

Chapter 06
자료 구조



목차

1. 배열
2. 정적 배열, 동적 배열, ArrayList
3. 구조체

1. 배열

■ 배열의 개념

- 데이터형이 같고 관련된 내용의 데이터 여러 개를 하나의 이름으로 묶어서 관리할 때 사용
- 배열에 포함된 하나하나의 변수가 배열의 요소
- 한 배열에 포함된 요소의 총 개수가 배열의 크기
- 배열 내에서 요소를 구분하기 위해 특정 변수의 위치를 가리키는 첨자라는 숫자를 사용하는데, 이 첨자는 0부터 시작
- 배열의 형태
 - 배열 구조에 따라 1차원, 2차원, 3차원 배열 등
 - 기억 장소를 할당하는 시점에 따라 컴파일 시 할당하는 정적 배열, 프로그램 실행 시 할당하는 동적 배열

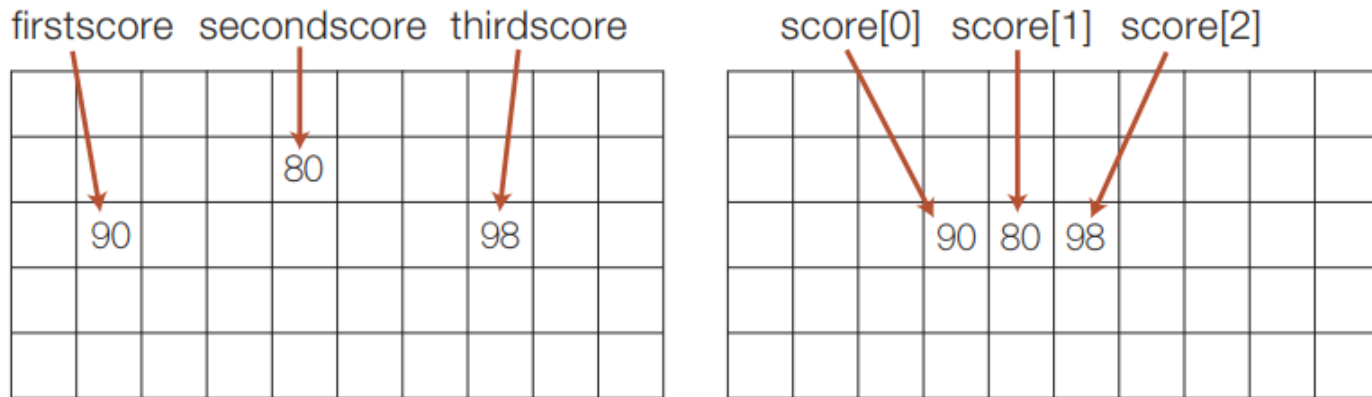


그림 6-1 3개의 일반 변수와 배열의 메모리 배치 비교

1. 배열

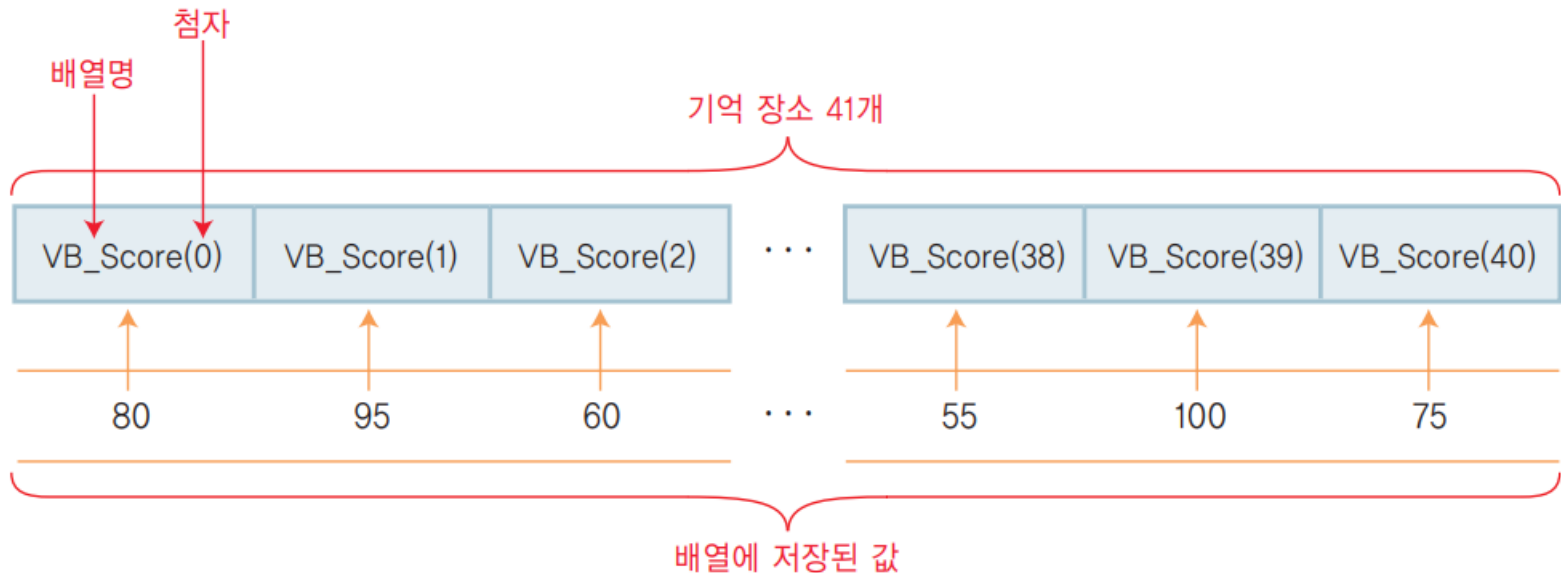
■ 1차원 배열의 선언과 초기화

■ 배열 선언

- 일반 변수를 선언할 때처럼 Dim, Public, Private으로 선언
- 배열 이름 뒤에 괄호를 사용하고 괄호 안에 배열의 크기를 지정
- As 뒤에 데이터형을 지정

형식 | [Dim | Public | Private] 배열명(크기) [As 데이터형]

예 | Dim VB_Score(40) As Integer



1. 배열

- 배열의 첨자를 1부터 시작하도록 설정

- 방법1. 모듈의 선언부에 'Option Base 1'이라고 선언

```
Option Base 1
```

```
Sub Array_Test()
```

```
    Dim tel(3) As String → tel(1), tel(2), tel(3) 변수 3개를 사용한다고 선언
```

```
End Sub
```

- 방법2. to를 이용하여 첨자의 범위를 지정

```
Sub Array_Test()
```

```
    Dim tel(1 to 3) As String → tel(1), tel(2), tel(3) 변수 3개를 사용한다고 선언
```

```
    Dim su(10 to 12) As Integer → su(10), su(11), su(12) 변수 3개를 사용한다고 선언
```

```
End Sub
```

1. 배열

■ 배열 초기화

- 선언된 배열의 변수에 처음 값을 대입하는 것

```
Dim jumsu(3) As Integer
```

jumsu(0) = 100 → 각 배열의 요소를 일일이 초기화

jumsu(1) = 90

jumsu(2) = 95

jumsu(3) = 85

- 배열의 선언과 초기화를 동시에 하는 방법

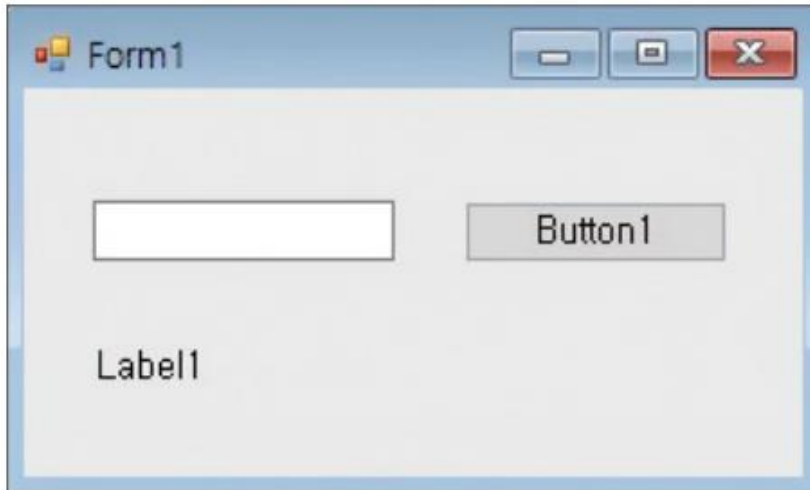
```
Dim jumsu() As Integer = {100, 90, 95, 85}
```

```
Dim Name() As String = {"홍길동", "이몽룡", "성춘향", "임꺽정"}
```

1. 배열

실습하기 6-1 1차원 배열을 이용하여 알파벳 출력하기

1) 윈도우 폼 디자인



2) 속성 설정

컨트롤	속성	속성값
TextBox1	Text	(빈칸)
Button1	Text	보기
Label1	Text	(빈칸)

1. 배열

실습하기 6-1 1차원 배열을 이용하여 알파벳 출력하기

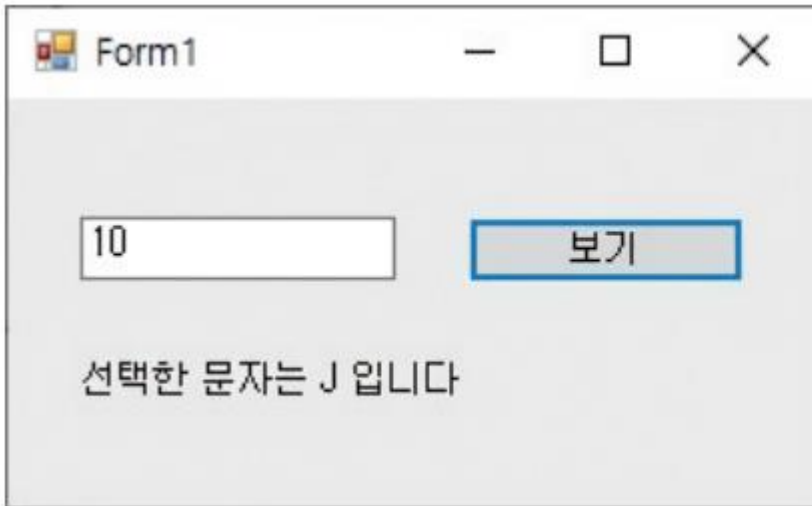
3) 코드 작성

```
01 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.  
    Click  
02     Dim Alphabet() As String = {"A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K",  
        "L", "M", "N", "O", "P", "Q", "R", "S", "T", "U", "V", "W", "X", "Y", "Z"}  
03     Dim order As Integer  
04     order = TextBox1.Text  
  
05     If order >= 1 And order <= 26 Then  
06         Label1.Text = "선택한 문자는 " & Alphabet(order - 1) & "입니다"  
07     Else  
08         Label1.Text = "1~26 사이의 숫자를 입력하십시오."  
09     End If  
10 End Sub
```


1. 배열

실습하기 6-1 1차원 배열을 이용하여 알파벳 출력하기

4) 실행 결과 확인

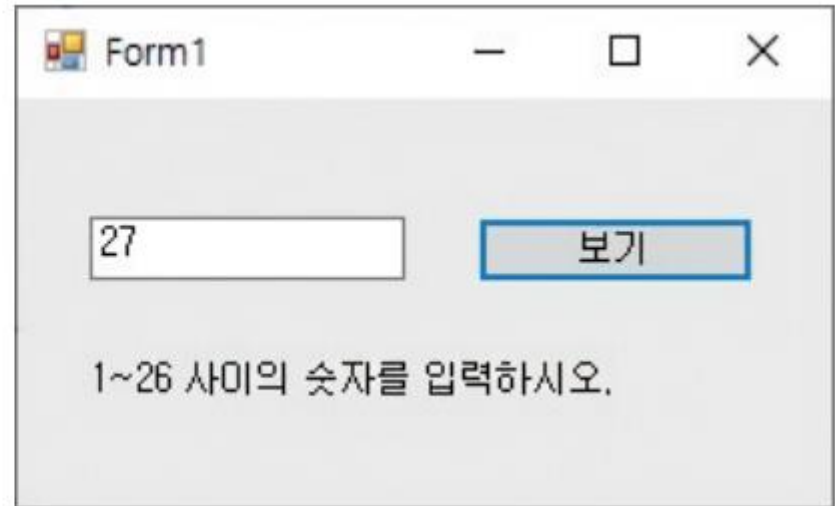


Form1

10

보기

선택한 문자는 J 입니다



Form1

27

보기

1~26 사이의 숫자를 입력하시오.

1. 배열

■ 다차원 배열

- 2개 이상의 첨자를 사용하는 경우

형식 | [Dim | Public | Private] 배열명(행 크기, 열 크기) [As 데이터형]

예 | Dim VB_Score(3, 4) As Integer

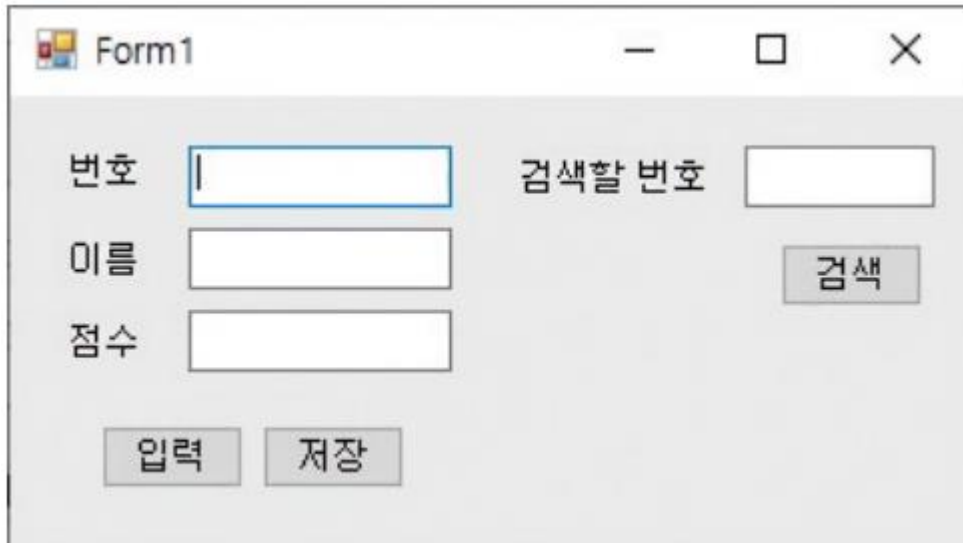
(0,0)	(0,1)	(0,2)	(0,3)	(0,4)
(1,0)	(1,1)	(1,2)	(1,3)	(1,4)
(2,0)	(2,1)	(2,2)	(2,3)	(2,4)
(3,0)	(3,1)	(3,2)	(3,3)	(3,4)

그림 6-3 2차원 배열의 논리적 메모리 배치

1. 배열

실습하기 6-2 2차원 배열을 이용하여 성적 저장하고 검색하기

- 실행 결과

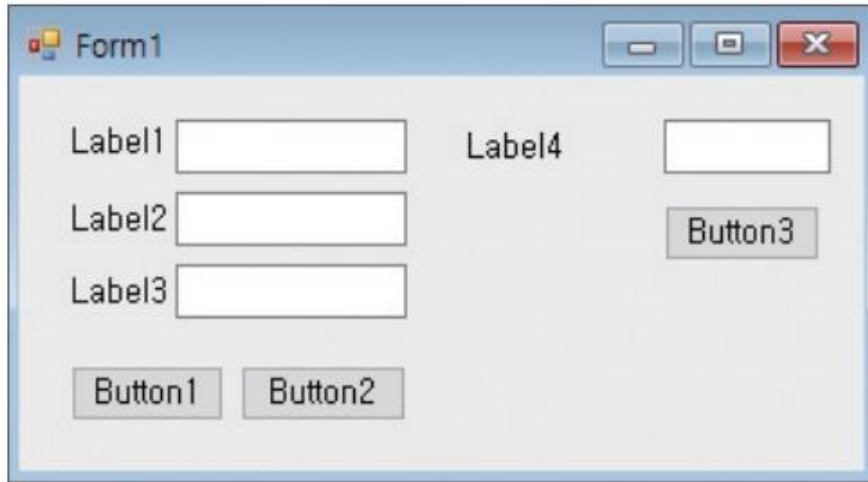


The screenshot shows a Windows application window titled "Form1". Inside the window, there are three text input fields on the left, each with a label to its left: "번호" (Number) with a value of "1", "이름" (Name), and "점수" (Score). To the right of these fields is a label "검색할 번호" (Search Number) followed by another text input field. Below the "이름" and "점수" fields are two buttons: "입력" (Input) and "저장" (Save). To the right of the "검색할 번호" field is a button labeled "검색" (Search).

1. 배열

실습하기 6-2 2차원 배열을 이용하여 성적 저장하고 검색하기

1) 윈도우 폼 디자인



2) 속성 설정

컨트롤	속성	속성값	컨트롤	속성	속성값
Label1	Text	번호	TextBox1~TextBox4	Text	(빈칸)
Label2	Text	이름	Button1	Text	입력
Label3	Text	점수	Button2	Text	저장
Label4	Text	검색할 번호	Button3	Text	검색

1. 배열

실습하기 6-2 2차원 배열을 이용하여 성적 저장하고 검색하기

3) 코드 작성

```
01 Public Class Form1
02     Dim student(3, 3) As String
03     Dim i As Integer = 0

04     Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.
        Click
05         i = i + 1
06         If i <= 3 Then
07             TextBox1.Text = i
08             TextBox2.Text = ""
09             TextBox3.Text = ""
10             TextBox2.Focus()
11         Else
12             TextBox1.Text = i
13             MessageBox.Show("입력이 불가능합니다.", "입력")
14             TextBox1.Text = ""
15         End If
16     End Sub
```

1. 배열

실습하기 6-2 2차원 배열을 이용하여 성적 저장하고 검색하기

3) 코드 작성

```
17 Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.  
    Click  
18     If i <= 3 Then  
19         student(TextBox1.Text, 1) = TextBox1.Text  
20         student(TextBox1.Text, 2) = TextBox2.Text  
21         student(TextBox1.Text, 3) = TextBox3.Text  
22         TextBox1.Text = ""  
23         TextBox2.Text = ""  
24         TextBox3.Text = ""  
25     Else  
26         MessageBox.Show("저장이 불가능합니다.", "저장")  
27     End If  
28 End Sub
```

1. 배열

실습하기 6-2 2차원 배열을 이용하여 성적 저장하고 검색하기

3) 코드 작성

```
29 Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.  
    Click  
30     If TextBox4.Text Like "[1-3]" Then  
31         TextBox1.Text = student(TextBox4.Text, 1)  
32         TextBox2.Text = student(TextBox4.Text, 2)  
33         TextBox3.Text = student(TextBox4.Text, 3)  
34         TextBox4.Focus()  
35     Else  
36         MessageBox.Show("찾는 데이터가 없습니다.", "검색")  
37         TextBox4.Focus()  
38     End If  
39 End Sub  
40 End Class
```

1. 배열

실습하기 6-2 2차원 배열을 이용하여 성적 저장하고 검색하기

4) 프로그램을 실행하고 다음과 같이 학생 3명의 정보를 입력(네 번째 학생부터는 입력 X)

번호	이름	점수
1	이순신	100
2	원균	90
3	유성룡	98

Form1 window showing the initial state. It has three input fields: '번호' (Number) with value '1', '이름' (Name) with value '이순신', and '점수' (Score) with value '100'. There is a '검색할 번호' (Search Number) field and a '검색' (Search) button. At the bottom, there are '입력' (Input) and '저장' (Save) buttons.

Form1 window showing the state after entering the second student's information. The '번호' field has '2', '이름' has '원균', and '점수' has '90'. The '입력' and '저장' buttons are visible.

Form1 window showing the state after entering the third student's information. The '번호' field has '3', '이름' has '유성룡', and '점수' has '98'. The '입력' and '저장' buttons are visible.

Form1 window showing the state after entering the fourth student's information. The '번호' field has '4', '이름' is empty, and '점수' is empty. The '입력' and '저장' buttons are visible.

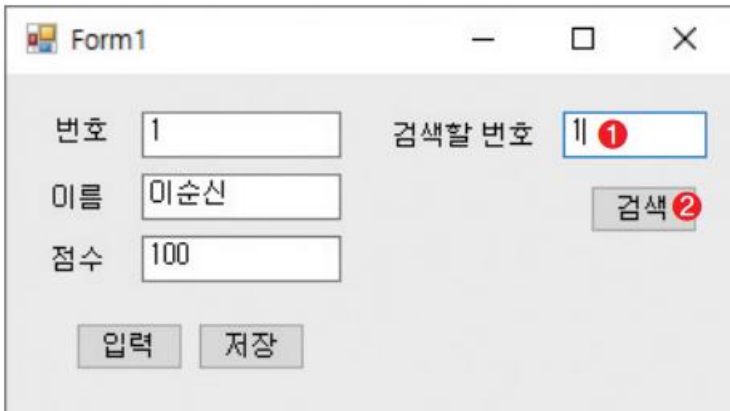
입력 (Input) dialog box showing an error message: '입력이 불가능합니다.' (Input is not possible). There is an '확인' (Confirm) button at the bottom.

1. 배열

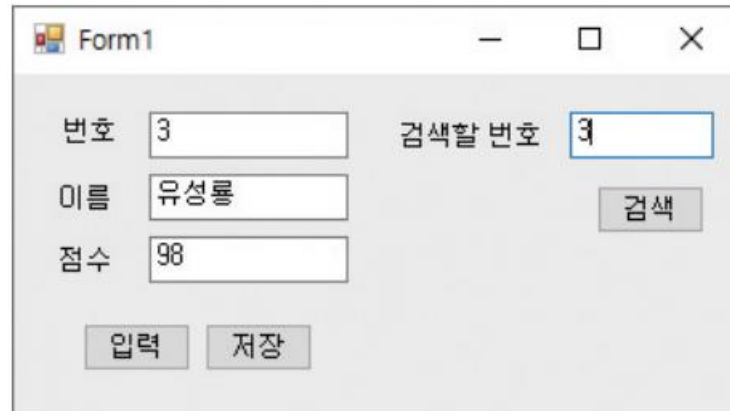
실습하기 6-2 2차원 배열을 이용하여 성적 저장하고 검색하기

5) 찾아볼 번호를 입력하고 <검색>을 클릭하면 해당 학생의 정보가 나타남

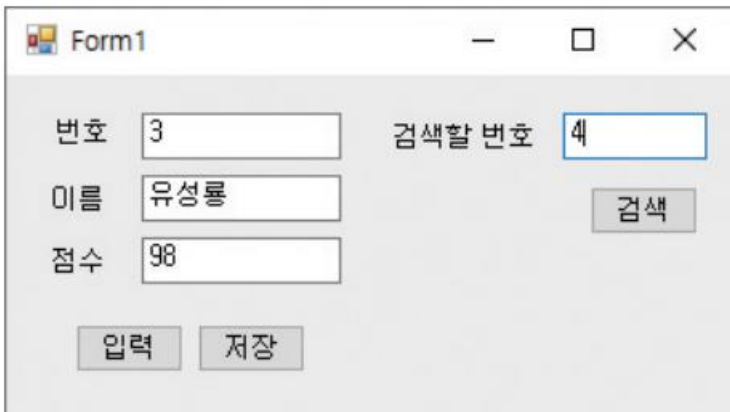
- 입력한 번호가 1~3이 아니면 오류 메시지가 나타남



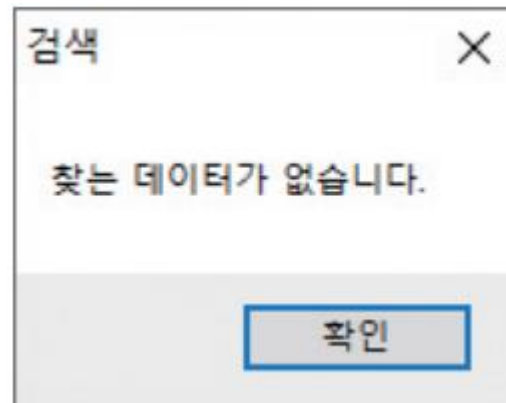
Form1 window showing input for student 1. The '번호' (Number) field contains '1'. The '검색할 번호' (Search Number) field contains '1' with a red '1' next to it. The '이름' (Name) field contains '이순신'. The '점수' (Score) field contains '100'. The '검색' (Search) button is highlighted with a red '2' next to it. The '입력' (Input) and '저장' (Save) buttons are also visible.



Form1 window showing input for student 3. The '번호' (Number) field contains '3'. The '검색할 번호' (Search Number) field contains '3'. The '이름' (Name) field contains '유성룡'. The '점수' (Score) field contains '98'. The '검색' (Search) button is visible. The '입력' (Input) and '저장' (Save) buttons are also visible.



Form1 window showing input for student 3. The '번호' (Number) field contains '3'. The '검색할 번호' (Search Number) field contains '4'. The '이름' (Name) field contains '유성룡'. The '점수' (Score) field contains '98'. The '검색' (Search) button is visible. The '입력' (Input) and '저장' (Save) buttons are also visible.



Search error message dialog. The title is '검색' (Search). The message is '찾는 데이터가 없습니다.' (No data found). The '확인' (Confirm) button is visible.

2. 정적 배열, 동적 배열, ArrayList

■ 정적 배열

- 배열을 선언할 때 크기를 결정하고 다시 바꿀 수 없음

■ 동적 배열

- 배열 선언 시 크기를 지정하지 않음
- 먼저 첨자 없이 빈 괄호로 배열을 선언하고 필요할 때 ReDim 문으로 배열의 크기를 재지정
- 배열의 크기를 다시 지정하면 이전 값이 삭제되며, Preserve 옵션을 사용하여 이전 값을 보존할 수 있음

형식 | Dim 배열명() [As 데이터형]

ReDim [Preserve] 배열명(첨자)

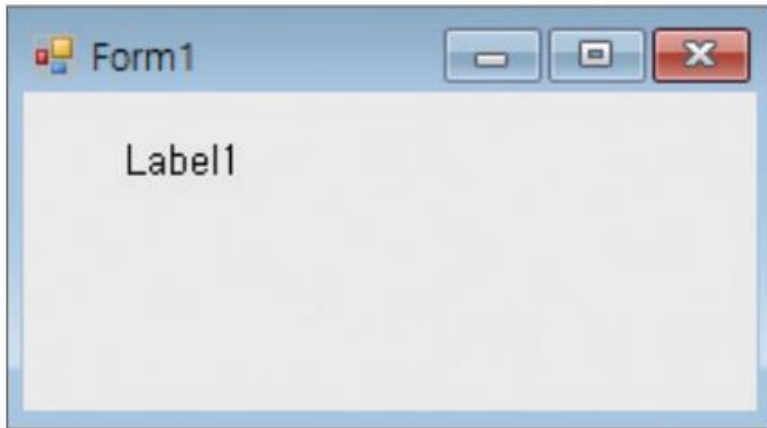
예 | Dim DynamicArray() As Integer

ReDim DynamicArray(5)

2. 정적 배열, 동적 배열, ArrayList

실습하기 6-3 정적 배열을 이용하여 레이블에 음계 출력하기

1) 윈도우 폼 디자인



2) 속성 설정

컨트롤	속성	속성값
Form1	Text	음계명
Label1	Text	(빈칸)

2. 정적 배열, 동적 배열, ArrayList

실습하기 6-3 정적 배열을 이용하여 레이블에 음계 출력하기

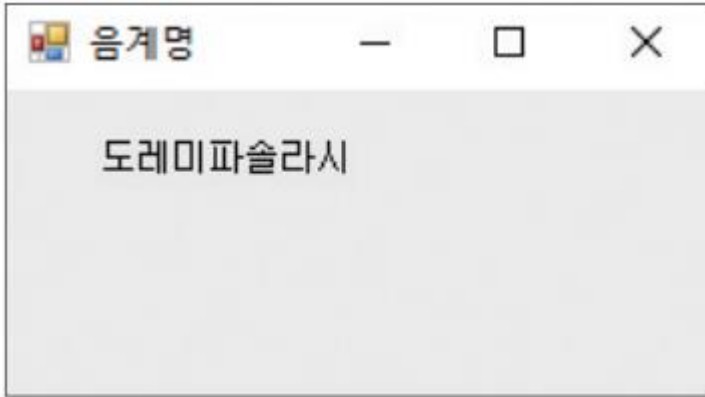
3) 코드 작성

```
01 Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.  
    Load  
02     Dim Doremi(7) As String  
03     Dim i As Integer  
  
04     Doremi(0) = "도"  
05     Doremi(1) = "레"  
06     Doremi(2) = "미"  
07     Doremi(3) = "파"  
08     Doremi(4) = "솔"  
09     Doremi(5) = "라"  
10     Doremi(6) = "시"  
  
11     For i = 0 To 6  
12         Label1.Text = Label1.Text + Doremi(i)  
13     Next i  
14 End Sub
```

2. 정적 배열, 동적 배열, ArrayList

실습하기 6-3 정적 배열을 이용하여 레이블에 음계 출력하기

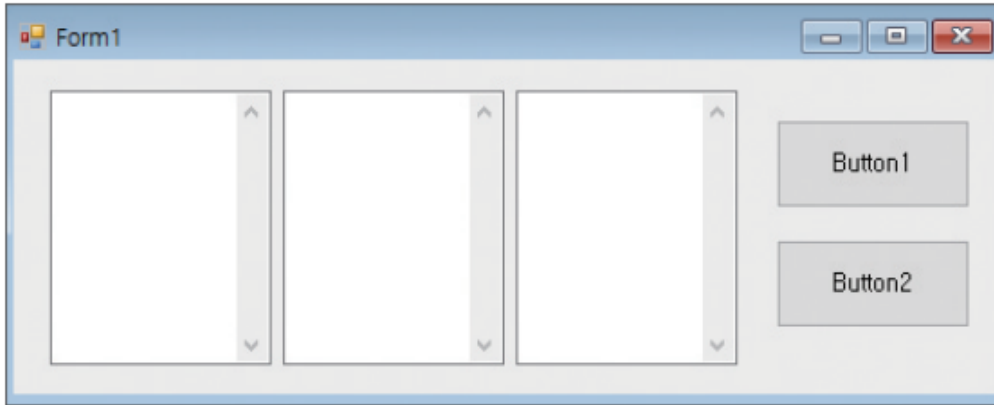
4) 실행 결과 확인



2. 정적 배열, 동적 배열, ArrayList

실습하기 6-4 ReDim 문을 이용하여 동적 배열 다루기

1) 윈도우 폼 디자인



2) 속성 설정

컨트롤	속성	속성값
Form1	Text	동적 배열
	Text	(빈칸)
TextBox1~TextBox3	MultiLine	True
	ScrollBars	Vertical
Button1	Text	ReDim
Button2	Text	ReDimPreserve

2. 정적 배열, 동적 배열, ArrayList

실습하기 6-4 ReDim 문을 이용하여 동적 배열 다루기

3) 코드 작성

```
01 Public Class Form1
02     Dim DynamicArray() As Integer
03     Dim i As Integer

04     Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.
        Load
05         ReDim DynamicArray(2)
06         For i = 1 To UBound(DynamicArray)
07             DynamicArray(i) = DynamicArray(i) + 8
08             TextBox1.Text = TextBox1.Text & "Array(" & i & ") = " & DynamicArray(i) & vbCrLf
09         Next
10     End Sub
```

2. 정적 배열, 동적 배열, ArrayList

실습하기 6-4 ReDim 문을 이용하여 동적 배열 다루기

3) 코드 작성

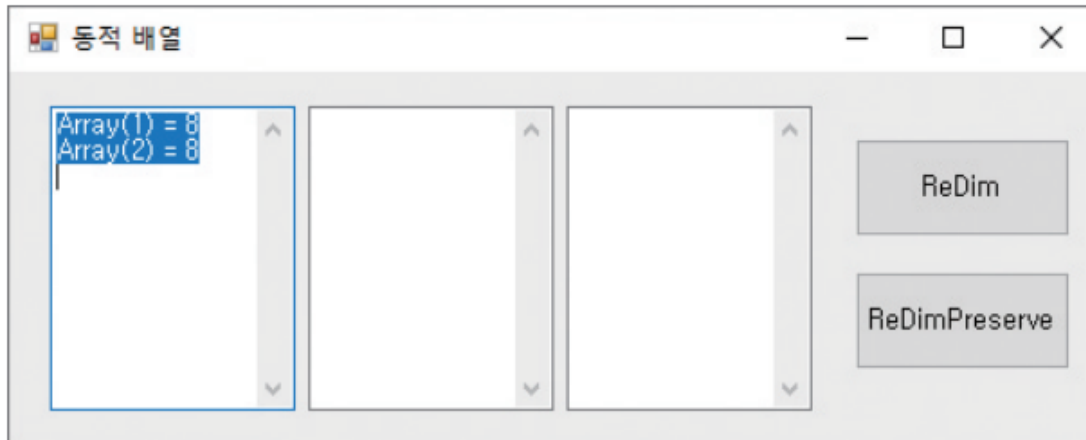
```
11 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles  
    Button1.Click  
12     ReDim DynamicArray(5)  
13     For i = 1 To UBound(DynamicArray)  
14         DynamicArray(i) = DynamicArray(i) + 10  
15         TextBox2.Text = TextBox2.Text & "Array(" & i & ") = " & DynamicArray(i)  
            & vbCrLf  
16     Next  
17 End Sub  
  
18 Private Sub Button2_Click(sender As Object, e As EventArgs) Handles  
    Button2.Click  
19     ReDim Preserve DynamicArray(10)  
20     For i = 1 To UBound(DynamicArray)  
21         DynamicArray(i) = DynamicArray(i) + 10  
22         TextBox3.Text = TextBox3.Text & "Array(" & i & ") = " & DynamicArray(i) &  
            vbCrLf  
23     Next  
24 End Sub  
25 End Class
```


2. 정적 배열, 동적 배열, ArrayList

실습하기 6-4 ReDim 문을 이용하여 동적 배열 다루기

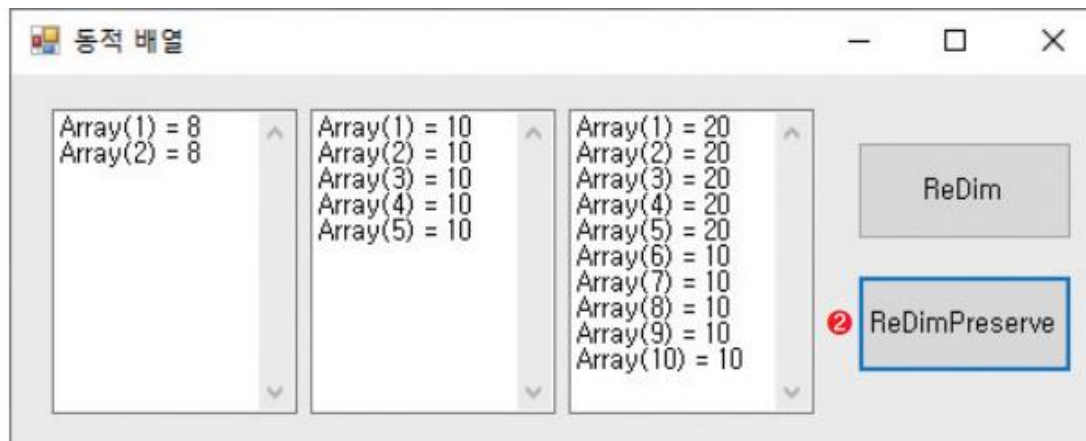
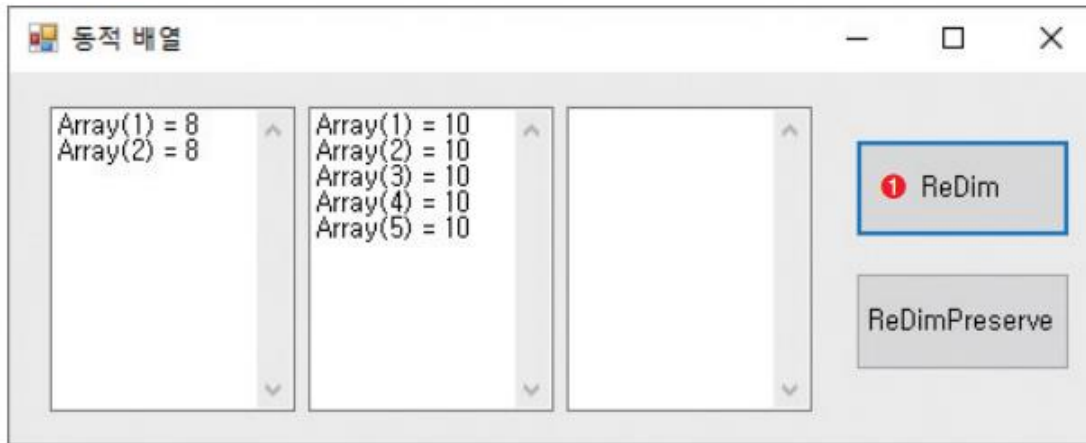
4) 실행 결과 확인

- <ReDim>을 클릭하면 기존값이 사라지고 크기가 5, 값이 10인 배열이 출력
<ReDimPreserve>를 클릭하면 앞의 배열값이 보존되어 Array(1)~Array(5)의 값이 20으로 출력



2. 정적 배열, 동적 배열, ArrayList

실습하기 6-4 ReDim 문을 이용하여 동적 배열 다루기



2. 정적 배열, 동적 배열, ArrayList

■ ArrayList

- 동적 배열처럼 프로그램 실행 시 크기를 조절할 수 있는 데이터 구조를 구현
- 사용자의 요구에 따라 동적으로 데이터를 저장하는 구조로 1차원 배열과 같이 첨자를 사용하여 데이터에 접근

형식 | `Dim ArrayList 이름 As New ArrayList()`

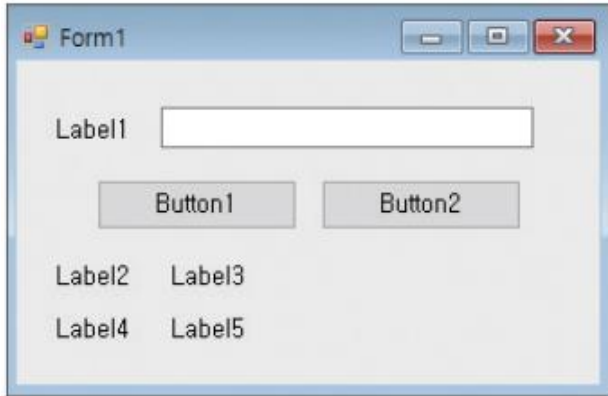
표 6-1 ArrayList의 주요 속성과 메서드

구분	이름	설명
속성	Count	ArrayList의 배열 요소 개수이다.
	Capacity	ArrayList의 배열 요소 개수를 설정한다.
	Item(Index)	지정한 Index를 첨자로 하는 요소의 값을 가져오거나 설정한다.
메서드	Add	ArrayList의 마지막에 요소를 추가한다.
	Clear	ArrayList의 모든 요소를 지운다.
	Contains	ArrayList에 요소가 있는지 여부를 True/False로 나타낸다.
	Remove	ArrayList에서 특정 요소를 지운다.
	Sort	ArrayList의 요소를 정렬한다.

2. 정적 배열, 동적 배열, ArrayList

실습하기 6-5 ArrayList를 이용하여 동적 데이터 구조 구현하기

1) 윈도우 폼 디자인



2) 속성 설정

컨트롤	속성	속성값	컨트롤	속성	속성값
TextBox1	Text	(빈칸)	Label4	Text	단어 수
Label1	Text	단어	Label5	Text	—
Label2	Text	문장	Button1	Text	단어 추가
Label3	Text	—	Button2	Text	단어 삭제

2. 정적 배열, 동적 배열, ArrayList

실습하기 6-5 ArrayList를 이용하여 동적 데이터 구조 구현하기

3) 코드 작성

```
01 Imports System.Collections

02 Public Class Form1
03     Dim words As New ArrayList()
04     Dim sentence As String = ""

05     Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.
        Click
06         words.Add(TextBox1.Text)
07         Print_Result()
08     End Sub

09     Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.
        Click
10         words.Remove(TextBox1.Text)
11         Print_Result()
12     End Sub
```

2. 정적 배열, 동적 배열, ArrayList

실습하기 6-5 ArrayList를 이용하여 동적 데이터 구조 구현하기

3) 코드 작성

```
13 Private Sub Print_Result()  
14     Dim i As Integer  
15     sentence = ""  
16     For i = 0 to words.Count-1  
17         sentence = sentence & " " & words.Item(i)  
18     Next i  
19     Label3.Text = sentence  
20     Label5.Text = words.Count  
21 End Sub  
22 End Class
```

2. 정적 배열, 동적 배열, ArrayList

실습하기 6-5 ArrayList를 이용하여 동적 데이터 구조 구현하기

여기서 잠깐

객체 지향 프로그래밍

소프트웨어 개발 방식의 한 종류로, 하드웨어를 제작할 때 미리 만들어진 부품을 조립하는 것처럼 소프트웨어를 부품화하고 필요할 때 사용하는 방식이다. 예를 들어 새 물건을 만드는 데 너트 2개가 필요하면 너트를 만드는 금형(틀)을 이용하여 제작하면 되는데 이 과정을 소프트웨어 제작에 적용한 것이다. 너트를 만드는 금형(틀)은 클래스, 만들어진 너트는 객체로 볼 수 있다. 객체 지향 프로그래밍은 데이터와 그 데이터를 사용하는 메서드(기능 단위)들을 함께 묶어 클래스라는 설계도를 만든 후, 이 클래스를 통해 객체를 만들고 만들어진 객체를 모아서 하나의 프로그램을 만드는 개념이다.

상속

클래스를 정의할 때 이미 존재하는 클래스의 특성(속성과 메서드)을 그대로 물려받아 이용할 수 있는데 이를 상속이라 한다. 상속을 이용하면 이미 존재하는 특성은 재사용하고 필요한 특성만 정의하여 사용할 수 있다.

오버라이딩

오버라이딩은 상속받은 클래스 내의 메서드를 재정의하는 것을 말한다. 상속받은 메서드의 기능이 원하는 기능과 다를 때 이름은 그대로 사용하고 기능을 변경할 수 있다.

라이브러리

.Net에서는 많이 사용되는 프로그램 단위 기능을 필요할 때마다 가져다 쓸 수 있도록 클래스로 미리 만들어두었는데 이를 라이브러리라고 한다. 라이브러리에 존재하는 클래스의 양이 많기 때문에 관련된 클래스끼리 묶어 제공한다. 최상위부터 마침표(.)로 구분하여 계층 구조로 관리한다.

네임스페이스

우리 집과 옆집에 아들이 한 명씩 있는데 우연히 이름이 둘 다 길동이일 수 있다. 그러면 '우리 집 길동이와 옆집 길동이'가 같이 체육관에 갔는데 우리 집 길동이는 농구를 하고 옆집 길동이는 배드민턴을 쳤다.'라고 말해야 듣는 사람이 구분할 수 있다. 하지만 우리 집에서는 그냥 길동이라고 해도 될 것이다. 이처럼 라이브러리에 포함된 많은 클래스에 사용되는 변수나 프로시저 등의 이름이 동일할 수 있는데, 이때 서로 구분하기 위해 등장한 개념이 네임스페이스이다. [실습하기 6-5]에서 System.Collections 네임스페이스는 System에 속한 Collections 그룹을 의미한다. 프로그램에서 System.Collections에 저장된 클래스를 사용하려면 Imports 문으로 불러와야 한다.

2. 정적 배열, 동적 배열, ArrayList

실습하기 6-5 ArrayList를 이용하여 동적 데이터 구조 구현하기

4) 실행 결과 확인

Form1

단어 ①

②

문장 -

단어 수 -

Form1

단어

문장 Hello ③

단어 수 1 ③

Form1

단어 ①

②

문장 Hello welcome to ArrayList ③

단어 수 4 ③

Form1

단어 ④

⑤

문장 Hello welcome ArrayList ⑥

단어 수 3 ⑥

3. 구조체

■ 구조체의 개념

- 데이터형이 다른 2개 이상의 변수를 한 세트로 묶어서 사용

```
Dim name(4) As String = {"홍길동", "이몽룡", "심청이"}
```

```
Dim mid(4) As Integer = {80, 90, 85}
```

```
Dim final(4) As Integer = {81, 93, 88}
```

```
Dim total(4) As Integer
```

```
Dim avg(40) As double
```

name(0)	홍길동	mid(0)	80	final(0)	81	total(0)	161	avg(0)	80.5
name(1)	이몽룡	mid(1)	90	final(1)	93	total(1)	183	avg(1)	91.5
name(2)	심청이	mid(2)	85	final(2)	88	total(2)	172	avg(2)	86

그림 6-4 배열을 이용한 학생 성적 데이터 처리

	name	mid	final	total	avg
첫 번째 데이터 →	홍길동	80	81	161	80.5
두 번째 데이터 →	이몽룡	90	93	183	91.5
세 번째 데이터 →	심청이	85	88	172	86

그림 6-5 구조체를 이용한 학생 성적 데이터 처리

3. 구조체

■ 구조체의 정의

- 구조체가 어떻게 이뤄졌는지 먼저 정의한 다음, 해당 구조를 가져다 쓰는 구조체 변수를 선언하고 초기화하여 사용
- 멤버 변수 - 구조체에 포함되는 각각의 변수

```
형식 | [Public | Private | Protected] Structure 구조체명
      {Dim | Public | Private} 멤버 변수명1 As 데이터형
      {Dim | Public | Private} 멤버 변수명2 As 데이터형
      :
      {Dim | Public | Private} 멤버 변수명n As 데이터형
End Structure
```

```
예 | Public Structure Score
    Dim name As string
    Dim mid As Integer
    Dim final As Integer
    Dim total As Integer
    Dim avg As double
End Structure
```

3. 구조체

■ 구조체 변수의 선언과 초기화

- 구조체 변수를 선언하면 메모리에 기억 공간을 확보
- 선언된 구조체 변수의 초기화는 멤버 변수별로 할당해야 하며, 선언과 동시에 초기화를 할 수도 있음

형식 | Dim 변수명 As 구조체명

예 | Dim VBscore As Score

Dim JAVAscore As Score = {"홍길동", 90, 99, 189, 94.5}

■ 구조체 멤버 변수의 참조

- 구조체 변수는 멤버 변수 단위로 참조하여 사용

VBscore.name = "홍길동"

VBscore.mid = 90

VBscore.final = 95

VBscore.total = 185

VBscore.avg = 92.5

3. 구조체

실습하기 6-6 구조체를 이용하여 성적 조회하기

1) 윈도우 폼 디자인

2) 속성 설정

컨트롤	속성	속성값
Form1	Text	성적 조회
Label1	Text	이름
Label2		중간점수
Label3		기말점수
Label4		총점
Label5		평균
TextBox1~TextBox5	Text	(빈칸)
Button1	Text	1번 학생
Button2		2번 학생
Button3		3번 학생

3. 구조체

실습하기 6-6 구조체를 이용하여 성적 조회하기

3) 코드 작성

```
01 Public Class Form1
02     Public Structure Score
03         Dim name As String
04         Dim mid As Integer
05         Dim final As Integer
06         Dim total As Integer
07         Dim avg As Double
08     End Structure

09     Dim VbScore(2) As Score

10     Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
        Button1.Click
11         VbScore(0).name = "홍길동"
12         VbScore(0).mid = 90
13         VbScore(0).final = 97
14         VbScore(0).total = VbScore(0).mid + VbScore(0).final
15         VbScore(0).avg = VbScore(0).total / 2.0
```

3. 구조체

실습하기 6-6 구조체를 이용하여 성적 조회하기

```
16     TextBox1.Text = VbScore(0).name
17     TextBox2.Text = VbScore(0).mid
18     TextBox3.Text = VbScore(0).final
19     TextBox4.Text = VbScore(0).total
20     TextBox5.Text = VbScore(0).avg
21 End Sub
22 Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
    Button2.Click
23     VbScore(1).name = "이몽룡"
24     VbScore(1).mid = 98
25     VbScore(1).final = 95
26     VbScore(1).total = VbScore(1).mid + VbScore(1).final
27     VbScore(1).avg = VbScore(1).total / 2.0

28     TextBox1.Text = VbScore(1).name
29     TextBox2.Text = VbScore(1).mid
30     TextBox3.Text = VbScore(1).final
31     TextBox4.Text = VbScore(1).total
32     TextBox5.Text = VbScore(1).avg
33 End Sub
```

3. 구조체

실습하기 6-6 구조체를 이용하여 성적 조회하기

```
34 Private Sub Button3_Click(sender As Object, e As EventArgs) Handles  
    Button3.Click  
35     VbScore(2).name = "심청이"  
36     VbScore(2).mid = 100  
37     VbScore(2).final = 97  
38     VbScore(2).total = VbScore(2).mid + VbScore(2).final  
39     VbScore(2).avg = VbScore(2).total / 2.0  
  
40     TextBox1.Text = VbScore(2).name  
41     TextBox2.Text = VbScore(2).mid  
42     TextBox3.Text = VbScore(2).final  
43     TextBox4.Text = VbScore(2).total  
44     TextBox5.Text = VbScore(2).avg  
45 End Sub  
46 End Class
```

3. 구조체

실습하기 6-6 구조체를 이용하여 성적 조회하기

4) 실행 결과 확인

성적 조회		
이름	홍길동	
중간점수	90	1번 학생
기말점수	97	2번 학생
총점	187	3번 학생
평균	93.5	

성적 조회		
이름	이몽룡	
중간점수	98	1번 학생
기말점수	95	2번 학생
총점	193	3번 학생
평균	96.5	

성적 조회		
이름	심청이	
중간점수	100	1번 학생
기말점수	97	2번 학생
총점	197	3번 학생
평균	98.5	



Thank You
