- MessageBox เป็นออบเจกต์ การใช้งานต้องเรียกใช้ผ่านเมธอด Show

MessageBox

- รูปแบบการใช้งาน MessageBox เบื้องต้น คือ
`MessageBox.Show(Text [,Caption] [,Buttons] [,Icon] [,DefaultButton])`
 - Text คือข้อความที่แสดงใน MessageBox
 - Caption คือข้อความบนแถบไตเติลบาร์ของ MessageBox
 - Buttons คือปุ่มที่จะให้ผู้ใช้เลือกหลังจากอ่านข้อความแล้ว
 - Icon คือไอคอนที่แสดงประกอบกับข้อความ (แสดงถึงอารมณ์ของข้อความนั้น)
 - DefaultButton เป็นการกำหนดปุ่มดีฟอลต์ (ถ้ากดปุ่ม Enter จะเท่ากับการคลิกปุ่มนี้) ว่าเป็นปุ่มที่เท่าไร (นับจากซ้ายมาขวา)

InputDialog

- InputBox เป็นฟังก์ชัน ใช้สำหรับรับข้อมูลจากผู้ใช้ โดยผู้ใช้ต้องกรอกข้อมูลที่ต้องการ แล้วคลิกปุ่ม OK เช่น การขอชื่อผู้ใช้งาน เป็นต้น
- ผลลัพธ์ที่ได้จากการเรียกใช้ฟังก์ชันนี้ คือ ข้อความที่ผู้ใช้ป้อน ซึ่งสามารถนำไปใช้งานในส่วนอื่นของโปรแกรมได้

InputDialog

- รูปแบบการใช้งาน InputBox คือ
`var_string = InputBox (Prompt [, Title] [, Default, xPos, yPos])`
 - var_string คือตัวแปรที่ใช้เก็บข้อความที่ป้อนเข้ามา
 - Prompt คือข้อความที่ปรากฏใน InputBox ซึ่งมักเป็นคำอธิบายถึงข้อมูลที่ต้องการให้ผู้ใช้กรอก
 - Title คือข้อความในแถบไตเติลบาร์ของ InputBox
 - Default คือค่าดีฟอลต์ (กรณี que ผู้ใช้ไม่ป้อนอะไรมา ก็จะได้มีข้อมูลนำไปใช้)
 - xPos กับ yPos เป็นพิกัดที่จะให้แสดง InputBox บนจอภาพ

ข้อมูลและตัวแปร (Data and Variable)

Data and Variable

- ในการสร้างแอปพลิเคชันมักมีการใช้ ข้อมูล (Data) รูปแบบต่างๆ ไม่ว่าจะเป็นข้อความ, ตัวเลข, วันที่ หรือเวลา เป็นต้น โดยข้อมูลชนิดต่างๆ นั้นจะถูกนำมาคำนวณ เปลี่ยนแปลง หรือประมวลผลในรูปแบบต่างๆ ซึ่งในขณะนั้นเรียกว่า ตัวแปร (Variable) ซึ่งทำหน้าที่เก็บค่าของข้อมูลที่เราต้องการใช้งาน
- ดังนั้น *ตัวแปร* จึงเป็นหน่วยความจำที่ใช้เก็บข้อมูลเพื่อการประมวลผล ในระหว่างการทำงานของแอปพลิเคชัน

ชนิดข้อมูล

- VB 2005 แบ่งข้อมูลเป็นกลุ่มๆ ดังนี้
 1. ข้อมูลตัวเลข เป็นข้อมูลที่เก็บตัวเลขชนิดต่างๆ ทั้งจำนวนเต็มและทศนิยม
 2. ข้อมูลข้อความ เป็นกลุ่มข้อมูลที่เก็บตัวอักษรทั้งอักษรตัวเดียวและข้อความ
 3. ข้อมูลวันที่เวลา เก็บข้อมูลวันเดือนปีและเวลา โดยเป็นข้อมูลที่มีลักษณะพิเศษคือ มีทั้งส่วนที่เก็บเป็นข้อมูลตัวเลขและตัวอักษร ซึ่งผู้ใช้สามารถเลือกรูปแบบได้ตามที่ต้องการ
 4. ข้อมูลตรรกศาสตร์ ใช้เก็บข้อมูลด้านตรรกศาสตร์ คือค่าจริง และเท็จ
 5. ข้อมูลออบเจกต์ เป็นข้อมูลที่สามารถเก็บข้อมูลชนิดอื่นๆ ไปได้

ชนิดข้อมูล	ขนาดหน่วยความจำ	ค่าที่เก็บในตัวแปร
Boolean	2 byte	มี 2 ค่า คือ True และ False
Byte	1 byte	0 ถึง 255 (หรือเรียกว่า Unsigned Byte)
SByte	1 byte	-128 ถึง 127 (หรือเรียกว่า signed Byte)
Char	2 byte	ตัวอักษรหนึ่งตัว ซึ่งเก็บแบบ Unicode (หรือเรียกว่า Unsigned Char)
Short	2 byte	-32,768 ถึง 32,767
UShort	2 byte	0 ถึง 65,535 (หรือเรียกว่า Unsigned Short)

ชนิดข้อมูล	ขนาดหน่วยความจำ	ค่าที่เก็บในตัวแปร
Integer	4 byte	-2,147,483,648 ถึง 2,147,647
UInteger	4 byte	0 ถึง 4,294,967,295 (หรือเรียกว่า Unsigned Integer)
Long	8 byte	-9,223,372,036,854,775,808 ถึง 9,223,372,036,854,775,807
ULong	8 byte	0 ถึง 18,446,774,073,551,615 (หรือเรียกว่า Unsigned Long)
Decimal	16 byte	เลขจำนวนเต็มและทศนิยม ซึ่งมีความละเอียดแล้วแต่จะกำหนด
String	แปรผันตามความยาวของข้อมูล	ข้อความ ซึ่งมีความยาวตั้งแต่ 0 ถึง 2 พันล้านตัวอักษร (เป็นชนิด Unicode Character)

ชนิดข้อมูล	ขนาดหน่วยความจำ	ค่าที่เก็บในตัวแปร
Single	4 byte	เลขทศนิยมระหว่าง -3.402823E38 ถึง -1.401298E-45 สำหรับค่าลบ และ 1.401298E-45 ถึง 3.402823E38 สำหรับค่าบวก
Double	8 byte	เลขทศนิยมระหว่าง -1.79769313486232E308 ถึง -4.94065645841247E-324 สำหรับค่าลบ และ 4.94065645841247E-324 ถึง 1.79769313486232E308 สำหรับค่าบวก
Date	8 byte	วันเวลาตั้งแต่วันที่ 1 มกราคม ค.ศ.0000 เวลา 00:00:00 ถึง 31 ธันวาคม 9999 เวลา 23:59:59
Object	4 byte	เป็นข้อมูลชนิดออบเจกต์ ซึ่งมีหลายรูปแบบ

การประกาศตัวแปร

- การประกาศตัวแปร (Variable Declaration) เป็นการบอกให้คอมพิวเตอร์ทราบว่าเรามีตัวแปรชื่ออะไรบ้าง แต่ละตัวแปรเป็นตัวแปรชนิดใด
- การประกาศตัวแปรเป็นเรื่องที่ต้องพิจารณาให้เหมาะสมกับข้อมูลที่ใช้ งาน เพื่อลดข้อผิดพลาดและเพิ่มประสิทธิภาพการทำงานของโปรแกรมที่เขียน
- รูปแบบการประกาศ คือ

Dim Variable_Name As TypeOfVariable

- *Dim* คือ คำสั่งสำหรับการประกาศค่าตัวแปร
- *Variable_Name* คือ ชื่อตัวแปร
- *TypeOfVariable* คือ ชนิดตัวแปร

หลักการตั้งชื่อตัวแปร

1. ต้องขึ้นต้นด้วยตัวอักษรเท่านั้น
2. ตัวอักษรถัดไปจะเป็นตัวเลขหรือตัวอักษรก็ได้
3. ห้ามตั้งชื่อตัวแปรซ้ำกัน และห้ามซ้ำกับคำสงวน (Reserved Word) ซึ่งเป็นคำสั่งของ VB 2005
4. อักษรตัวเล็กหรือตัวใหญ่ไม่มีผลใดๆ (Non-Case Sensitive) เช่น ตั้งชื่อว่า Name, NAME, name ทั้ง 3 ชื่อจะถือเป็นตัวแปรเดียวกัน โดย Code Editor จะแก้ไขให้ตรงกับคำสั่งที่ Dim ไว้

การระบุชนิดข้อมูลไว้ในชื่อตัวแปร

- เป็นวิธีที่โปรแกรมเมอร์นิยมใช้ คือ ระบุชนิดข้อมูลของตัวแปรไว้ในชื่อตัวแปรด้วย โดยใช้ตัวย่อของชื่อชนิดข้อมูลนำหน้าชื่อตัวแปร (Prefix)

ชนิดข้อมูล	Prefix	ตัวอย่าง
Integer	int	intSalary, intQuality
Single	sng	sngTaxRate, sngPrice
Double	dbl	dblFactor, dblArea
String	str	strName, strAddress
Decimal	dec	decSpeed, decBenefit
Boolean	bln	blnGender, blnMarried
Date	dat	datBirthDate, datExpired

การประกาศและใช้งานตัวแปรใน VB 2005

- ตัวอย่างการประกาศและใช้งานตัวแปร

```

Dim strName As String
Dim intSalary As Integer
Dim bytAge As Byte = 24
    
```

ประกาศตัวแปร

ประกาศตัวแปรพร้อมกับกำหนดค่าตัวแปร

```

strName = "Big"
intSalary = 25000
    
```

กำหนดค่าตัวแปรหลังการประกาศตัวแปร

```

MessageBox.Show("ตัวแปร strName = " & strName)
MessageBox.Show("ตัวแปร intSalary = " & intSalary.ToString & _
Chr(13) & "ตัวแปร bytAge = " & bytAge.ToString)

Me.Close()
    
```

แสดงค่าตัวแปร โดยใช้ MessageBox

การประกาศและใช้งานตัวแปรใน VB 2005

- ตัวอย่างการประกาศพร้อมกับกำหนดค่าตัวแปร

```
Dim strFirstName As String = "John"  
Dim datBirthDay As Date = #1/1/2001#  
Dim blnMarried As Boolean = False
```

- ไม่จำเป็นต้องประกาศตัวแปรที่ตอนต้นของโปรแกรม

การใช้งานค่าคงที่

- ค่าคงที่ (Constant) คือ ข้อมูลที่มีค่าคงที่ไม่เปลี่ยนแปลง เช่น ค่า PI สำหรับคำนวณพื้นที่วงกลม เป็นต้น

- รูปแบบการประกาศค่าคงที่ คือ

```
CONST NameOfConstant As TypeOfConstant = Constant_Value
```

- CONST คือ คำสั่งที่ใช้ประกาศค่าคงที่
- *NameOfConstant* คือ ชื่อค่าคงที่
- *TypeOfConstant* คือ ชนิดข้อมูลของค่าคงที่
- *Constant_Value* คือ ค่าที่ระบุให้กับค่าคงที่

การใช้งานค่าคงที่

- ชื่อค่าคงที่ใน VB มักนิยมใช้ตัวอักษรพิมพ์ใหญ่ทั้งหมด โดยต้องกำหนดค่าคงที่นั้นพร้อมกับการระบุชนิดของค่าคงที่ และการประกาศค่าคงที่มักจะประกาศไว้ตอนต้นของโปรแกรม เพื่อให้มองเห็นและใช้งานค่าคงที่ได้ง่าย

การใช้งานตัวดำเนินการ

- ตัวดำเนินการ (Operator) เป็นสัญลักษณ์สำหรับสั่งให้คำนวณ หรือหาผลลัพธ์ที่เกิดขึ้นจากการกระทำในแต่ละตัว เช่น $a + b$ มีตัวดำเนินการคือ $+$ ทำการบวกค่าตัวแปร a กับ b เข้าด้วยกัน เป็นต้น
- คำสั่งที่มีตัวดำเนินการประกอบด้วยข้อมูล 2 ส่วน โดยมีตัวดำเนินการคั่นกลาง ซึ่งเมื่อมีตัวดำเนินการก็หมายความว่า จะต้องได้ผลลัพธ์การทำงานของตัวดำเนินการออกมาเสมอ เรียกส่วนคำสั่งมีตัวดำเนินการคั่นกลางข้อมูลสองส่วนนี้ว่า **Expression**
- ตัวดำเนินการมี 3 ประเภทคือ Comparison Operator, Arithmetic Operator และ Logical Operator

Comparison Operator

- คือ ตัวดำเนินการเปรียบเทียบ ใช้สำหรับเปรียบเทียบข้อมูลสองด้านของตัวดำเนินการ ซึ่งผลการเปรียบเทียบจะได้เป็นค่าจริง (True) หรือเท็จ (False)

ตัวดำเนินการ	คำอธิบาย	ตัวดำเนินการ	คำอธิบาย
<	น้อยกว่า	=	เท่ากับ
<=	น้อยกว่าหรือเท่ากับ	Is	เป็นออบเจกต์เดียวกัน
>	มากกว่า	IsNot	เป็นออบเจกต์คนละตัว
>=	มากกว่าหรือเท่ากับ	TypeOf...Is	เป็นออบเจกต์ชนิดเดียวกัน
<>	ไม่เท่ากับ	Like	รูปแบบเหมือนกัน

Arithmetic Operator

- คือ ตัวดำเนินการทางคณิตศาสตร์

ตัวดำเนินการ	คำอธิบาย	ตัวอย่าง
^	ยกกำลัง	2^3 เท่ากับ 8
-	Negation	ให้ $a = -3$ เพราะฉะนั้น $-a$ เท่ากับ 3
*	คูณ	$5 * 2$ เท่ากับ 10
/	หาร	$21 / 2$ เท่ากับ 10.5
\	หารแบบได้ผลลัพธ์จำนวนเต็ม	$12 \setminus 5$ เท่ากับ 2

Arithmetic Operator

ตัวดำเนินการ	คำอธิบาย	ตัวอย่าง
Mod	หาเศษจากการหาร	5 Mod 3 เท่ากับ 2
+	บวก	3 + 4 เท่ากับ 7
-	ลบ	9 - 3 เท่ากับ 6
<<	เลื่อนบิตมาทางซ้าย	11001110 << 1 ผลลัพธ์คือ 10011100
>>	เลื่อนบิตมาทางขวา	11001110 >> 1 ผลลัพธ์คือ 01100111
&	String Concatenation	a & b ผลลัพธ์คือ การนำข้อความ b มาต่อท้ายข้อความ a

Logical Operator

- คือ ตัวดำเนินการเปรียบเทียบข้อมูลด้านตรรกศาสตร์ ผลลัพธ์ที่ได้จะเป็นค่าจริง (True) หรือเท็จ (False) ใช้เพื่อการตัดสินใจในการทำงานของโปรแกรม

Logical Operator

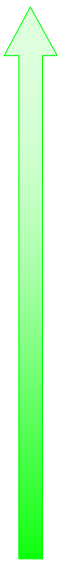
ตัวดำเนินการ	คำอธิบาย	ตัวอย่าง
Not	นิเสธ	Not a ผลลัพธ์คือ นิเสธของ a
And	และ	a And b ผลลัพธ์เป็นจริง เมื่อ a และ b มีค่าจริง
Or	หรือ	a Or b ผลลัพธ์เป็นจริง เมื่อ a หรือ b มีค่าจริง
Xor	Exclusive Or	a Xor b ผลลัพธ์เป็นจริง เมื่อ a กับ b มีค่าความจริงต่างกัน
AndAlso	และ (แบบรวดเร็ว)	a AndAlso b ผลลัพธ์เป็นจริงเมื่อ a และ b ต่างเป็นจริง
OrElse	หรือ (แบบรวดเร็ว)	a OrElse b ผลลัพธ์เป็นจริง เมื่อ a หรือ b มีค่าจริง

Logical Operator

- จะเห็นว่า AndAlso คล้ายกับ And ส่วน OrElse ก็คล้ายกับ Or แต่ AndAlso กับ OrElse นั้นเป็นตัวดำเนินการที่ช่วยลดเวลาการเปรียบเทียบ โดยใช้หลักว่า a AndAlso b นั้น ถ้าพบว่า a เป็นเท็จแล้วก็ไม่ต้องเปรียบเทียบ b ให้เสียเวลา สรุปได้เลยว่าเป็นเท็จ ส่วน a OrElse b นั้น ถ้าพบว่า a เป็นจริงแล้วก็ไม่ต้องเปรียบเทียบ b ให้เสียเวลา สรุปได้เลยว่าเป็นจริงแน่นอน ซึ่งตัวดำเนินการนี้จะช่วยลดเวลาในกรณีที่ a หรือ b เป็นข้อมูลขนาดใหญ่ หรือมีความซับซ้อนมาก

Operator Precedence

- คือ ลำดับความสำคัญของตัวดำเนินการ
- ในคำสั่งใช้งานตัวดำเนินการ 1 ตัว ปกติแล้วจะใช้วงเล็บล้อมคู่ของข้อมูลไว้ การประมวลผลจะเริ่มจากวงเล็บคู่ในสุดก่อนแล้วเรียงลำดับออกมาจนถึงวงเล็บนอกสุด แต่ถ้าไม่มีการใช้วงเล็บแยกตัวดำเนินการ ผู้เขียนโปรแกรมจำเป็นต้องคำนึงถึงลำดับความสำคัญของตัวดำเนินการ จึงจะทราบผลลัพธ์ที่ถูกต้องได้

ลำดับความสำคัญ	ตัวดำเนินการ
 <div style="display: flex; justify-content: space-between; width: 100%;"> สูง ต่ำ </div>	^
	* , /
	\
	Mod
	+ , -
	&
	<< , >>
	= , <> , < , <= , > , >= , Like , Is , IsNot , TypeOf...Is
	Not
	And , AndAlso
	Xor , Or , OrElse

การแปลงชนิดข้อมูล

- เนื่องจากข้อมูลแต่ละชนิดเป็นออบเจกต์ จึงไม่สามารถนำมาใช้งานร่วมกันได้โดยตรง แต่ต้องนำมาแปลงให้เป็นข้อมูลชนิดเดียวกันก่อนจึงสามารถใช้งานได้
- การแปลงชนิดข้อมูลมี 2 วิธี คือ
 1. แปลงชนิดข้อมูลด้วย Data Type Conversion Function (แสดงในตาราง)
 2. แปลงชนิดข้อมูลด้วยฟังก์ชัน CType
มีรูปแบบดังนี้
`CType(expression, typename)`
โดย typename คือ ชื่อชนิดข้อมูลที่ต้องการแปลง

Data Type Conversion Function

ฟังก์ชัน	คำอธิบาย
CBool	แปลงข้อมูลให้เป็นข้อมูลชนิด Boolean
CByte	แปลงข้อมูลให้เป็นข้อมูลชนิด Byte
CChar	แปลงข้อมูลให้เป็นข้อมูลชนิด Char
CDate	แปลงข้อมูลให้เป็นข้อมูลชนิด Date
CDbl	แปลงข้อมูลให้เป็นข้อมูลชนิด Double
CDec	แปลงข้อมูลให้เป็นข้อมูลชนิด Decimal
CInt	แปลงข้อมูลให้เป็นข้อมูลชนิด Integer
CLng	แปลงข้อมูลให้เป็นข้อมูลชนิด Long Integer

Data Type Conversion Function

ฟังก์ชัน	คำอธิบาย
CObj	แปลงข้อมูลให้เป็นข้อมูลชนิด Object
CByte	แปลงข้อมูลให้เป็นข้อมูลชนิด SByte
CShort	แปลงข้อมูลให้เป็นข้อมูลชนิด Short
CSng	แปลงข้อมูลให้เป็นข้อมูลชนิด Single
CStr	แปลงข้อมูลให้เป็นข้อมูลชนิด String
CUInt	แปลงข้อมูลให้เป็นข้อมูลชนิด UInteger
CULng	แปลงข้อมูลให้เป็นข้อมูลชนิด ULong
CUShort	แปลงข้อมูลให้เป็นข้อมูลชนิด UShort