

Visual Studio_최준용

1주차) 사이즈조절

창을 작게>>디버깅>>디버깅시작(F5:로그에 오류가 있는지 확인하고 실행파일을 만드는 버튼)
>>오류가 없는 경우: 좌측 하단 진행창에 “빌드 시작...”
>>오류가 있는 경우: 다른 창들(진단 도구, 솔루션 탐색기, 속성, 직접실행창)이 뜸
>>디버깅 이후 bin>Debug>net5.0-windows에서 “사이즈변경” 응용프로그램 생성 확인
>>더블클릭 시 만든 창이 실행(비주얼 스튜디오 없이도 실행시킬 수 있는 프로그램이 생성)

2주차)

***주의점:파일을 실행시킬때는 상관없지만, 수정할 경우 이름, 위치를 절대 변경해서는 안된다.**

1.비주얼 베이직 프로그램의 구조, 작성과정

-비주얼 베이직의 구조: 시각적 요소(Vidual)+명령코드(Basic)
-작성과정: 프로젝트 시작>>폼 디자인,개체 속성 설정>>코드 작성>>프로젝트 실행 및 저장
*모든 Windows Forms: 모든 요소, 공용 컨트롤: 많이쓰는거, 컨테이너: 묶는거
*속성: control마다 각각의 속성을 속성창을 통해 알 수 있다.
>>코드를 짤 때는 항상 “컨트롤 이름.속성(ex.text, font...)”이 짝으로 입력되어야 한다.

2.“Hello!!!”메세지 출력 프로그램 작성

새 프로젝트>>Windows앱>>*Textbox클릭>>원하는 위치에 옮겨 드래그하여 크기조절

*Textbox(입출력 모두 가능)

>Text box 1개, button 2개 생성>>2개의 버튼 크기, 위치 설정(버튼 2개 선택: 흰색-기준)>>서식에서 크기 먼저, 이후 맞춤설정>>Form 속성에서 text에 “인사프로그램” 입력
>>버튼1 의 text에 “출력”입력>>버튼2 의 text에 “종료”입력
>>버튼 속성에서 폰트 수정(font옆 ...버튼 눌러서 수정)>>버튼 속성 Fore color에서 색 변경
>>종료 버튼 설정(더블클릭 후 End입력)>>이후 디버깅 시작(F5) 누르고 종료버튼 누를시 종료확인>>버튼1 더블클릭 후 TextBox1.Text="hello"입력>>F5>>출력 해보기(hello)

*글자(text) 속성

text=컨트롤에 표시되는 문자열을 나타내는 속성
font=글꼴, 글꼴스타일, 크기, 효과 등 설정
forecolor=문자열의 색상을 바꾸어 주는 속성
BackColor=전체 배경색을 지정하는 속성‘사용자 지정’,‘웹’,‘시스템’
textAlign=문자열의 위치를 설정하는 속성

*대입연산자(=)

오른쪽에 있는 값을 왼쪽에 대입하는 연산자 ex)TextBox1.Text="hello"

*여러줄 입력하고 싶을 때(Multiline)

multiline=false>>true로 변경 시 가능

*비밀번호처럼 지정한 문자를 특수문자로 출력하는 기능>PasswordChar)

```
>>새 프로젝트>>Windows앱>>*Textbox클릭>>원하는 위치에 옮겨 드래그하여 크기조절
>>text box1개, button 4개 생성>>텍스트박스 속성(Text:이름 또는 학번 출력)
>>버튼 text(1:이름, 2:학번, 3:삭제, 4:종료)>>모두 선택 후 font변경>
버튼 back color, text color변경>>텍스트박스의 텍스트 중앙으로 정렬(TextAlign:Center)
>>버튼의 경우 TextAlign의 선택지가 다름>>종료버튼 더블클릭 후 “End”입력
>>로그입력창 상단에서 객체(ex.버튼)와 이벤트 발생조건(ex.더블클릭)을 수정할 수 있음
>>버튼1의 로그에 TextBox1.Text = "최주봉" 입력 후 아랫줄에 Button1.Text = "최주봉"입력.
(이 때 줄 순서에 따라 순차적으로 이벤트가 실행되므로 주의해야함)
>>삭제버튼 더블클릭후 TextBox1.Text = " " 입력, 다음줄에 “Button1.Text = "이름””입력
>>학번버튼 더블클릭 후 TextBox1.Text = "1730648" 입력
>>F5눌러서 프로그램 실행시 작동되는 것을 확인가능
```

```

새 프로젝트>>Windows앱>>*Textbox클릭>>원하는 위치에 옮겨 드래그하여 크기조절
>>text box2개, button 4개 생성>>텍스트박스 속성(Text:이름 또는 학번 출력)
>>버튼 text(1:이름, 2:학번, 3:삭제, 4:종료)>>버튼, 텍스트박스 정렬>>라벨 2개생성 후 정렬
>>라벨 속성에서 오토사이즈 False>>폰트 크기상관없이 라벨크기 수정 가능
>>라벨폰트(16,bold로 수정)>>라벨1 텍스트: 입력값, 라벨2 텍스트: 출력값
>>버튼 1에 TextBox2.Text = "안녕하세요" *& TextBox1.Text & " 입니다." 입력 (텍박1에 있
는거 2에 입력하는 기능. 연결연산자는 문자와 컨트롤을 연결)
>>입력값에 이름 입력 후 이름 버튼 누르면 출력값에 안녕하세요 '이름'입니다. 나옴
>>버튼 3에 TextBox1.Text = " "
        TextBox2.Text = " " 입력.
>>버튼 2 에 TextBox2.Text= Textbox1.Text 입력>>F5

```

ex) “안녕하세요” & 컨트롤>>안녕하세요 ‘입력값’

3주차)

-변수와 데이터형

-변수: 변하는 값을 저장하는 공간. 프로그램이 실행되는 동안 사용자가 입력한 값이나 계산한 출력값들을 저장하는 기억공간.

-변수 선언: A, B, C 라는 공간에 각각 20, 30, 100을넣는다고 가정. A~C와 같은 공간을 만들어 주는 기능. 프로그램을 작성하다가 기억공간이 필요하면 ***변수의 이름과 크기**를 컴퓨터에 알려주고, 컴퓨터는 요청받은 크기만큼 기억공간을 확보하고 시작주소를 프로그램에 알려준다. 그러면 프로그램은 할당된 시작 주소로 선언된 변수명을 매칭하여 사용한다.

***변수 선언 형식:** 변수의 크기를 선언할 때는 직접 몇 바이트인지 입력하지 않고 해당 데이터형을 입력한다.

>>형식: Dim 변수명 As 데이터형(크기)

ex) Dim score As integer/Dim name As string

↳컴퓨터는 1행, 2행... 순차적인 흐름을 가지는데 Dim을 입력시 이를 저장하는 저장공간을 생성한다. 이는 변수가 되고 이름이 필요하다.

***변수명 작성 규칙**

- 1) 영문자,한글,숫자, 밑줄(_)을 조합하여 사용할 수 있으며, 다른 특수기호는 사용할 수 없다.
- 2) 첫글자는 반드시 영문자/한글이어야 하며, 기호는 사용할 수 없다.
- 3) 예약어(함수, 명령어-로그에서 파란색으로 표시: Sub, Private, End...)는 사용할 수 없다.
- 4) 255자를 넘지 않아야 한다.
- 5) 대/소문자를 구분하지 않는다.

***변수 선언의 예**

올바른 변수 선언	잘못된 변수 선언
<pre>Dim student_name As String Dim score As Byte Dim 학과 As String Dim kor, mat, eng As Integer</pre>	<pre>Dim name\$ As String >> 불가한 특수 기호(\$) 사용 Dim 123_adderess As String >> 숫자로 시작 Dim Private As Integer >> 예약어 사용(Private)</pre>

>>위 규칙에 어긋나지 않는 선에서 변수명을 사용할 수 있다.

***데이터형: 변수가 몇 바이트의 기억공간을 차지해야 하는지, 그 기억 공간에 어떤 형태로 저장되는지 결정한다.**

데이터형	크기 (바이트)	표현 범위와 설명	정수/ 실수	Decimal	16	0~+/-79,228,162,514,264,337,593,543,950,335(정수) 0~+/-7,922,816,251,426,433,759,354,395,033,35E28(실수)
정수	Byte	0~255(부호 없는 정수)	실수	Single	4	음수: -3.4028235E38~-1.401298E-45 양수: 1.401298E-45~3.4028235E38
	UShort	0~65,535(부호 없는 정수)		Double	8	음수: -1.79769313486232E308~-4.94065645841247E-324 양수: 1.79769313486232E308~4.94065645841247E-324
	UInteger	0~4,294,967,295(부호 없는 정수)	문자	Char	2	한 유니코드 문자 표현(0~65535의 매핑되는 문자 표현 가능)
	ULong	0~18,446,744,073,709,551,615(부호 없는 정수)		String	4	문자열 표현(약 20억 개의 유니코드 문자)
	SByte	-128~127	기타	Boolean	2	True 또는 False 표현. 기본 값은 False
	Short	-32,768~32,767		Date	8	1년 1월 1일 00:00~9999년 12월 31일 11:59:59 PM
	Integer	-2,147,483,648~2,147,483,647	Object	Object	4	모든 형식 저장 가능
	Long	-9,223,372,036,854,775,808~9,223,372,036,854,775,807				

숫자 데이터형을 결정할 때는 정수-Integer/UInteger/Long만, 실수-Single/Double

-가장 많이 쓰는 정수형태 데이터형-

- 1.Integer(4바이트): 부호가 있는 짧은 정수 입력시 사용. 표현범위는 -32,768~32,767
- 2.Uninteger(4바이트): 부호가 없는 짧은 정수를 입력시 사용. 표현범위는 0~4,294,967,295
Ex) 시험점수를 1~100점까지 입력할 때는 부호가 필요x
>>부호를 안쓰면서 범위가 넓은 Uninteger를 사용한다.
- 3.Long(8바이트): 부호가 있으면서 긴 정수를 입력시 사용.
Ex) 50!(팩토리얼)과 같이 값의 크기가 큰 경우
>>편리하지만 프로그램의 크기가 커지기 때문에 효율성 있게 사용해야한다.

-가장 많이 쓰는 실수형태 데이터형-

- 1.Single(4바이트): 소수점이 있는 경우, 범위는
음수: -3.402823E38~ -1.401298E, 양수: 1.401298E
- 2.Double(8바이트): 소수점이 있는 경우, 범위는
음수: -1.79769313486232E308~ -4.94065645841247E-324
양수: 1.79769313486232E308~ 4.94065645841247E-324
>>예를 들어 1.5를 입력시 용량 효율을 위해 Single을 사용해야한다.

-가장 많이 쓰는 문자형태 데이터형-

- 1.Char(2바이트): 한 유니코드 문자 표현시 사용.
- 2.String(4바이트): 문자열을 표현할 때 사용.

-기타-

- Boolean(2바이트): 참과 거짓을 분류할 때 True/False를 선택.

-QUIZ-

150~300의 값을 입력 시 변수선언은 어떻게 해야 하는가?

-Answer: 양수이며 정수이고, 범위를 만족하는 데이터형: Uninteger

>>변수선언은 Dim 주노미 As Uninteger (주의: 변수명(주노미)는 한영역에서 중복 불가능)

*변수의 동시 선언

1) Dim 주, 뇨, 미 As Integer 와 같이 여러개의 변수명을 한번에 입력가능.

>>주, 뇨, 미 모두 Integer형이다.

2) Dim 주, 뇨, 미 As Integer, 파 as Single.

>>주, 뇨, 미는 Integer형이고, 파는 Single 형이다.

>>>>근데 2번 쓰는거 나중에 버그잡기 어려워서 비추

*데이터형을 미지정하는 경우

- 기본값으로 Object형이 할당된다.

- Object형은 모든 데이터형을 나타낼 수 있어 편리하지만 처리속도가 매우 느려지고, 오류를 찾을 때 매우 힘들어진다. 때문에 알맞은 데이터형을 선언하는 것이 더 효율적이다.

Ex) Dim 주, 뇨, 미는 기본형인 Object형이 자동 할당된다.

>>>사용하는거 비추

*변수에 값을 대입하는법

- (대입)연산자 '=': 변수명 = 값/다른 변수명/컨트롤의 속성값

>> 오른쪽 값만 왼쪽으로 넣어줄 수 있음. 윈)오 안담 ㄴㄴ

변수 종류	대입연산자 적용 예
숫자	price = 1000, amount = 47, total = price*amount >>47000
문자	firstname = "a" (" "가 붙은 경우, 변수가 아닌 문자로 취급된다.) name = "VisualBasic" secondname = Firstname (" "가 없으므로 변수이다.) Firstname은 변수이기 때문에 Secondname의 값은 a로 같아진다. age = TextBox1.text (텍스트박스 1의 문자가 age에 입력)
기타	isEqual = True (여기서의 True는 변수가 아닌 논리값 '참'이다.) isSelected = (10>5) >> 10은 5보다 크다. 라는 논리값이 들어감.

*변수선언을 할 때 바로 값을 대입해 줄 수 있다.

Ex)

-Dim score As Integer = 100

-Dim name As string = "Visual Basic"

-Dim isSelected As Boolean = (10>5)

-대입연산자의 명시적 선언과 묵시적 선언

1)명시적 선언: 변수를 선언하고 대입연산자 사용 (Dim a as @@@ = @@@)

2)묵시적 선언: 변수를 선언하지 않고 대입연산자 사용 (@@@ = @@@)

>>묵시적 선언의 경우 첫 번째로 저장되는 값에 따라 데이터형이 자동으로 결정되며 변수를 선언하지 않기 때문에 오류가 생기면 오류를 찾기 힘들며, 속도 효율이 좋지 않다.

*강제로 명시적 선언을 사용하도록 설정하는법

프로젝트 >> WindowsApplication속성(P) >> 컴파일 >> 컴파일 옵션 첫 번째 메뉴를 On.

실습하기4-1) 변수 선언하고 사용하기

새로만들기>>버튼3개 텍박1개>> 크기 정렬 맞추기>> 폰트 20 >>텍박1에 출력, 2에 종료,
텍박3에 삭제>>텍박1 더블클릭 후 변수선언(Dim x as integer)>> 줄띄운후 x=Textbox1.text
>>줄띄운후 Textbox2.Text = x>> F5시 입력값에 입력한거 출력값에 출력>> 버튼2 더블클릭 후
End입력>> 버튼3(삭제)더블클릭 후 Textbox1.Text = " ">>
줄 띄운 후 Textbox.2.Text = " " >>F5 시 모든 기능 작동

***변수선언 했을 때 변수밀이 초록색인 경우=변수에 값이 저장되어있지 않은 경우**

***focus() 속성:** 텍박 내용 삭제 후 자동으로 다시 텍박1에 입력가능하도록 커서 위치기능
>>textbox1.focus()
변수 선언 및 사용 프로그램실행>> 버튼3(삭제) 더블클릭후 Textbox1/2 코드 아랫줄에
textbox1.Focus()입력

***편집기 글자 크기 및 변경**

-도구>>옵션>>환경>>글꼴 및 색 변경 가능

실습하기4-2) 성적 합계 구하기

새로만들기>> 라벨 5개, 텍박3개, 버튼3개 만들기 후 정렬>> 라벨1:국어, 2:영어, 3:수학,
4:총점, 5:백컬러 설정 후 텍스트 지우기(사라지는것처럼 보이지만 이후 총점 표시되도록함)
>> 버튼1: 총점, 버튼2: 삭제, 버튼3:종료>> 변수선언, 변수입력, 연산자, 출력 순으로 입력>> 버튼1
더블클릭후 변수선언 ㄱ>>Dim 국어, 영어, 수학, 총점 As Integer>> 줄 띄우고 국어
=Textbox1.Text 띄우고 영어= Textbox2.Text 띄우고 수학=Textbox3.Text 입력>>
총점 = 국어 + 영어 + 수학 >> label5에 출력하므로 줄띄워서 Label5.Text = 총점
>>버튼2 더블클릭 후 Textbox1/2/3.text= " ">>버튼3 클릭 후 End입력>>F5

***프레시저:** 한 개체에서 Public Class로 열면 End Class로, Private Sub>End Sub로 닫음

*변수의 수명과 유효범위

변수는 자신을 선언하는 Dim문을 포함한 코드 범위 안에서만 값을 가질 수 있다.

1) 유효범위: 변수가 메모리를 차지하고 값을 유지하는 구간

유효 범위		설명
지역 변수	블록	<p>-선언 시작과 종료문 사이의 구간 Ex) If~Endif, For~Next, Do~Loop, While~EndWhile, Select~Endselecr 등</p> <p>-블록 내에 선언된 변수는 해당 블록 내에서만 사용이 가능하다. Ex) If~EndIf 사이 구간을 블록이라 한다.</p> <pre>If score > 60 Then Dim pass As String pass = "ok" End If</pre>
	프로시저	<p>-프로시저 내에서 선언된 변수는 해당 프로시저 내에서만 사용 가능.</p> <p>-이 프로시저 내에 포함되는 변수를 지역변수 라고함.</p> <p>Ex) Private Sub~End Sub의 구간을 프로시저라고한다.</p> <pre>Private Sub button1_Click~~~ Dim~~~ Dim~~~ If~~~ pass End if ~~~~ Textbox1 " 2 End Sub</pre>

* 지역변수의 사용

- 지역변수는 메모리 반납 여부에 따라 Dim과 Static(정적)변수로 나뉜다.
- Dim 변수: 프로시저 실행이 종료되면 메모리를 반납하고 값이 없어진다.
Ex) 버튼1의 프로시저를 실행시킨 뒤 버튼2의 프로시저 실행하면 이전의 값이 사라진다.
- Static 변수: 메모리를 반납하지 않고 저장된 값을 그대로 유지하므로, 프로시저가 종료되더라도 프로그램을 종료하기 전까지 그 값을 유지해야 될 경우 사용한다.

*지역변수 프로그램

새로만들기(지역변수)>>버튼3개 텍박1개>> 버튼1: Dim문, 버튼2: 종료 버튼3: Static문>>
 버튼1더블클릭 후 Dim x As Integer>> 띄운 후 x=x + 100>> Textbox1.Text=x>>
 버튼3 더블클릭 후 Static x As Integer>>띄운 후 x=x+100>> Textbox1.Text=x>>버튼2에 End입력>>F5시 Dim은 버튼을 눌러도 100만 나오고 Static은 누를때마다 100씩 추가된다.

Quiz)

1) 중간고사 0~100

2) 기말고사 0~100

3) 출석 0~100

4) 과제 0~100

5) 평균을 구하는 프로그램

6) +, -, *(곱하기), / (나누기)

7) $\text{평균} = (\text{중간고사} + \text{기말고사} + \text{출석} + \text{과제}) / 4$

Answer)

Public Class Form1

Private Sub Label5_Click(sender As Object, e As EventArgs) Handles Label5.Click

End Sub

Private Sub Label1_Click(sender As Object, e As EventArgs) Handles Label1.Click

End Sub

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

Dim 중간고사, 기말고사, 출석, 과제, 평균 As Integer


```
중간고사= TextBox1.Text  
기말고사= TextBox2.Text  
출석= TextBox3.Text  
과제= TextBox4.Text  
평균= (중간고사+ 기말고사+ 출석+ 과제) / 4
```

```
Label6.Text = 평균
```

```
End Sub
```

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles  
Button2.Click  
    Label6.Text = " "  
    TextBox1.Text = " "  
    TextBox2.Text = " "  
    TextBox3.Text = " "  
    TextBox4.Text = " "
```

```
End Sub
```

```
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles  
Button3.Click
```

```
End
```

```
End Sub
```

```
End Class
```

4주차)

-연산자: (+, -, *(=x), /(=÷))와 같이 이미 정의된 연산을 수행하는 기호

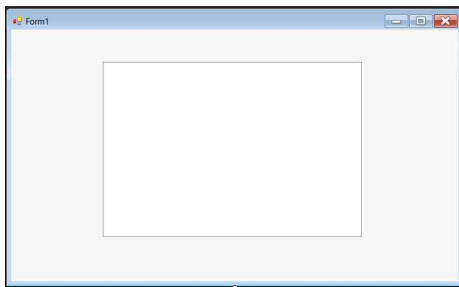
-피연산자: 연산의 대상 또는 연산자의 동작을 받는 변수나 값.

ex) $3=1+2$ (1,2,3-피연산자, $=,+$)연산자)

1.산술연산자: 더하기, 빼기, 곱하기, 나누기 등 계산을 위한 연산자

-산술 연산자의 종류

연산자	일반 형식	의미	사용 예
+	$x+y$	덧셈	$2+5$
-	$x-y$	뺄셈	$5-3$
*	$x*y$	곱셈	$4*6$
/	x/y	나눗셈	$8/3$
^	x^y	거듭제곱	3^4
\ = ₩	$x \setminus y$	x를 y로 나눈 몫의 정수를 구함	$3 \setminus 3$
Mod	$x \text{ Mod } y$	x를 y로 나눈 나머지를 구함	$8 \text{ Mod } 3$



〈산술연산자 프로그램〉

>>비주얼 스튜디오에서 구현시

텍박 하나에 여러 수식을 출력해야함

>>텍박 속성에 "MultiLine=True"로 바꾼다.

>>이후 텍박 확장 가능

>> 텍박 말고 바깥 창 더블클릭

>> Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load

```

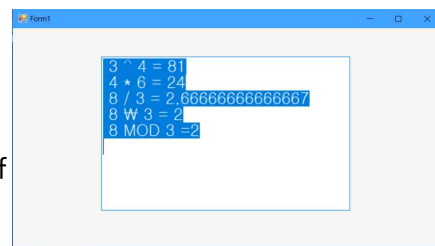
TextBox1.Text = " 3 ^ 4 = " & 3 ^ 4 & vbCrLf &
" 4 * 6 = " & 4 * 6 & vbCrLf &
" 8 / 3 = " & 8 / 3 & vbCrLf &
" 8 ₩ 3 = " & 8 ₩ 3 & vbCrLf &
" 8 MOD 3 = " & 8 Mod 3 & vbCrLf

```

End Sub

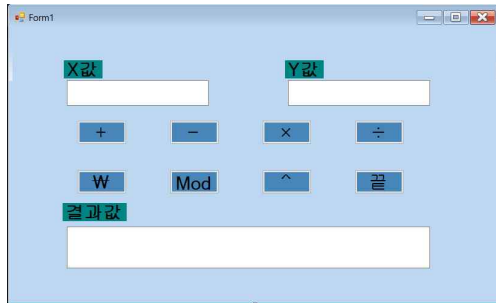
* "3^4="(=텍스트) & 3^4(=산술연산) &

vbCrLf(=비주얼 베이직 상수) &(=다음 연산없을시 사용x)



4주차 논리연산자 과제

텍박 3개 + 라벨 3개 + 버튼 8개)) 텍박3(결과값창) MultiLine=True 로 설정



다음과 같이 7개의 산술연산자 + 종료버튼 생성.

<코드>

Public Class Form1

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

```
    Dim x, y As Long
    x = TextBox1.Text
    y = TextBox2.Text
    TextBox3.Text = "X + Y = " & x + y & vbCrLf
```

```
End Sub
```

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
```

```
    Dim x, y As Long
    x = TextBox1.Text
    y = TextBox2.Text
    TextBox3.Text = "X - Y = " & x - y & vbCrLf
```

```
End Sub
```

```
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles
Button3.Click
```

```
    Dim x, y As Long
    x = TextBox1.Text
    y = TextBox2.Text
    TextBox3.Text = "X × Y = " & x * y & vbCrLf
```

```
End Sub
```

```
Private Sub Button4_Click(sender As Object, e As EventArgs) Handles
Button4.Click
```

```
    Dim x, y As Long
    x = TextBox1.Text
    y = TextBox2.Text
    TextBox3.Text = "X ÷ Y = " & x / y & vbCrLf
```

```
End Sub
```

```
Private Sub Button5_Click(sender As Object, e As EventArgs) Handles
```

```

Button5.Click
    Dim x, y As Long
    x = TextBox1.Text
    y = TextBox2.Text
    TextBox3.Text = "X * Y= " & x * y & vbCrLf
End Sub

Private Sub Button6_Click(sender As Object, e As EventArgs) Handles
Button6.Click
    Dim x, y As Long
    x = TextBox1.Text
    y = TextBox2.Text
    TextBox3.Text = "X Mod Y= " & x Mod y & vbCrLf
End Sub

Private Sub Button7_Click(sender As Object, e As EventArgs) Handles
Button7.Click
    Dim x, y As Long
    x = TextBox1.Text
    y = TextBox2.Text
    TextBox3.Text = "X ^ Y= " & x ^ y & vbCrLf
End Sub

Private Sub Button8_Click(sender As Object, e As EventArgs) Handles
Button8.Click
    End
End Sub
End Class

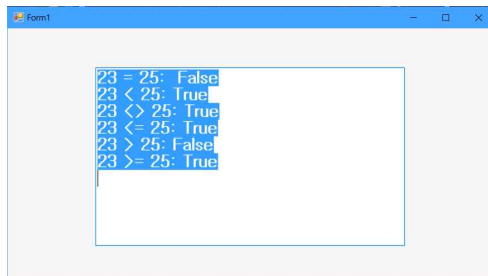
```

2.비교연산자: =, <, > 등 관계를 나타내기 위한 연산자. True or False의 결과만 나옴.

-비교연산자의 종류

연산자	일반형식	의미	사용 예
<	$x < y$	작다.(x가 작으면 True)	$9 < 15$ ----True
≤	$x \leq y$	작거나 같다(x가 작거나 같으면 True).	$21 \leq 21$ ---True
>	$x > y$	크다.(x가 크면 True)	$31 > 33$ --False
≥	$x \geq y$	크거나 같다(x가 크거나 같으면 True).	$11 \geq 10$ ---True
=	$x = y$	같다.(x와 y가 같으면 True)	$21 = 23$ ----False
<>	$x < > y$	같지 않다.(x와 y가 같지 않으면 True)	$24 < > 24$ ---False
Is	$x \text{ Is } y$	두 개체가 가리키는 값이 같은지 알아낸다. (x, y가 가리키는 변수가 같으면 True, 다르면 False)	set Form_ob = Form1 set Label_ob = Label1 Form_ob Is Label_ob -----False-----
Like	$x \text{ Like } y$	두 문자열을 비교.(x 와 y를 비교, 같으면 True, 다르면 False)	"basic" Like "basi" -----False

〈비교연산자 프로그램〉



Public Class Form1

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    TextBox1.Text = "23 = 25: " & CStr(23 = 25) & vbCrLf &
        "23 < 25: " & CStr(23 < 25) & vbCrLf &
        "23 <> 25: " & CStr(23 <> 25) & vbCrLf &
        "23 <= 25: " & CStr(23 <= 25) & vbCrLf &
        "23 > 25: " & CStr(23 > 25) & vbCrLf &
        "23 >= 25: " & CStr(23 >= 25) & vbCrLf
End Sub
```

End Sub

End Class

*주의점!!!!!!

1. 비교연산자 쓸 때 Cstr(숫자를 문자로 바꿔주는 함수)를 사용해야함
2. <=, >=를 쓸때는 부등호 먼저, 그다음 =를 입력한다.

3.논리연산자: 연산을 하려는 두 대상의 두 값을 and, equal 등을 이용하여 True, False로

논리값을 구할 수 있는 연산자.

>> 비교대상이 연산자를 이용하여 논리적으로 비교할 수 있어야 한다.

-논리 연산자의 종류

연산자	일반 형식	의미
And	x And y	x, y 둘다 True 일 때만 True
Or	x Or y	x, y 둘다 False일 때만 False
Xor	x Xor y	x와 y가 다르면 True, 같으면 False
Not	Not x	x가 True이면 False, False이면 True
Andalso	x Andalso y	x와 y가 같으면 True, 다르면 False
OrElse	x OrElse y	x가 True고 y가 False면 False, 외는 True

4.대입연산자: '=' 오른쪽에 있는 값을 왼쪽에 대입시켜주는 기능. y값을 x에 저장하기도 함.

ex) x = 10 --- 올바른 표현

x = y + 10 --- 올바른 표현

10 = x ---잘못된 표현

x + y = 15 ---잘못된 표현

5. 연결연산자

문자열들을 결합하여 하나의 문자열을 반환하는 연산자.

-연결연산자의 종류

연산자	일반형식	의미	사용 예
&	"x"&"y"	두 식을 연결(결과는 "xy")	"Cook"&"Book" >>CookBook
+	"문자열x"+"문자열y"	두 문자열 연결(결과는 "문자열x문자열y")	"Cook"+"Book" >>CookBook

-연결연산자 사용 규칙

연산자	일반형식	의미
x+y	x와 y모두 수식	덧셈(산술연산)
	x와 y모두 문자열	문자열 연결(연결연산)
	x는 수식,y는 문자열 또는 그 반대	(오류 발생)

*(+)사용시 문제 발생 가능성

두 숫자를 5 + 5처럼 연결연산자 사용시 5를 문자로 인식하여 55라는 값이 나올 수 있음

>> 해결방법: 두 숫자를 변수로 두거나 함수를 사용한다.

*연산자 우선순위(1)2)3)4)

1. 산술연산자

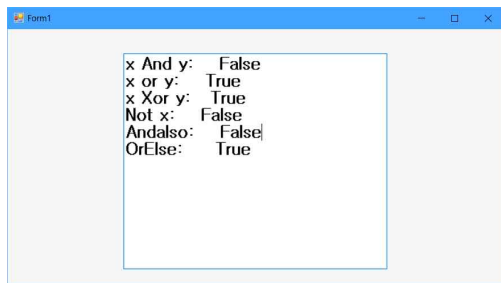
^(거듭제곱) > *, /(곱셈, 나눗셈) > %(몫의 정수) > Mod(나머지) > +, - (덧셈, 뺄셈)

2. 연결 연산자: & (문자열 연결)

3. 관계연산자: <, ≤, =, >, ≥

4. 논리연산자: Not > And > Or > Xor > Not > Andalso > OrElse

<논리연산자 프로그램>



텍스트박스 하나 만들기

텍박 밖에 Form 더블클릭 후

```
Public Class Form1
```

```
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
        Dim x, y As Boolean( True or False 구분시 사용)
```

```
        x = True
```

```
        y = False
```

```
        TextBox1.Text = "x And y:  " & CStr(x And y) & vbCrLf &
```

```
                        "x or y:  " & CStr(x Or y) & vbCrLf &
```

```
                        "x Xor y:  " & CStr(x Xor y) & vbCrLf &
```

```
                        "Not x:  " & CStr(Not x) & vbCrLf &
```

```
                        "Andalso:  " & CStr(x AndAlso y) & vbCrLf &
```

```
                        "OrElse:  " & CStr(x OrElse y) & vbCrLf
```

```
    End Sub
```

```
End Class
```

Cstr - 숫자를 문자로 바꾸는 함수

5주차) 제어문

컴퓨터 프로그램은 항상 “순차적 흐름”을 가진다. (1)2)3)4....순으로, 3번 반복하고 싶어도x) 하지만 제어문을 사용시 위의 순차적 흐름을 제어할 수 있다.

-제어문의 종류

- 1)조건문: 조건을 검사하여 그 결과에 따라 선택적인 작업을 수행
- 2)반복문: 조건에 따라 원하는 명령문을 반복 수행

구분	제어문	설명
조건문	If 문	조건을 검사하여 그 결과에 따라 선택적인 작업 수행
	Select~Case 문	
반복문	For 문	조건에 따라 특정 명령문을 반복 수행한다.
	Do 문	
	While 문	
기타 제어문	Exit 문	Exit문을 포함한 블록을 빠져나간다
	Goto 문	제어를 특정 위치로 옮긴다.
	With 문	반복되는 부분을 생략한다.

〈조건문:If문〉

*If문의 유형

유형1) 참만 있는 경우

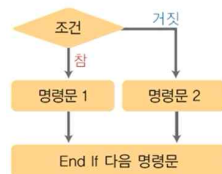
형식: If 조건 Then (명령문(들))
End If

조건은 사용자가 지정 가능, 조건이 **참**인 경우 Then 명령문(들) 수행.

거짓인 경우 End If 다음 명령문(들) 수행.

유형2) 참/거짓이 공존

형식: If 조건 Then
명령문 1
Else
명령문 2
End If



>> “+”를 쓰면 안됨. 반드시 ‘비교연산자(<,>,...)’를 사용해야함.

조건이 참이면 Then~Else 사이의 명령문1을,

거짓이면 Else~End If 사이의 명령문 2를 수행한다.

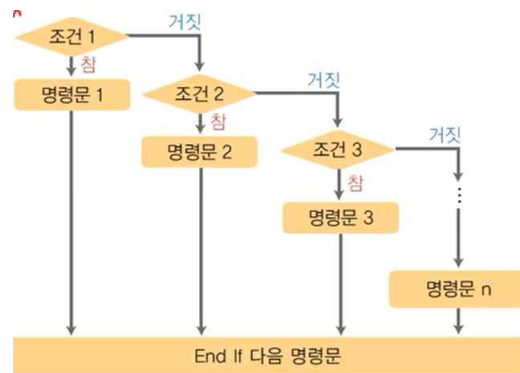
명령문 1/2 이후에는 End If의 다음 명령문을 수행한다.

유형3) 참/아닐 경우~ 분류가 반복 (다중If)


```

If 조건1 Then
    명령문 1
Elseif 조건2 Then
    명령문 2
Elseif 조건3 Then
    명령문 3
    :
Else
    명령문n
End If

```



참인 조건을 만나면 해당 명령문을 수행하고 이후 조건은 모두 무시한 채 End If 다음으로 제어를 옮겨 If문을 종료한다.

***Elseif (만약 참이 아닐 경우) 사용 시 'Else'와 'If'를 띄워 쓰면 안됨!!**

(If문 이용해서 학점계산 프로그램 만들기)

-유형1(참만 있는 경우: 70점 이상이면 통과)

(버튼1)

```
Dim 중간고사 As Integer
```

'선언부변수선언(정수형태의값을저장)

```
중간고사= TextBox1.Text '입력부
```

```
If 중간고사>= 70 Then
```

```
TextBox2.Text = "합격"
```

```
End If
```

>>다음과 같이 70점이 넘으면 “합격”이라는 문구 나옴

***주의점!**

중간고사=TextBox1.Text와 같이 변수를 만들 때와 TextBox2.Text="합격"은 다르다. "를 쓰는 경우 변수를 지정하는 것이 아니라 문자를 표시한 것이다

***명령어 입력 후 동일 행 뒤에 (') 하나 입력 후 주석을 달 수 있다.**

-유형2(Else;거짓 이 있는 경우)

(버튼1)

```
Dim 중간고사As Integer '선언부변수선언(정수형)
```

```
Dim 합격여부As String '선언부변수선언(문자형)
```

```
중간고사= TextBox1.Text '입력부
```

```
If 중간고사>= 70 Then '제어문
```

```
합격여부= "합격입니다" or "합격"
```

```
Else
```

```
합격여부= "불합격입니다." or "합격"
```

```
End If
```

```
TextBox2.Text = 합격여부 or &"입니다." (저장된 결과 출력)
```

-유형3) 참/아닐 | 경우~ 분류가 반복

(다중If)

```
Dim 점수As Integer      '변수(정수형)
Dim 학점As String       '변수(문자형)
Dim 이름As String       '변수(문자형)
점수= TextBox1.Text
If 점수>= 95 Then
    학점= "A+"
Elseif 점수>= 90 Then
    학점= "A"
Elseif 점수>= 85 Then
    학점= "B+"
Elseif 점수>= 80 Then
    학점= "B"
Elseif 점수>= 75 Then
    학점= "C+"
Elseif 점수>= 70 Then
    학점= "C"
Elseif 점수>= 65 Then
    학점= "D+"
Elseif 점수>= 60 Then
    학점= "D"
Elseif 점수>= 55 Then
    학점= "F"
End If
TextBox2.Text = 학점
```

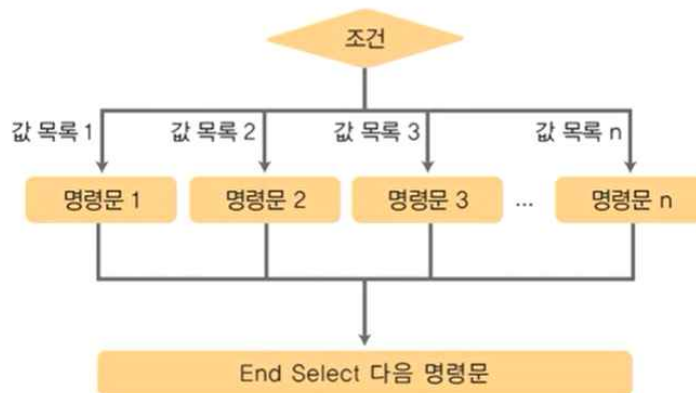
-유형4) 스스로 학습하기

조건) 중간고사/기말고사/출석점수 합 평균
이를 If문을 이용하여 학점 구하기
95이상 A+/90이상 A/..../59점이하 F
"최준용님의 평균은 n점입니다. 학점-"
위의 형태로 표시.

```
Dim 중간고사As Integer  '변수(정수형)
Dim 기말고사As Integer  '변수(정수형)
Dim 출석As Integer      '변수(정수형)
Dim 학점As String       '변수(문자형)
Dim 이름As String       '변수(문자형)
Dim 점수As Integer      '변수(정수형)
이름= TextBox3.Text
중간고사= TextBox1.Text
기말고사= TextBox4.Text
출석= TextBox5.Text
점수= (중간고사+ 기말고사+ 출석) / 3
If 점수>= 95 Then
    학점= "A+"
Elseif 점수>= 90 Then
    학점= "A"
Elseif 점수>= 85 Then
    학점= "B+"
Elseif 점수>= 80 Then
    학점= "B"
Elseif 점수>= 75 Then
    학점= "C+"
Elseif 점수>= 70 Then
    학점= "C"
Elseif 점수>= 65 Then
    학점= "D+"
Elseif 점수>= 60 Then
    학점= "D"
Elseif 점수<= 59 Then
    학점= "F"
End If
TextBox2.Text = 이름 & "님의평균은" & 점수 & "입니다." & "학점은" & 학점 & "입니다."
End Sub
```

〈조건문: Select Case문〉

형식) Select Case 조건
 Case Is 값 목록1
 명령문1
 Case Is 값 목록2
 명령문2
 Case Is 값 목록3
 명령문3
 :
 Case Else
 명령문n
 End Select



-If문의 경우 선택사항이 많아질 경우 구조가 복잡해짐.

이에 따라 Select Case문을 사용하여 좀 더 단순한 구조로 작성 가능.

-조건을 주고 그 조건이 가지는 값을 각 Case 목록에 나타낸다.

-변수에 따라 해당하는 Case에 연결된 명령문을 수행하고 Select Case문을 빠져나간다.

***Case만 쓸 경우 해당하는 값만, Case Is를 쓸 경우 “비교연산자(>,<=,...)”사용하여야함.**

Select Case문 프로그램)

```
Dim select문As Integer
select문= TextBox1.Text
Select Case select문
    Case 100
        TextBox2.Text = "A+"
    Case 95 To 99
        TextBox2.Text = "A+"
    Case 90, 91, 92, 93, 94
        TextBox2.Text = "A"
    Case Is >= 85
        TextBox2.Text = "B+"
    Case Is >= 80
        TextBox2.Text = "B"
    Case 75 To 79
        TextBox2.Text = "C+"
    Case Is >= 70
        TextBox2.Text = "C"
    Case Is >= 65
        TextBox2.Text = "D+"
    Case Is >= 60
        TextBox2.Text = "D"

    Case Is <= 59
        TextBox2.Text = "F"

End Select
```

〈스스로학습하기:Select문〉

Dim 학점As String

Dim 중간고사As Integer

Dim 기말고사As Integer

Dim 출석As Integer

Dim 이름As String

Dim 점수As Integer

중간고사= TextBox1.Text

기말고사= TextBox3.Text

출석= TextBox4.Text

이름= TextBox5.Text

점수= (중간고사+ 기말고사+ 출석) / 3

Select Case 점수

Case 100

학점= "A+"

Case 95 To 99

학점= "A+"

Case 90, 91, 92, 93, 94

학점= "A"

Case Is >= 85

학점= "B+"

Case Is >= 80 >>> 비교연산자 사용시 Case Is 사용

학점= "B"

Case 75 To 79 >>> 지정 변수, 범위 사용시 Case만 사용

학점= "C+"

Case Is >= 70

학점= "C"

Case Is >= 65

학점= "D+"

Case Is >= 60

학점= "D"

Case Is <= 59

학점= "F"

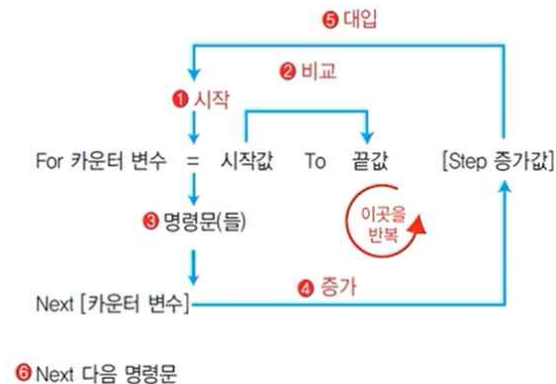
End Select

TextBox2.Text = 이름& " 님의점수는" & 점수& "점이며, 학점은" & 학점& " 입니다."

〈반복문: For문〉



그림 5-7 For 문의 동작 원리



형식) For 카운터변수 = 시작값 To 끝값 [Step 증가값]
명령문(들)
Next [카운터 변수]

-반복문은 주어진 조건에 따라 명령문을 반복해서 실행시킬 때 사용한다.

-반복횟수를 지정하고 싶은 경우 "For문", 특정 조건을 만족할 경우에만 반복시키려면

"Do문-조건을 만족할 때 까지", "While문"을 쓴다.

*Step: 예를 들어 1,3,5,7,9처럼 변화시키는 숫자의 간격을 지정할 수 있음.

Step이 없는 경우 무조건 변화시키는 숫자의 간격은 '1'

*Next: For 단계로 회귀

>>For >> (Step) >> Next를 '참'이면 다시 For로 회귀, '거짓'이 나올 때 까지 반복한다.

〈For문 프로그램〉

Dim i As Integer >>숫자가 커지면 Long사용

Dim Sum As Integer

i = 0 >> 초기값 (안줘도됨. 어차피 밑에서줌)

Sum = 0 >> 초기값 (보통 '0'으로 줌)

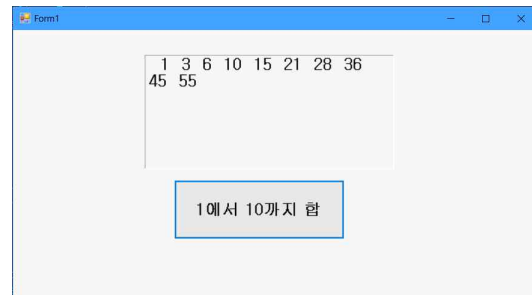
For i = 1 To 10 Step 1

Sum = Sum + i

Label1.Text = Label1.Text & " " & Sum >> 위처럼 계산과정을 나타내기위함.

Next

만약 (Label1.Text = Sum)이면 결과(55)만출력.



〈For문 스스로 학습하기〉

1) 1에서 100까지 합 구하기

For i = 1 To 10 Step 1 >>> For i = 1 To 100 Step 1

2) 1에서 100까지 짝수 합 구하기

For i = 1 To 100 Step 2

로 바꿔주면 됨

7주차)대화상자/반복문

1.대화상자

-컨트롤을 사용하지않고 입/출력을 원할 때 사용

-입력하는 상자: Input Box("설명", "제목", "입력창메세지")

("두 수의 곱을 구하기", "첫번째 숫자입력", "여기에 입력하십시오")

-출력하는 상자: Msg Box(값)

〈대화상자를 이용하여 두 수의 곱 구하는 프로그램〉

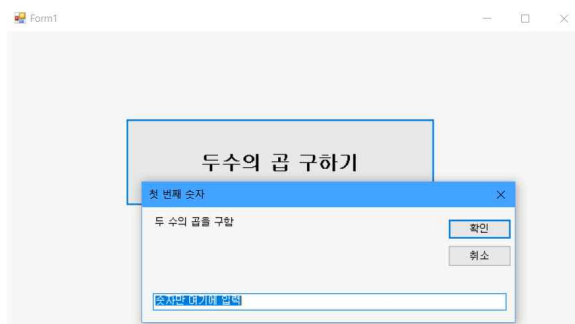
```
Dim x As Integer
```

```
Dim y As Integer
```

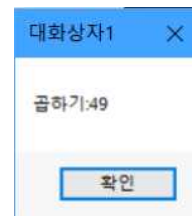
```
x = InputBox("두수의곱을구함", "첫번째숫자", "숫자만여기에입력")
```

```
y = InputBox("두수의곱", "두번째숫자", "숫자만입력")
```

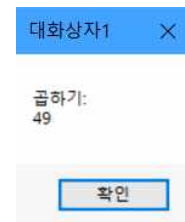
```
MsgBox("곱하기:" & vbCrLf & x * y)
```



- vbCrLf입력x



-vbCrLf입력o



>>
값의 행
띄워짐

>>다음과 같이 TextBox나 Label등의 "control" 없이 두 수의 곱을 msg box로 입/출력 가능

〈반복문(For문)을 이용, 정수 1부터 20중에서 5의 배수의 합을 대화상자로 표시하기〉

```
Dim x As Integer
```

```
Dim sum As Integer
```

```
sum = 0
```

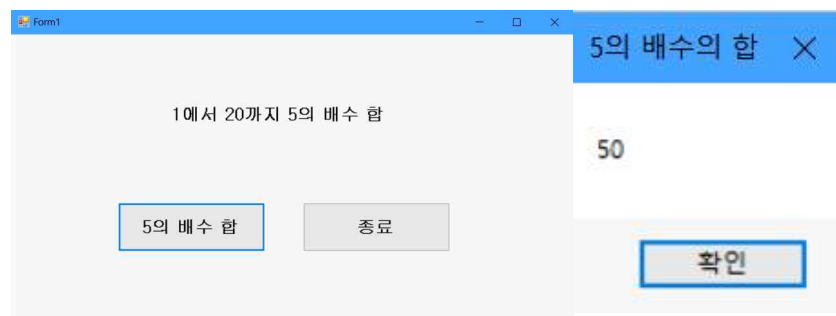
```
For x = 0 To 20
```

```
Step 5
```

```
sum = sum + x
```

```
Next
```

```
MsgBox(sum)
```



〈위 프로그램을 이용하여 1에서 20 중에서 n의 배수 합을 대화상자로 표시〉

```
Dim n As Integer
```

```
Dim x As Integer
```

```
Dim sum As Integer
```

```
n = TextBox1.Text
```

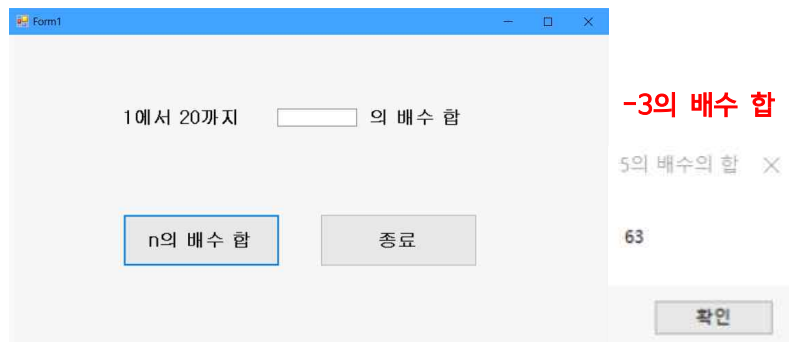
```
sum = 0
```

```
For x = 0 To 20 Step n
```

```
sum = sum + x
```

```
Next
```

```
MsgBox(sum)
```



-3의 배수 합

5의 배수의 합

63

확인

〈반복문(For문)을 이용, 입력받는 범위내 숫자의 합 구하기〉

Dim 시작값, 끝값, 증가값, 합계, i As Integer

Dim 곱하기As Long>>값이대>>Long/Double

시작값= TextBox1.Text

끝값= TextBox2.Text

증가값= TextBox3.Text

곱하기= 1 '곱하기는초기값필수

For i = 시작값To 끝값Step 증가값

합계= 합계+ i

곱하기= 곱하기* i

Next

TextBox4.Text = 합계'합계=sum >>For를 이용한 반복연산 후 출력하는 값이므로

TextBox5.Text = 곱하기 For 문이 끝나는 Next 아래에 합계/곱하기 둬

〈중첩 반복문(2중 For문)을 이용, 구구단 출력하기〉

*2중 For문: 반복문인 For문 내에 반복되는 또 다른 For문을 입력가능

Dim x, y, z As Integer

x = 1

y = 1

For x = 2 To 9

For y = 1 To 9

TextBox1.Text = TextBox1.Text & x &

"*" & y & "=" & x * y & vbCrLf

Next

TextBox1.Text = TextBox1.Text & vbCrLf >> 1중 For문이 끝날 때 마다 띄우기 위함

Next

〈중첩 반복문을 이용하여 원하는 단만 출력하는 프로그램 만들기〉시발존나어렵네

(버튼'n의 구구단')

Dim a, b, c As Integer

a = TextBox2.Text

b = 1

c = TextBox3.Text

For a = TextBox2.Text To c

For b = 1 To 9

TextBox1.Text = TextBox1.Text & vbCrLf >> 1중 For문이 끝날 때 마다 띄우기 위함

Next

〈중첩 반복문을 이용하여 원하는 단만 출력하는 프로그램 만들기〉시발존나어렵네

(버튼'n의 구구단')

Dim a, b, c As Integer

a = TextBox2.Text

b = 1

c = TextBox3.Text

For a = TextBox2.Text To c

For b = 1 To 9

TextBox1.Text = TextBox1.Text & a & "*" & b & "=" & a * b & vbCrLf

Next

TextBox1.Text = TextBox1.Text & vbCrLf

Next

(버튼'삭제')

TextBox1.Text = " ", TextBox2.Text = " ", TextBox3.Text = " ",

TextBox2.Focus() >> *Focus는 동작 진행 후 커서위치를 정하는 명령어

9주차) 다양한 컨트롤/메모장 만들기/ 반복(Loop)문

-다양한 컨트롤

- 1) RadioButton: 그룹 내에서 하나의 항목만 선택하고자 할 때 사용
- 2) Checkbox: 그룹 내에서 여러 개의 항목을 선택하고자 할 때 사용
- 3) HScrollBar(수평), VScrollBar(수직): 마우스의 휠을 이용하여 드래그로 값을 결정함

>> 그룹박스 내에 컨트롤들을 넣을 때 그룹박스 먼저 두고 그 안에 컨트롤을 넣어야함.

-코드를 수정하여 글자 속성을 지정하는법

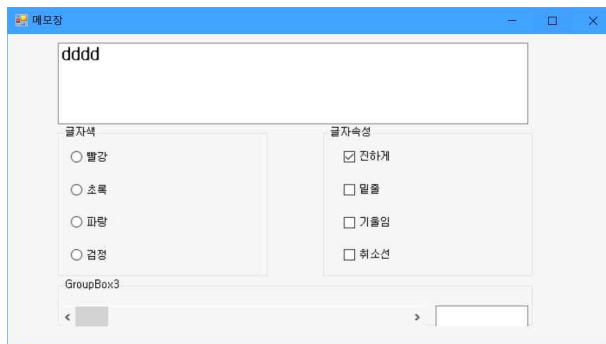
1) ForeColor(글자색):

- 컨트롤명.ForeColor=Color.지정색
- TextBox1.ForeColor=Color.RED(지정색)

2) Font(글꼴, 글자 속성):

- 컨트롤명.Font=New Font(컨트롤명.Font.Name, 글자크기, 글자스타일)
- TextBox1.Font=New Font(TextBox1.Font.Name, 15, FontStyle.Bold)

*메모장 프로그램



(1)도구상자의 '컨테이너'에서 Group Box 우선 설치

(2)Radio Button을 '글자색' 그룹에 설치

(3)checkbox을 '글자속성' 그룹에 설치

(4)Hscrollbar를 '글자크기' (그룹3)에 설치

>>Hscrollbar는 모든 Windows form에

-글자색 바꾸기 Radio button (ForeColor)

```
Private Sub RadioButton2_CheckedChanged(sender As Object, e As EventArgs) Handles  
RadioButton2.CheckedChanged
```

```
    TextBox1.ForeColor = Color.Green
```

```
End Sub
```

-글자속성 바꾸기 Check Box (FontStyle)

```
Private Sub CheckBox1_CheckedChanged(sender As Object, e As EventArgs) Handles  
CheckBox1.CheckedChanged
```

```
    If CheckBox1.Checked = True Then(CheckBox 선택시)
```

```
        TextBox1.Font = New Font(TextBox1.Font.Name, 15, FontStyle.Bold)>>진한글씨
```

```
    Else(CheckBox 해제시)
```

```
        TextBox1.Font = New Font(TextBox1.Font.Name, 10, FontStyle.Bold)>>원래대로
```

```
    End If
```

```
End Sub
```

-글자크기 바꾸기 HScrollBar >> 디자인 속성에서 'Maximum/Minimum' 값 설정(1~n)

```
Private Sub HScrollBar1_Scroll(sender As Object, e As ScrollEventArgs) Handles  
HScrollBar1.Scroll
```

```
    TextBox1.Font =New Font(TextBox1.Font.Name, HScrollBar1.Value)>>TextBox1글자크기
```

```
    TextBox2.Text = HScrollBar1.Value >> 스크롤바의 값을 텍스트박스에 표시
```


*반복문의 Count: For/Do 문의 반복시 반복되는 횟수의 Count가 필요할 수 있음

1. Do While~Loop문

-For~Next 반복문은 몇 번 반복하는지 아는 경우에 유리한 반복문인 것에 비해

Do While ~ 반복문은 **몇 번 반복하는지 알지 못하는 경우에 유리**한 반복문이다.

Do문이 참일 경우: Do While / 거짓일 경우: Do Until

ex)

Do While i<=10 sum = sum+1 i=i+1 Loop
--

 >> 10까지는 1회 루프당 1씩 더해지나 11이 된 경우 다음 명령수행.

-Do While ~문 이용 프로그램(label에 1~5까지 출력)

1부터 5까지

1 2 3 4 5

출력

```
Button1.Click
Dim str As String
Dim i As Integer
i = 1
str = " " >>연산된 값 표시
Do While i <= 5 >>While에서 참일때만 아래 명령 수행
    str = str & i & " "
    i = i + 1
Loop >> i=1에서부터 5가 될 때 까지 Do로 돌려보냄
Label1.Text = str
End Sub
```

2. Do~ Loop While문

-Do While ~Loop문은 반복구조 처음부터 조건 비교 후 명령 수행.

Do~ Loop While문은 명령 모두 수행 후 조건 비교.

>> 조건 미충족시 전자의 경우 Do문이 아예 실행되지x. 후자는 명령 실행 후 조건 비교.

-Do~Loop while 문 이용 프로그램(label에 1~10까지 짝수의 합)

1부터 10까지 짝수 합

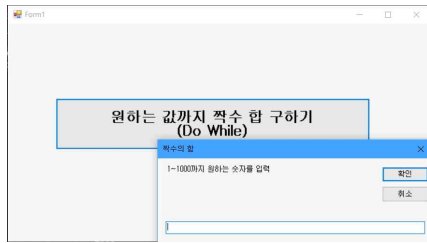
30

출력

```
Dim i As Integer
Dim sum As Integer
sum = 0
i = 0
Do
    i = i + 1
    If ((i Mod 2) = 0) Then >>연산된 i를 2로나누면 0이다.
                                                (즉, i=짝수)
        sum = sum + i
Loop While i <= 10
Label1.Text = sum
End If
```

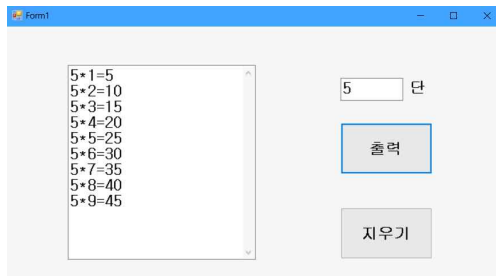
*여러 Do문을 이용한 프로그램

1)임의의 값까지의 짝수 합하기



```
Dim i As Integer
Dim sum As Integer
Dim x As Integer
i = 0
sum = 0
x = InputBox("1~1000까지원하는숫자를입력", "짝수의합")
Do While i <= x
    If (i Mod 2 = 0) Then
        sum = sum + i
    End If
Loop
MsgBox("sum=" & sum)
```

2)원하는 단의 구구단 출력하기(p.23하단의 For 문을 이용한 구구단과 유사)



```
Dim 단 As Integer
Dim i As Integer
i = 1
단 = Val(텍스트)숫자함수 (TextBox2.Text)
i = 1
Do While i <= 9
    TextBox1.Text = TextBox1.Text & 단 & "*" & i & "=" & 단 * i & vbCrLf
    i = i + 1>>매 루프당 i값 증가량 꼭써라
Loop
```

[매크로와 "VisualBasic"]

-이름학교학과 입력 프로그램

B

3 최준용 동아대 신소재공학 >>매크로 기록>매크로이름:이름학교학과>
바로가기 키: Ctrl+i (컨+C/V/Z같이 자주쓰는거x)

Sub 이름학교학과()

,

' 이름학교학과 Macro

,

' 바로 가기 키: Ctrl+i

,

ActiveCell.FormulaR1C1 = "최준용"

Range("C3").Select

ActiveCell.FormulaR1C1 = "동아대"

Range("D3").Select

ActiveCell.FormulaR1C1 = "신소재공학"

Range("D4").Select

End Sub

최준용	동아대	신소재공학
단추 4		

>>컨트롤 삽입(리본메뉴)>>양식컨트롤의 버튼 삽입 후
입력했던 매크로 작용시 B3을 선택 후 버튼 눌렀을 때
설정한 매크로 작동

***저장시 파일 형식은 반드시 "Excel 매크로 사용 통합 문서"로 저장해야함!!!!!!!!!!!!**

*셀 영역

Visual Studio와 VBA의 가장 큰 차이점은 '셀'의 존재 유무이다. 또한

-Range(범위)를 둘 때 Visual Studio에서는 Range.Select등의 형식으로 명령어 입력.

VBA에서는 Range(x~y).Select의 형식으로 특정 셀/범위를 선택 해야 한다.

-수식(Fomula/Value) 입력시 Range(A5) 선택 후 Fomula/Value등을 설정할 수 있다.

-Value>> Range(A5).Value=100 (A5셀에 100이라는 특정 값 입력됨)

- Fomula>> Range(A5).Fomula=100 (Fomula= "=Sum(A1:A5)"의 수식이 입력됨)

*Range:셀에서 이뤄지는 모든 작업을 제어할 때 사용

-Range("A1").select : A1셀을 선택

-Range("A1:A10").select : A1셀부터 A10셀까지 선택(범위)

-Range("A1","A10").select : A1셀부터 A10셀까지 선택(범위)

-Range("A1,A10").select : A1셀과 A10셀을 선택

-Range("셀 주소").value=100 : 특정 셀 주소에 100이라는 '값'대입

*Cells:행/열번호로 특정 셀을 선택

-Cells(행 번호, 열 번호).select/Range/Fomula

*직접 실행

VBA진입 후 보기>>직접실행 창 선택후 질문시 대답 들을 수 있음.

Ex) ?application.Name>>프로그램 이름이 무엇이니?

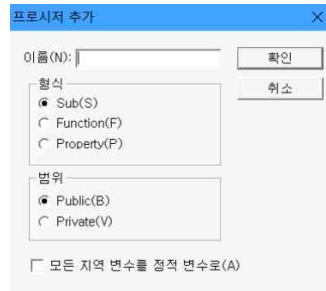
Microsoft Excel>>마소엑셀

?worksheets.Count>>시트 몇 개 있니?

3 >>3개

***한 프로젝트는 여러 Module의 집합이며, 한 Module은 여러 프로시저의 집합이다!**

***프로시저: 모듈 진입 후 보기>>프로시저 추가 선택 시**



>>다음과 같은 창 생

-Sub: 매크로/VBA 언어 입력시 최소의 프로시저 단위

-Function: 함수 입력시 필수

-Public: 전역범위; 여러 프로시저 내에서 적용가능

-Private: 지역범위; 한 프로시저 내에서만 적용가능

***VBA Module내 입력으로 매크로 작성하기**

-A1~A20선택 후 B1~B10에 100의 값 입력

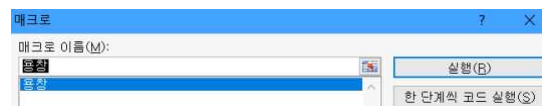
Module1에

Sub 용창()

Range("a1").Select

Range("a1:a10").Select >>f5,Alt11>

Range("b1:b20").Value = 100



End Sub

>시트로 되돌아와 "매크로 보기"클릭시 보임

-B1~B20에 입력했던 값 지우기

Sub 용짱짱()

Range("b1:b20").Delete 입력 후 F5, Alt11 시 지워져있음

-앞에 입력한 두 프로시저 도형으로 실행 버튼 만들기

도형 입력>>마우스 우클릭>>매크로 지정

*매크로 기록

- 1) 절대 참조 기록: (셀)주소가 절대 바뀌지 않고 기록
- 2) 상대 참조 기록: (셀)주소가 상대적으로 원하는 주소로 바뀌며 기록

-상대참조 매크로 기록 하는법:

보기>>매크로 버튼>>상대참조 선택 >> 매크로 기록

-절대참조 vs 상대참조 프로시저

Sub 절대참조()

,

' 절대참조 Macro

,

' 바로 가기 키: Ctrl+j

,

ActiveCell.FormulaR1C1 = "절대참조"

Range("C5").Select

End Sub

Sub 상대참조()

,

' 상대참조 Macro

,

' 바로 가기 키: Ctrl+k

,

ActiveCell.FormulaR1C1 = "상대참조"

ActiveCell.Offset(1,0).Range("A1").Select

End Sub

↳ **ActiveCell**:마우스포인터

↳ **Offset(행 번호, 열 번호)**:상대적인 값

ex) **ActiveCell.Offset(0,1)**: 커서가 선택한 셀에서 1열 아래의 셀에 명령어 수행

[사용자 정의 함수]

*지우기

- ClearContents: 내용만 지우기
- ClearFormats: 서식만 지우기
- Clear: 내용과 서식 모두 지우기

*설정 범위의 내용만 지우는 프로그램(ClearContents사용)

Dim I As Integer

I= MsgBox("범위 설정한 영역의 내용만 삭제하시겠습니까?", vbYesNo, "삭제 확인")

>>매크로로 삭제한 내용은 복구가 되지 않음. 따라서 한번 더 확인.

If I = 7 Then Exit Sub >> Yes=6/No=7 따라서 7일 경우 알아서 If문 나감>>End if 필요x

Selection.ClearContents >>선택된 범위의 내용만 삭제

MsgBox ("범위 설정된 영역의 내용만 삭제되었습니다.", "삭제 확인")

*설정 범위 내의 내용/서식/내용, 서식 모두지우기 프로그램

Sub 내용만지우기()

Selection.ClearContents

End Sub

Sub 서식만지우기()

Selection.ClearFormats

End Sub

Sub 내용서식모두지우기()

Dim i As Integer

i = MsgBox("범위 설정한 영역의 내용만 삭제하시겠습니까?", vbYesNo, "삭제확인")

If i = 7 Then Exit Sub

Selection.Clear

MsgBox ("삭제완료")

End Sub

이름	직위	부서	점수	내용만 지우기
박진희	사원	영업	100	
김진수	대리	영업	88	서식만 지우기
한명희	대리	인사	78	
이원형	과장	경리	75	내용서식모두지우기
안영찬	과장	인사	46	
이애경	대리	마케팅	80	
정길수	대리	마케팅	83	
김미애	사원	인사	75	
최영란	사원	경리	85	
김미수	사원	영업	93	

*셀 범위를 입력박스로 받기

-Application.InputBox("셀범위", "입력값", Type:=8)

-Application.Inputbox(제목, 기본값, Type:=숫자)

-InputBox의 종류

1)InputBox: 숫자/문자만 저장 가능

2)Application InputBox: 다양한 입력값(셀 범위, Type:=숫자) 저장 가능

-Type의 종류

0: 수식/1: 숫자/2: 텍스트/4: 논리값/

★8: Range와 같은 '셀 참조'(셀 주소 저장가능)/

16: #N/A의 오류값/64: 값의 배열

***입력한 글자(=Type:=2)를 찾아 배경색 바꾸기(ApplicationBox를 이용)**

-range("a1").CurrentRegion.Find(변수)

>>range("a1"):a1셀을 기준으로/CurrentRegion:특정 범위 내의/Find(변수): 변수(텍스트)찾기

-변수.Interior.ColorIndex=4

>>찾은 변수(텍스트)의/Interior: 내부/ColorIndex: 배경색/=4: 4로 설정된 색으로 바꾸기

Ex)배경색 바꾸기 프로그램

Sub 배경색바꾸기()

Dim r as range >> range(셀의 주소)를 변수로 받을 수 있음.

Dim s as string

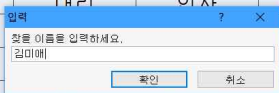
s=Application.inputBox("찾을 이름을 입력하세요“, type:=2(텍스트))

set r=range("a1").CurrentRegion.Find(s) >> 범위 설정 및 범위 내 특정 변수 찾기

r.interior.ColorIndex=4 >>찾은 변수의 배경색 바꾸기

>>

이름	직위	부서	점수
박진희	사원	영업	100
김진수	대리	영업	88
한영희	대리	인사	78
이원형			75
안영찬			46
이애경	대리	마케팅	80
정길수	대리	마케팅	83
김미애	사원	인사	75
최영란	사원	경리	85
김민수	사원	영업	93



이름	직위	부서	점수
박진희	사원	영업	100
김진수	대리	영업	88
한영희	대리	인사	78
이원형	과장	경리	75
안영찬	과장	인사	46
이애경	대리	마케팅	80
정길수	대리	마케팅	83
김미애	사원	인사	75
최영란	사원	경리	85
김민수	사원	영업	93

>>범위 드래그 후 매크로 실행>>김미애 입력>>색 바뀜

***입력한 범위에 데이터 입력하기 프로그램**

Sub rangeType()

Dim returnsel As Range

Set Returnsel = Application.InputBox

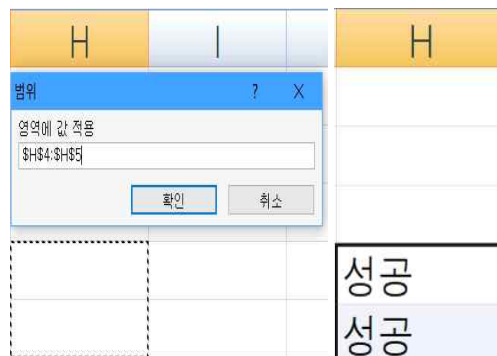
("영역에 값 적용“, "범위", Type:=8(셀 주소 저장))

returnsel.Select

SelectionValue = "성공"

End Sub

>>셀 주소 지정시 설정된 데이터가 입력.



***입력한 범위에 데이터 지우기 프로그램**

Sub 선택영역지우기()

Dim 지우기 As Range

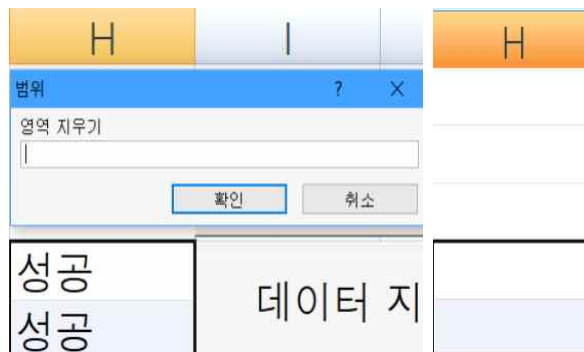
Set 지우기 = Application.InputBox

("영역 지우기", "범위", Type:=8)

지우기.Select

Selection.clear

End Sub



***선택한 셀의 함수 구하기(WorksheetFunction 이용)**

-WorksheetFunction: 매크로에서 함수를 이용할 수 있도록 해줌

***원하는 셀을 입력받아 함수 구하기(WorksheetFunction사용)**

1) 일정 범위 내 값 더하기(Sum)

Dim 입력1 As range

Dim x As integer

Set 입력1 = Application.InputBox ("셀 범위", "입력값", Type:=8)>>Range 쓰기위함

x=Application.WorksheetFunction.Sum(입력1)>>입력1 범위를 받아 x의 연산 진행

MsgBox(x)

2) 일정 범위 내 셀 개수 구하기

Dim 입력1 As Range

Dim x As integer

Set 입력1 = Application.Inputbox ("셀 범위", "입력값", Type:=8)

x= Application.WorksheetFunction.CountA(입력1)>>Counta:문자/숫자가 입력된 셀 개수

MsgBox(x)

3) 일정 범위 내 숫자 평균 구하기

Dim 입력1 As Range

Dim x As Single(실수)

Set 입력1 = Application.Inputbox("셀범위", "입력값", Type:=8)

x= Application.WorksheetFunction.Average(입력1)

MsgBox(x)

***사용자 지정 함수 만들기:Function 프로시저를 작성하여 사용자가 원하는 함수 만들기**

〈형식〉

Sub--x 함수 만들때는 Sum이 아닌 Function 사용.

>>

Function 함수명 (인수1,인수2,인수3; 없을 때는 비움, '인수'= 수/셀/조건 등을 의미)

함수에서 처리할 명령1

함수에서 처리할 명령2

:

★함수명=결과값 >>“반환”:결과값을 함수명에 입력해야 함.

***두 수를 입력받아 큰 값을 구하는 함수**

Public Function MaxVal(a(변수) as range, b(변수) as range)

Dim c as single(실수)

if a.value>b.value

c=a.value

else

c=b.value

End If

Max Val=c

12주차)사용자 지정 함수/Worksheet, Worksheets 개체

-Function 프로시저

〈형식〉

Public Function 함수명 (인수1, 인수2,.../인수가 없는'today'함수같은 경우 공란으로 둔다)

함수에서 처리할 명령1

“ 2

★함수명=결과값>>>“반환”:결과값을 함수명에 입력해야 함.

End Function

*두 수를 입력받아 큰 값을 구하는 함수

Public Function MaxVal(a(변수) as range, b(변수) as range)

Dim c as single(실수)

if a.value>b.value

c=a.value

else

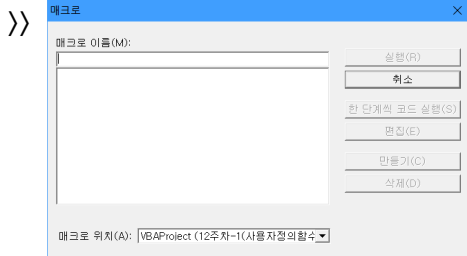
c=b.value

End If

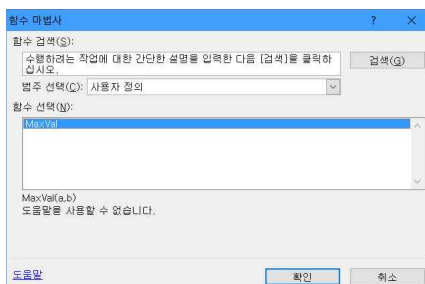
Max Val=c



>> Public Function 삽입 또는 직접 입력하는법 2가지 존재



>> 위의 코드를 입력했음에도 매크로 목록에 뜨지 x
“Function” 사용시 함수로 추가되어
함수 마법사-사용자 정의에서 찾아야함



>> 함수마법사 (f(x)버튼) 누른 후 사용자 정의에서
Function 프로시저로 작성한 함수가 추가된 것을
확인 가능!!

6	홍길동	34,560	789,900	(B6,C6)
7	함수 인수			? x
8	MaxVal			
9	A B6	= 34560		
	B C6	= 789900		

>>다음과 같이 두 변수를
지정하여 두 수중 더 큰
수를 찾아낼 수 있음

* “=MaxVal(A,B)” 함수를 셀에 직접 작성할 수도 있음!

***사용자 정의 폼**

:엑셀작업을 보다 편리하게 처리하기 위해 폼을 만들어 사용(Visual Studio와 같은 느낌)

- [개발 도구]-[vb]-[삽입]-사용자 정의 폼

*엑셀에서 사용자 정의 폼을 만들어 쓸 때 장점: 엑셀의 함수를 폼에 적용시킬 수 있다.

-주의사항

- 1) 엑셀에서 바로 실행 불가능
- 2) 반드시 **매크로로 사용자 정의 폼을 연결** 해주어야함 >> 코드 작성 시 Sub폼 이용
- 3) 따라서 코드 작성시 ★**사용자 정의 폼 명령어(.show)**를 입력 해주어야함.

*** 폼 실행 명령어**

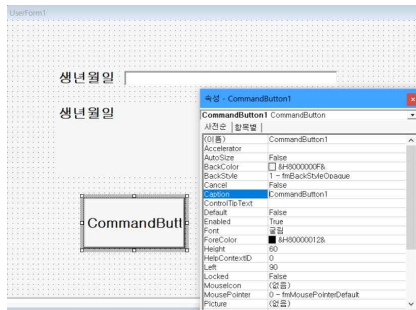
```
Sub 폼열기()
userform1.show
end sub
```

***사용자 정의 폼을 이용하여 나이 구하는 프로그램 만들기**

- 사용되는 함수 설명

- 1) DateDiff 함수: 두 날짜 사이의 기간을 구하는 함수.
>> y 입력시 년도/ m입력시 월/ d입력시 일을 함수가 구해줌
- 2) Cdate 함수: 입력된 값을 날짜로 변환해주는 함수
- 3) Date: 오늘 날짜를 구해주는 함수

적용>> DateDiff("yyyy","mm","dd",Cdate(textbox N. text),Date)



>>도구의 텍스트를 수정하기 위해서는 VS와 달리
"caption"이라는 속성을 수정해야함

〈사용자 정의 폼 명령어:Button1〉

```
Private Sub CommandButton1_Click()
```

```
Dim result As Integer
```

```
result = DateDiff("yyyy", CDate(TextBox1.Text), Date)
```

```
Label2.Caption = "당신의 나이는" & result +  
1 & "세 입니다."
```

```
End Sub
```

〈폼 실행 명령어〉

Module 추가 후

```
Sub 폼열기()
```

```
UserForm1.Show
```

```
End Sub
```

입력

***주민등록번호를 입력받아 성별을 구하는 함수**

-사용되는 함수 설명

: mid(텍스트 함수): 변수=mid(주민번호.value,8,1)

>>'주민번호'라는 텍스트 값의 8번째 자리 수를 한 개만 뽑는다.

(주민번호에 들어가는 하이픈(-)도 자리수에 포함됨(문자라서))

-코드

Public Function 성별(주민번호 as range)

Dim l as Single

l=mid(주민번호.value,8,1)

if l=1 or l=3 then

성별="남자"

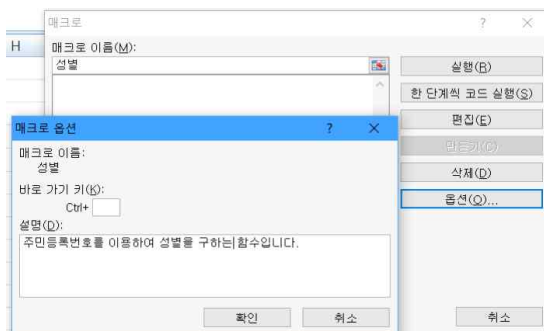
else

성별="여자"

end if

end function

이름	주민등록번호	성별	이름	주민등록번호	성별
차미양	891010-2539123	=성별(C5)	차미양	891010-2539123	여자
소보희	901212-2765789		소보희	901212-2765789	여자
김진혁	771111-1928011		김진혁	771111-1928011	남자
정영수	980123-1919191		정영수	980123-1919191	남자
김미여	883456-2098757		김미여	883456-2098757	여자
김경애	057867-4906565		김경애	057867-4906565	여자
이경호	021278-3990087		이경호	021278-3990087	남자
이미진	791212-2929345		이미진	791212-2929345	여자



-함수 마법사에서 '성별'함수의 설명달기

:매크로 보기 >> 성별 검색 >> 옵션

>> 설명 적기 >> 확인

<WorkSheet-시트 1개일 때, Worksheets-시트 여러 개일 때 개체>

-WorkSheet 속성

: 작업 중에 원하는 특정 시트를 선택하기 위해 사용

(1)Worksheets("시트이름").Select : 시트이름=문자

(2)Worksheets(시트번호).Select : 시트번호=숫자

-ActiveSheet: 현재 마우스 포인트로 선택되어 있는 시트

ex)ActiveSheet.Range("A5").Value=15 (Range("A5")=15와 동일)

*시트선택하기

Sub 시트선택()

Worksheets(1).Select

Range("b15").Value = "신소재공학과"

End Sub

*시트 추가

Sub 시트추가()

Worksheets.Add

End Sub

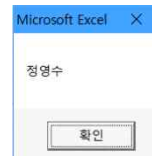
*시트 선택 후 메시지박스로 데이터 보여주기

Sub 시트선택후데이터보여주기()

Worksheets(1).Select

MsgBox Range("b8").Value

End Sub



〈Application개체〉

1.StatusBar: 상태 표시줄의 표시 여부를 설정하거나 메시지를 표시하는 등의 작업을 처리

1) Application.DisplayStatusBar=True/False (상태 표시줄을 보일 것인지 여부)

2) Application.StatusBar="메세지입력"

-WorksheetFunction: 비주얼 베이직 함수가 아닌 엑셀 함수를 VBA에서 사용할 때

>>Application.WorksheetFunction.함수명 (의 형태로만 사용가능)

***메세지 박스로 Statusbar 표시여부 선택 후 표시**

Sub 상태표시줄()

Dim x As Integer

Application.DisplayStatusBar = True

Application.StatusBar = "신소재공학과 최준용 작성 2021.12.21 문의사항 010-3929-7793
문의바람."

x = MsgBox("상태표시줄 선택하기", vbYesNo, "상태표시줄")

>>변수=msg() 형식 지키기/True=6,False=7이므로,

If x = 6 Then **>>6=msg박스의 "예(True)"**

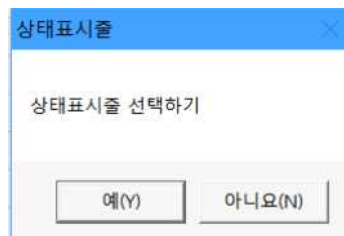
Application.DisplayStatusBar = True

Else

Application.DisplayStatusBar = False

End If

End Sub



신소재공학과 최준용 작성 2021.12.21 문의사항 010-3929-7793 문의바람.

2.PrintOut: 인쇄하기

-PrintPreview: 프린트 작동을 확인하기 위한 미리보기 기능

>>ActiveSheet.PrintPreview: 띄워져있는 시트 인쇄 미리보기

>>workSheets(3).PrintPreview: 특정 시트(n번째 시트=" "없이, 시트이름으로 지정 시=" ")

-PrintOut:인쇄하기

***선택 시트 인쇄 미리보기**

Sub 인쇄()

ActiveSheet.PrintPreview

End Sub

***특정 시트 미리보기**

Sub 특정시트미리보기()

Worksheets(2).PrintPreview

End Sub

***선택 시트 인쇄하기**

Sub 인쇄()

ActiveSheet.PrintOut

End Sub

***전체 시트 인쇄하기**

Sub 시트개수만큼미리보기()

Dim i As Integer

For i = 1 To Worksheets.Count >>전체 시트 개수 변동 될 수 있음>>Worksheet.Count

Worksheets(i).PrintPreview

Next >> 반복문(For)를 끝내기 위해 Next 꼭 적어줘야함!!!

End Sub

∴ 매크로 실행 시 각 시트를 한 장씩 인쇄 미리보기 가능

13-2)

〈자동데이터 입력하기로 전화요금표 만들기〉

-Worksheet데이터형: VB의 long, integer 다르게 VBA에서 적용되는 Range의 데이터형
ex) Dim ws As Worksheet

Set ws = Sheets("전화요금")

〈표〉

전화번호	가입자코드	종별	지역명	통화수	기본요금	통화료	청구금액
284-1930	2	영업용	동부	452	56,700	430	251,060
411-3132	3	가정용	서부	263	34,200	370	131,510
472-8399	1	영업용	남부	523	45,100	420	264,760
123-4050	2	다목적용	남부	722	56,200	380	330,560
432-1945	3	영업용	북부	459	36,700	420	229,480
829-8982	3	가정용	서부	231	63,400	380	151,180
123-4567	2	영업용	동부	535	51,800	320	353,512
411-3920	3	영업용	동부	521	63,100	360	250,660
833-9292	2	영업용	서부	527	48,900	250	180,650
934-5321	1	다목적용	북부	642	52,900	410	316,120
234-8345	2	영업용	남부	623	62,300	320	261,660
823-5876	3	영업용	서부	581	51,900	280	214,580

〈양식〉

통 화 내 역 확 인 서	
전 화 번 호 :	
종 별 :	
통 화 수	
청 구 금 액	

>>전화번호/종별/통화수/청구금액 표시 셀에

각각의 이름 정하기>>range("전화번호")지정가능.

-코드

Sub 전화요금()

Dim ws As Worksheet

Dim irow As Integer

Set ws = Sheets("전화요금")

irow=InputBox("행번호 입력(4~15중 입력)","전화요금표"))>>application없이 행번호 입력받기 위함

Range("전화번호") = ws.Cells(irow, 2)>>입력된 행의 전화번호 얻기 가능

Range("종별") = ws.Cells(irow, 4)>>입력된 행의 종별 얻기 가능

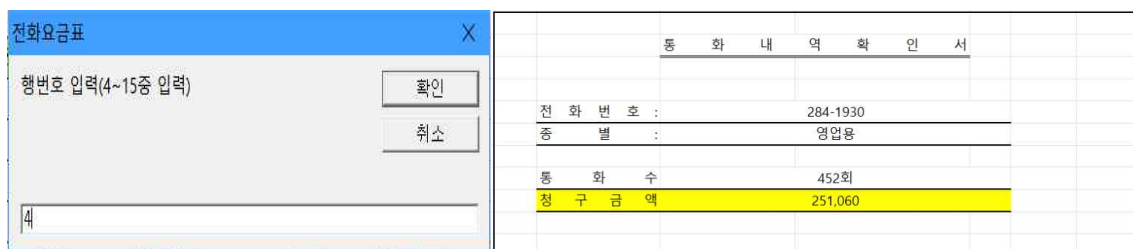
Range("통화수") = Format(ws.Cells(irow, 6), "###회") >>입력된 행의 통화수 얻기 o

Range("청구금액")=Format(ws.Cells(irow, 9), "#,##0"))>> 표시할 때 #,##0으로 표시하라는뜻(돈)

If MsgBox("인쇄 하시겠습니까?", vbYesNo, "전화요금") = vbYes Then

ActiveSheet.PrintPreview >>바로 인쇄 가능하도록함(주의점: 버튼이 인쇄되는시트에 있어야함!)

End Sub



>>4행에 대한 통화내역확인서, 인쇄 미리보기 나옴

14주차)절대 참조하는 매크로+주문조회 프로그램 만들기

-절대/상대 참조 매크로 (29페이지 참고)

*부분합

-부분합 사용시 주의점

1)부분합을 하기전에 반드시 정렬이 되어 있어야함

2)정렬을 하기 전 부분합을 제거

3)정렬 후 원하는 부분합 기능 활용 가능>>합계/평균 등

4)정렬은[정렬]-[정렬 및 필터]-부분합 할 항목을[오름차순 정렬]or[내림차순 정렬]

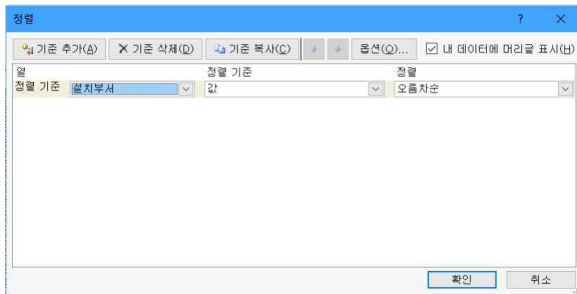
-부분합이란?

:워크시트에 있는 데이터를 일정한 기준으로, 즉 그룹별로 통계처리를 수행하는 기능(예:부서별..)

>> [데이터]-[유효성]-[부분합]으로 진입

-정렬이란?

:워크시트에 있는 데이터를 오름/내림차순으로 정렬



<부서별로 정렬>

>>>

<부서별로 부분합>

>>이후 부분합 다시 누르고 제거 누르면 부분합 취소됨

*부분합으로 데이터 분석 매크로 만들기

설치부서	비품명	구매일자	내림차순	정렬년수	취득원가	잔존가액	평가상장액	평가금액	불량수준
경영팀	케비넷	00-10-08	5	21	800,000	80,000	3,024,000	-2,224,000	C
경영팀	컴퓨터	02-04-01	5	19	1,200,000	120,000	4,104,000	-2,904,000	C
경영팀	컴퓨터	02-07-10	5	19	1,200,000	120,000	4,104,000	-2,904,000	C
경영팀	소파	01-12-31	8	19	500,000	50,000	1,068,000	-568,000	C
경영팀	사무용책상	99-11-01	8	22	700,000	70,000	1,782,000	-1,082,000	C
영업팀	프린터기	03-02-08	4	18	200,000	20,000	1,417,000	-1,087,000	C
영업팀	태이블	02-01-08	7	19	400,000	40,000	977,143	-577,143	C
영업팀	케비넷	00-06-30	5	21	800,000	80,000	3,024,000	-2,224,000	C
인사팀	프린터기	03-02-08	4	18	200,000	20,000	1,417,000	-1,087,000	C
인사팀	태이블	01-09-30	7	20	400,000	40,000	1,028,571	-628,571	C
인사팀	컴퓨터	00-09-31	5	21	1,500,000	150,000	5,670,000	-4,170,000	C
인사팀	영입팀	00-12-09	5	21	1,500,000	150,000	5,670,000	-4,170,000	C
전산팀	프린터기	01-05-02	4	20	350,000	35,000	1,575,000	-1,225,000	C
전산팀	프린터기	01-08-10	4	20	350,000	35,000	1,575,000	-1,225,000	C
전산팀	컴퓨터	01-05-06	5	20	1,200,000	120,000	4,320,000	-3,120,000	C
전산팀	컴퓨터	01-08-14	5	20	1,200,000	120,000	4,320,000	-3,120,000	C
전산팀	모니터	01-08-14	5	20	340,000	34,000	1,234,000	-894,000	C
전산팀	모니터	01-09-30	5	20	340,000	34,000	1,234,000	-894,000	C
총무팀	케비넷	00-08-31	5	21	800,000	80,000	3,024,000	-2,224,000	C
총무팀	케비넷	00-12-09	5	21	800,000	80,000	3,024,000	-2,224,000	C
총무팀	소파	02-04-14	8	19	600,000	60,000	1,282,500	-682,500	C
총무팀	소파	02-01-04	8	19	600,000	60,000	1,282,500	-682,500	C
총무팀	모니터	02-11-09	5	19	340,000	34,000	1,152,000	-812,000	C
무팀	요약							3,140,000	
총합계								17,480,000	

1.설치부서별 정렬 매크로

매크로 기록>>설치부서 항목 제일 윗칸의 셀 선택>>설치부서별정렬>>기록중지

2.설치부서별 부분합 매크로

매크로 기록>>설치부서 항목 제일 윗칸 셀 선택>>부분합>>

부분합 할 항목(숫자는 숫자끼리,문자는 문자끼리!!!혼입 되면 안됨!!)>>>기록중지

3.부분합 제거(다른 항목에도 공통적으로 사용 가능)

매크로 기록>>부분합>>제거>>기록중지

4.비품명별 정렬/부분합 하셈

5.버튼에 집어넣고 눌러보면 부분합에서 다른 부분합 넘어갈때 오류뜸

:모듈 드가서 각 정렬/부분합 코드 이름(ex.설치부서정렬매크로)아래칸에 "call 부분합제거" 써놓으셈

>>이 때 "call 부분합제거"앞에 따옴표있으면 지우셈 초록색뜨면 안됨

6.이거고쳐놔더니 부분합>부분합누르면 정렬안된채로 부분합됨: 위에 call 밑에 "call~정렬" 입력

설치부서	비품명	구매일자	내림차순	정렬년수	취득원가
경영팀	케비넷	00-10-08	5	21	800,000
경영팀	컴퓨터	02-04-01	5	19	1,200,000
경영팀	컴퓨터	02-07-10	5	19	1,200,000
경영팀	소파	01-12-31	8	19	500,000
경영팀	사무용책상	99-11-01	8	22	700,000
영업팀	프린터기	03-02-08	4	18	200,000
영업팀	태이블	02-01-08	7	19	400,000
영업팀	케비넷	00-06-30	5	21	800,000
인사팀	프린터기	03-02-08	4	18	200,000
인사팀	태이블	01-09-30	7	20	400,000
인사팀	컴퓨터	00-09-31	5	21	1,500,000
인사팀	컴퓨터	00-12-09	5	21	1,500,000
전산팀	프린터기	01-05-02	4	20	350,000
전산팀	프린터기	01-08-10	4	20	350,000
전산팀	컴퓨터	01-05-06	5	20	1,200,000
전산팀	컴퓨터	01-08-14	5	20	1,200,000
전산팀	모니터	01-08-14	5	20	1,200,000
전산팀	모니터	01-09-30	5	20	340,000
총무팀	케비넷	00-08-31	5	21	800,000
총무팀	케비넷	00-12-09	5	21	800,000
총무팀	소파	02-04-14	8	19	600,000
총무팀	소파	02-01-04	8	19	600,000
총무팀	모니터	02-11-09	5	19	340,000
무팀	요약				3,140,000
총합계					17,480,000



설치부서	비품명	구매일자	내용년수	경과년수	취득원가	잔존가액	감가상각액	장부가액	품질수준
경리팀	사무용책상	99-11-01	8	22	700,000	70,000	1,732,500	-1,032,500	C
경리팀	소파	01-12-31	8	19	500,000	50,000	1,068,750	- 568,700	C
경리팀	컴퓨터	02-04-01	5	19	1,200,000	120,000	4,104,000	-2,904,000	C
경리팀	컴퓨터	02-07-10	5	19	1,200,000	120,000	4,104,000	-2,904,000	C
경리팀	케비넷	00-10-08	5	21	800,000	80,000	3,024,000	-2,224,000	C
경리팀 요약				100	4,400,000	440,000			
영업팀	케비넷	00-06-30	5	21	800,000	80,000	3,024,000	-2,224,000	C
영업팀	테이블	02-01-08	7	19	400,000	40,000	977,143	- 577,100	C
영업팀	프린터기	03-02-08	4	18	350,000	35,000	1,417,500	-1,067,500	C
영업팀 요약				58	1,550,000	155,000			
인사팀	컴퓨터	00-08-31	5	21	1,500,000	150,000	5,670,000	-4,170,000	C

7.셀선택: 이거 그냥 정렬이나 부분합전에 표 밖에 셀 선택되었으면 오류떠서 표 첫칸 선택하는거
>>다하면 부분합 제거 프로시저 가서 젤 밑에 call로 쏘셔박기 ㄱㄱ

8.용지방향가로: 뽑아야되는데 세로라 잘 안보임 가로로정렬 매크로 ㄱ
매크로 기록>> 페이지레이아웃>>가로>>기록중지

9.미리보기: 모듈에

Sub 프린트미리보기()

call 용지방향가로 >>>>>이거 해놓으면 가로버튼 안눌러도됨

Activsheet.PrintPreview

End Sub

4.선적 선택

>>그룹박스 만들고“선적구분을 선택하시오”입력>>옵선단추3개 만들기(셀 연결전 좌클릭 금지!)>>옵선단추 우클릭>> 컨트롤서식>> 셀연결(빈셀)

선적구분을 선택하시오		
<input type="radio"/> 옵선 단추 3	<input type="radio"/> 옵선 단추 6	<input checked="" type="radio"/> 옵선 단추 7
3		

5.이대로 만들기

주문 횟수:		선적구분을 선택하시오						
직원:	윤대현 ▼	<input checked="" type="radio"/> 국제특송	<input type="radio"/> 대륙통운	<input type="radio"/> 한일특급				
주문번호	고객	직원	수주일	납기일	선적일	선적 구분	운임	수하인 주소
		윤대현 3	>>콤보박스 연결 셀			국제 특송 (₩)	1	>>옵선 단추 연결 셀
주문번호	고객	직원	수주일	납기일	선적일	선적 구분	운임	수하인 주소
0001	해바라기 백화점 (₩)	안정훈	2003-07-03	2003-07-31	2003-07-15	국제 특송 (₩)	32,000	도봉구 창 1동 218-16
0002	산타페 (₩)	김찬진	2003-07-04	2003-08-15	2003-07-09	대륙 통운 (₩)	12,000	중구 원미 1동 120-16
0003	삼왕 통상 (₩)	윤대현	2003-07-07	2003-08-04	2003-07-14	대륙 통운 (₩)	41,000	중구 답동 78
0004	삼양 상사 (₩)	최소라	2003-07-08	2003-08-05	2003-07-10	한일 특급 (₩)	51,000	용산구 서빙고동 111-16

새로 만든 위에 목록 ‘직원 셀’ 아래 Index함수 사용:

=INDEX(조건!A4:A12,주문목록!D9(=콤보박스 연결 셀))

>>콤보박스에서 직원이름 선택 시 Index함수 입력한 셀에 그대로 표시

>>선적 구분도 동일히 진행

6.원본 표로 메크로 필터링+해제

-필터링: 메크로 기록>>이름 정하고 확인 누르면 기록 시작>>데이터-고급필터>>

목록범위:원본 표/조건 범위:초록색 항목라인+아래행>>확인>>기록중지

-해제: 메크로 기록>>이름 정하고 확인>> 데이터-지우기 >>기록 종료

7.메크로 버튼이랑 연결: 이걸 알아서하셈

8.숫자 보이는 셀+조건들 안보이게 하기:

숫자 보이는 셀 서식>> 사용자 지정 >> 형식: “세미콜론(;)”만 입력

+ 목록, 이름 뜨는 행 선택>>우클릭>>숨기기

주문 횟수:		선적구분을 선택하시오						조회
직원:	김찬진 ▼	<input checked="" type="radio"/> 국제특송	<input type="radio"/> 대륙통운	<input type="radio"/> 한일특급				해제
주문번호	고객	직원	수주일	납기일	선적일	선적 구분	운임	수하인 주소
0047	알프스 무역 (₩)	김찬진	2004-04-16	2004-05-14	2004-04-26	국제 특송 (₩)	368,000	강정동 718-1
0067	알프스 무역 (₩)	김찬진	2004-05-31	2004-06-28	2004-06-02	국제 특송 (₩)	252,000	강정동 718-1
0194	삼영 식품 (₩)	김찬진	2005-01-06	2005-02-03	2005-01-14	국제 특송 (₩)	7,000	가학동 301
0226	삼양 상사 (₩)	김찬진	2005-02-10	2005-03-10	2005-02-16	국제 특송 (₩)	6,000	용산구 서빙고동 111-16
0264	우주 상사 (₩)	김찬진	2005-03-18	2005-04-15		국제 특송 (₩)	144,000	도봉구 미아 7동 334-7

∴ 김찬진 직원이 국제특송을 이용한 주문만 필터링 완료