

본 강의에서 수업자료로 이용되는 저작물은  
저작권법 제25조 수업목적 저작물 이용 보상금제도에 의거,  
한국복제전송저작권협회와 약정을 체결하고 적법하게 이용하고 있습니다.  
약정범위를 초과하는 사용은 저작권법에 저촉될 수 있으므로  
수업자료의 재 복제, 대중 공개·공유 및 수업 목적 외의 사용을 금지합니다.

2024. 8. 30.

부천대학교·한국복제전송저작권협회

C#

6주차 1차시

11장 고급 컨트롤

# 6주차 학습 내용

## \* 1차시

- \* 5주차 과제 피드백(6주차 2, 3차시 수업 완료 후 피드백 점검)
- \* 11장 고급 컨트롤
  - \* 리스트 뷰
  - \* 프로그래스바
  - \* 타이머 컴포넌트

## \* 2차시

- \* 12장 그리기
  - \* 그리기 개요
  - \* 그리기 관련 자료형

## \* 3차시

- \* 12장 그리기
  - \* 도형 그리기
  - \* 문자열 그리기
  - \* 이미지 그리기

# C# 프로그래밍 입문



## 11. 고급 컨트롤



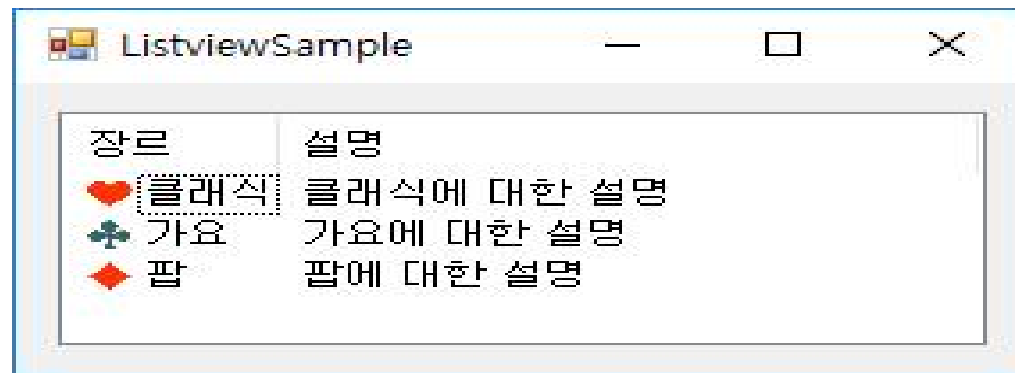
## 목차

- 리스트 뷰
- 트리 뷰
- 업다운 컨트롤
- 트랙 바
- 프로그레스 바
- 타이머 컴포넌트



## 리스트 뷰

- 리스트 상자와 유사한 형태를 지니며 목록을 구조적으로 장식할 수 있는 컨트롤
  - 리스트 상자 + 추가적인 정보 (아이콘, 설명)

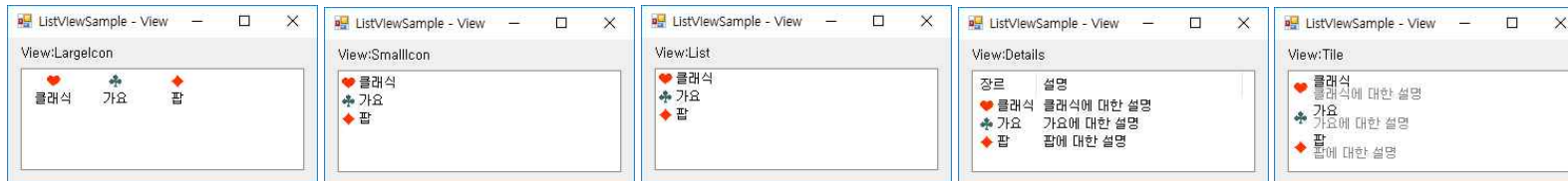




## 리스트 뷰의 형태

- View 프로퍼티의 값에 따라 다양한 형태를 가짐
  - System.Windows.Forms 네임스페이스에 포함된 View 열거형을 값으로 가짐
  - View 열거형

기호상수	설명
LargeIcon	큰 아이콘의 형태 (1)
SmallIcon	작은 아이콘의 형태 (2)
List	간단한 리스트 형태 (3)
Detail	자세한 리스트 형태 (4)
Tile	큰 아이콘이 표시되는 자세한 리스트 형태 (5)





## 리스트 뷰 항목의 선택 [1/6]

- SelectedItems 프로퍼티
  - 리스트 뷰에서 선택된 항목을 저장하는 프로퍼티
  - 반환형
    - ListViewItem 클래스형
      - 리스트 뷰의 MultiSelect 프로퍼티가 거짓일 경우
    - ListViewItem 클래스의 배열형
      - 리스트 뷰의 MultiSelect 프로퍼티가 참일 경우

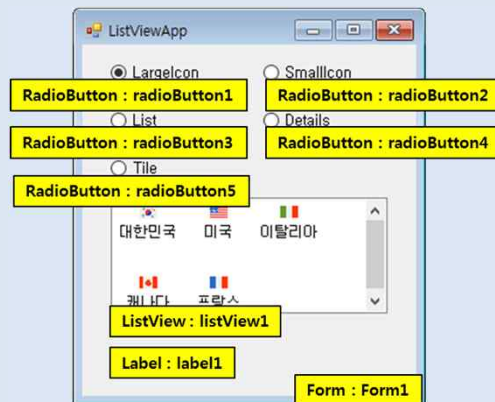




## 리스트 뷰 항목의 선택 [2/6]

예제 11.1 [w6\_ex11\_1\_xxxx] : 리스트 뷰

1) 디자인



컴포넌트 : (Name)	프로퍼티	인덱스	값
ImageList : imageList1	Images	0	South Korea.png
		1	USA.png
		2	Italy.png
		3	Canada.png
		4	France.png

이미지 경로 : <https://icons8.com/web-app/category/all/Flags>



## 리스트 뷰 항목의 선택 [3/6]

컨트롤 : (Name)	프로퍼티	값
Form : Form1	Text	ListViewApp
RadioButton : radiobutton1	Text	LargeIcon
	Checked	True
RadioButton : radioButton2	Text	SmallIcon
RadioButton : radioButton3	Text	List
RadioButton : radioButton4	Text	Details
RadioButton : radioButton5	Text	Details
Label : label1	Text	
ListView : listView1	Columns	columnHeader1
		columnHeader2
	LargeImageList	imageList1
	SmallImageList	imageList1
ColumnHeader : columnHeader1	Text	국가
ColumnHeader : columnHeader2	Text	국가번호

ListViewApp

☐ LargeIcon ☐ SmallIcon

☐ List ☒ Details

☐ Tile

국가	국가번호
대한민국	82
미국	1
이탈리아	39
캐나다	1

columnHeadercollection => Columns  
columnHeader  
ListViewItem  
ListViewsubItemCollection  
ListViewsubItem

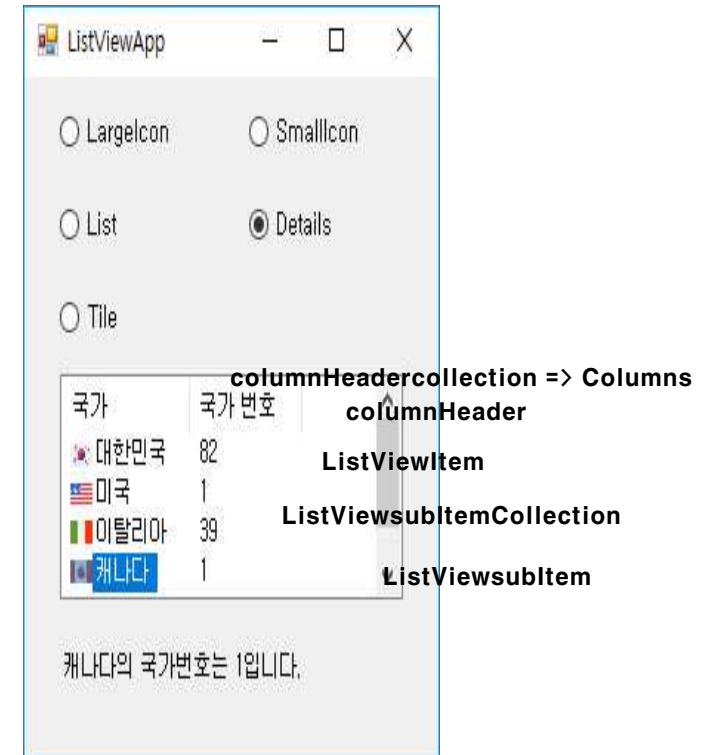
캐나다의 국가번호는 1입니다.



## 리스트 뷰 항목의 선택 [4/6]

listView1의 Items 프로퍼티

Items	프로퍼티	값	프로퍼티	값
ListViewItem0	ImageIndex	0		
	SubItems	ListViewSubItem0	Text	대한민국
		ListViewSubItem1	Text	82
ListViewItem1	ImageIndex	1		
	SubItems	ListViewSubItem0	Text	미국
		ListViewSubItem1	Text	1
ListViewItem2	ImageIndex	2		
	SubItems	ListViewSubItem0	Text	이탈리아
		ListViewSubItem1	Text	39
ListViewItem3	ImageIndex	3		
	SubItems	ListViewSubItem0	Text	캐나다
		ListViewSubItem1	Text	1
ListViewItem4	ImageIndex	4		
	SubItems	ListViewSubItem0	Text	프랑스
		ListViewSubItem1	Text	33



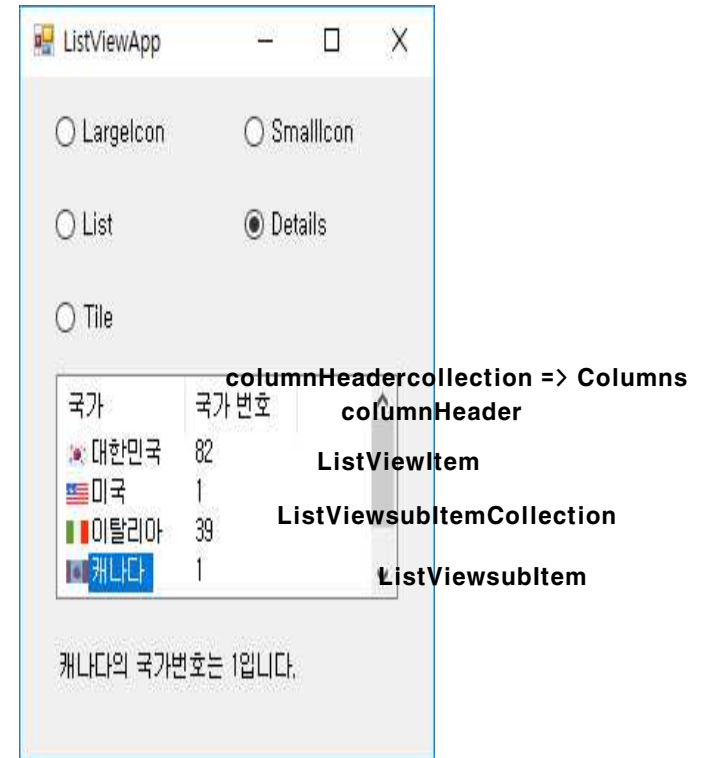


## 리스트 뷰 항목의 선택 [5/6]

컨트롤 : (Name)	이벤트	메소드명
RadioButton : radioButton1	CheckedChanged	radioButton1_CheckedChanged
RadioButton : radioButton2	CheckedChanged	radioButton2_CheckedChanged
RadioButton : radioButton3	CheckedChanged	radioButton3_CheckedChanged
RadioButton : radioButton4	CheckedChanged	radioButton4_CheckedChanged
RadioButton : radioButton5	CheckedChanged	radioButton5_CheckedChanged
ListView : listView1	Click	listView1_Click()

### 2) 코드

```
private void radioButton1_CheckedChanged(object sender, EventArgs e) {  
    if (radioButton1.Checked)  
        // 리스트 뷰의 항목을 큰 아이콘 형태로 보여준다.  
        listView1.View = View.LargeIcon;  
}  
private void radioButton2_CheckedChanged(object sender, EventArgs e) {  
    if (radioButton2.Checked)  
        // 리스트 뷰의 항목을 작은 아이콘 형태로 보여준다.  
        listView1.View = View.SmallIcon;  
}  
private void radioButton3_CheckedChanged(object sender, EventArgs e) {  
    if (radioButton3.Checked)  
        // 리스트 뷰의 항목을 간단한 리스트 형태로 보여준다.  
        listView1.View = View.List;  
}
```



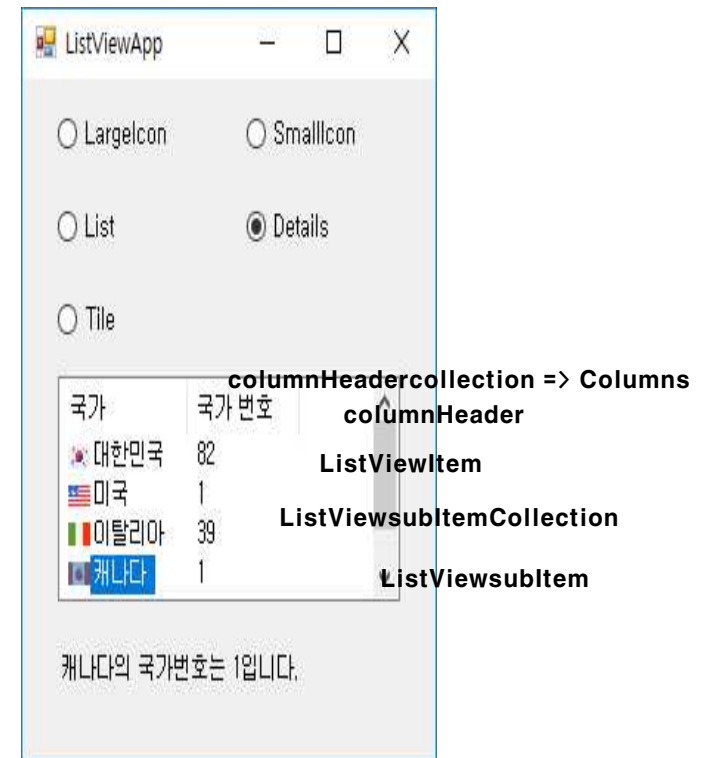


## 리스트 뷰 항목의 선택 [6/6]

```
private void radioButton4_CheckedChanged(object sender, EventArgs e) {
    if (radioButton4.Checked)
        // 리스트 뷰의 항목을 자세한 리스트 형태로 보여준다.
        listView1.View = View.Details;
}
private void radioButton5_CheckedChanged(object sender, EventArgs e) {
    if (radioButton5.Checked)
        // 리스트 뷰의 항목을 타일 형태로 보여준다.
        listView1.View = View.Tile;
}
private void listView1_Click(object sender, EventArgs e) {
    foreach (ListViewItem item in listView1.SelectedItems) {
        ListViewItem.ListViewSubItemCollection subItem = item.SubItems;
        // 각 항목에 대한 부항목을 얻기 위해 SubItems 프로퍼티를 사용
        label1.Text = subItem[0].Text + "의 국가번호는 " + subItem[1].Text + "입니다.";
    }
}
```

실행 방법 : ① 라디오 버튼 중 하나를 선택한다.  
② 리스트 뷰의 항목을 선택한다.

실행 결과 :

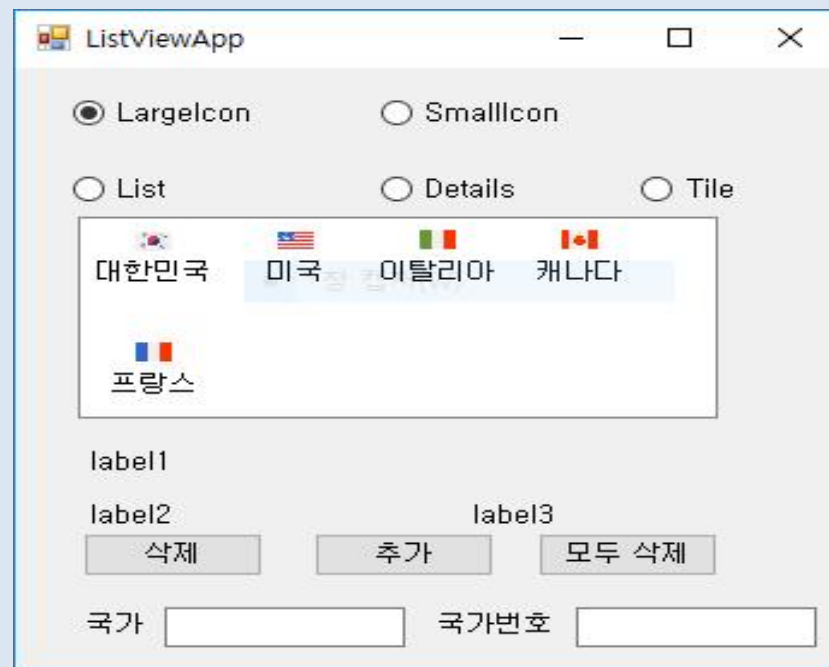




# 리스트 뷰 항목 추가, 삭제(퀴즈 11.1)

quiz 11.1 [w6\_quiz11\_1\_xxxx] : 리스트 뷰 퀴즈

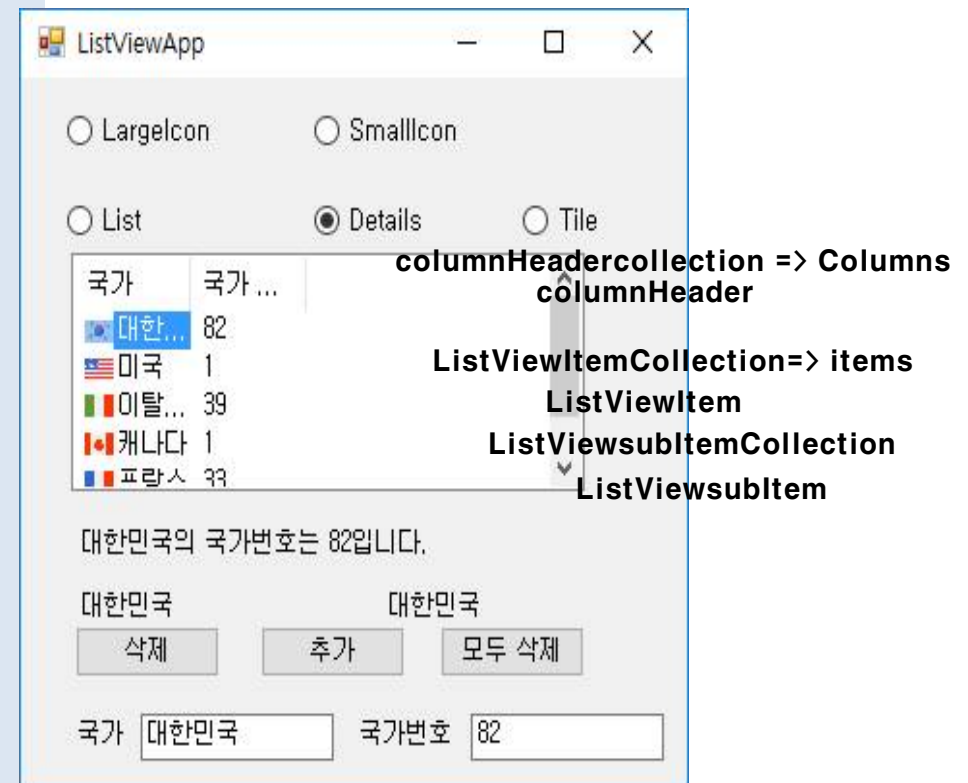
1) 디자인





## 리스트 뷰 항목의 추가, 삭제(퀴즈 11.1)

```
private void radioButton4_CheckedChanged(object sender, EventArgs e) {
    if (radioButton4.Checked)
        // 리스트 뷰의 항목을 자세한 리스트 형태로 보여준다.
        listView1.View = View.Details;
}
private void radioButton5_CheckedChanged(object sender, EventArgs e) {
    if (radioButton5.Checked)
        // 리스트 뷰의 항목을 타일 형태로 보여준다.
        listView1.View = View.Tile;
}
private void listView1_Click(object sender, EventArgs e) {
    foreach (ListViewItem item in listView1.SelectedItems) {
        ListViewItem.ListViewSubItemCollection subItem = item.SubItems;
        // 각 항목에 대한 부항목을 얻기 위해 SubItems 프로퍼티를 사용
        label1.Text = subItem[0].Text + "의 국가번호는 " + subItem[1].Text + "입니다.";
    }
}
```





## 리스트 뷰 항목 추가, 삭제(퀴즈 11.1)

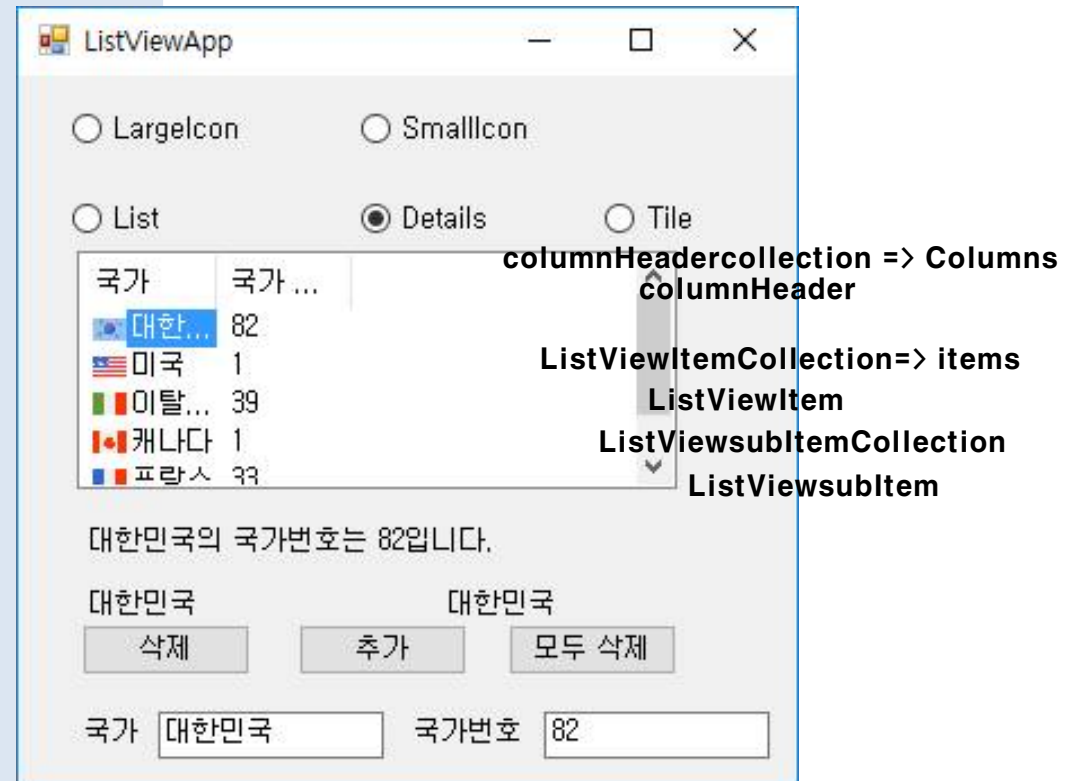
quiz 11.1 [w6\_quiz11\_1\_xxxx] : 리스트 뷰 퀴즈

=> 추가된 내용

```
private void button1_Click(object sender, EventArgs e)
{
    if (listView1.Count <= 0)
        MessageBox.Show("선택된 항목이 없음!");
    else
        while (listView1.Count > 0)
            listView1.Items.Remove(listView1.SelectedItems[0]);
}

private void button2_Click(object sender, EventArgs e)
{
    int count ;
    ListViewItem lsvitem ;
    if (listView1.Items.Count == 0 )
        count = 0;
    else
        count = listView1.Items.Count;

    lsvitem = new ListViewItem(textBox1.Text, count);
    listView1.Items.Add(lsvitem);
    listView1.Items[count].Text = textBox2.Text;
}
```







# 리스트 뷰 항목 추가, 삭제(퀴즈 11.1)

quiz 11.1 [w6\_quiz11\_1\_xxxx] : 리스트 뷰 퀴즈

3) 실행 화면

The screenshot shows the 'ListViewApp' window with the 'Details' view selected. The list displays country information with columns for country name and country number. Annotations on the right side of the window identify the following components:

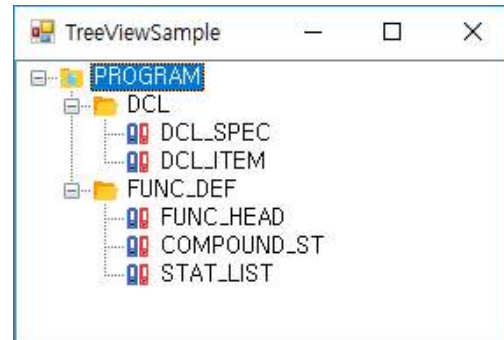
- columnHeadercollection => Columns**: Points to the header row containing '국가' and '국가 ...'.
- columnHeader**: Points to the '국가 ...' header cell.
- ListViewItemCollection=> items**: Points to the list of data items.
- ListViewItem**: Points to a single data row, such as '대한민국 82'.
- ListViewSubItemCollection**: Points to the collection of sub-items within a row.
- ListViewSubItem**: Points to a specific sub-item, such as the country number '82'.

Below the list, the text '대한민국의 국가번호는 82입니다.' is displayed. At the bottom, there are input fields for '국가' (set to '대한민국') and '국가번호' (set to '82'), along with buttons for '삭제', '추가', and '모두 삭제'.



## 트리 뷰

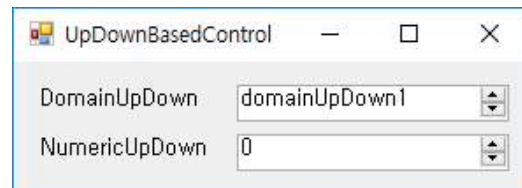
- 목록을 계층적으로 보여주기 위한 컨트롤
  - 노드를 계층적으로 표시
  - 노드에 이미지 아이콘을 추가할 수 있음





## 업다운 컨트롤

- 주어진 목록에서 항목을 선택할 수 있는 컨트롤
  - 업다운 버튼을 이용하여 필요한 값을 선택
  - 스펀 컨트롤(spin control)
- 영역 업다운 컨트롤
  - 문자열로 이루어진 항목에서 특정한 항목을 선택할 수 있는 컨트롤
- 수치적 업다운 컨트롤
  - 지정한 범위 내에서 수치적 값을 선택할 수 있는 컨트롤





## 수치적 업다운 컨트롤의 작성 [1/3]

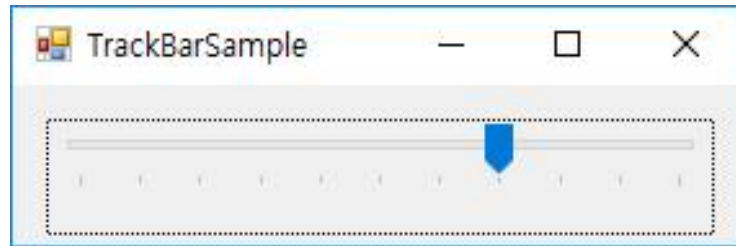
- 수치적 업다운 컨트롤의 추가
  - 【도구상자】➡【NumericUpDown】을 선택하여 폼에 추가
- 수치적 업다운 컨트롤의 항목에 대한 범위와 증가/감소량을 설정
  - 수치적 업다운 컨트롤의 프로퍼티를 통해 설정

프로퍼티	설 명
Minimum	수치적 업다운 컨트롤의 최소 값.
Maximum	수치적 업다운 컨트롤의 최대 값.
Increment	수치적 업다운 컨트롤의 증가/감소 양.
Value	수치적 업다운 컨트롤의 현재 값.
DecimalPlaces	수치적 업다운 컨트롤에 표시할 소수 자릿수.
ThousandsSeparator	10진수 3자리마다 구분 기호를 삽입 여부.
Hexadecimal	수치적 업다운 컨트롤의 값을 16진수로 표시.



## 트랙 바

- 범위 내에서 값을 선택할 수 있는 컨트롤
  - 슬라이더와 눈금으로 구성



- 슬라이더의 이동
  - 마우스 드래그
  - 슬라이더의 좌우 공간 클릭
  - 마우스 휠의 회전
  - 키보드의 좌우 방향키, 페이지 업다운키



## 트랙 바의 작성 [1/4]

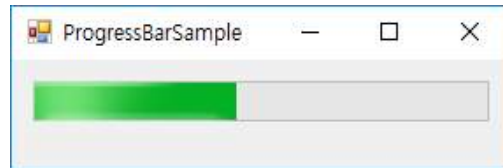
- 트랙 바의 추가
  - 【도구상자】➡【TrackBar】를 선택하여 폼에 추가
- 트랙 바의 값에 대한 범위와 이동량을 설정
  - 트랙바의 프로퍼티를 통해 설정

프로퍼티	설 명
Minimum	트랙 바의 최소 값
Maximum	트랙 바의 최대 값
Value	트랙 바의 현재 값
LargeChange	마우스 클릭이나 PageUp/PageDown 키에 대한 이동량
SmallChange	마우스 휠의 회전이나 키보드의 방향키에 대한 이동량
TickFrequency	눈금이 표시되는 값의 범위
TickStyle	트랙 바에 눈금이 표시되는 위치
Orientation	트랙 바의 방향(Horizontal   Vertical)



## 프로그레스 바

- 작업의 진행상황을 보여주는 컨트롤
  - 좌측에서 우측으로 사각형의 조각을 채우면서 진행
  - 애플리케이션의 설치과정이나 파일 복사과정에서 사용





## 프로그레스 바의 작성 [1/2]

- 프로그레스 바의 추가
  - 【도구상자】➡【ProgressBar】를 선택하여 폼에 추가
- 프로그레스 바의 값에 대한 범위를 설정
  - 프로그레스 바의 프로퍼티를 통해 설정
    - Maximum
      - 프로그레스 바의 최대값
    - Minimum
      - 프로그레스 바의 최소값

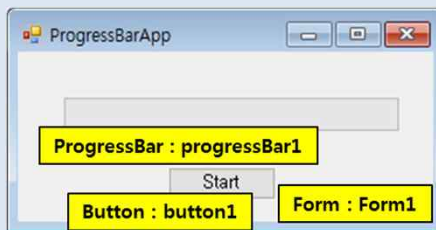




## 프로그레스 바의 작성 [2/2]

예제 11.7 [w6\_ex11\_7\_xxxx] : 프로그레스 바

### 1) 디자인



컨트롤 : (Name)	프로퍼티	값
Form : Form1	Text	ProgressBarApp
Button : button1	Text	Start
ProgressBar : progressBar1	Minimum	0
	Maximum	100000

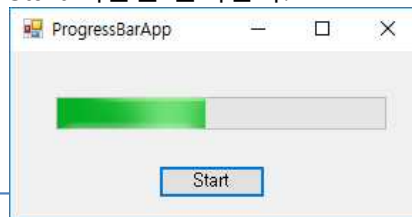
컨트롤 : (Name)	이벤트	메소드명
Button : button1	Click	button1_Click()

### 2) 코드

```
private void button1_Click(object sender, EventArgs e) {  
    for (int i = 0; i < 100000; i++)  
        progressBar1.Value += i;  
}
```

실행 방법 : Start 버튼을 클릭한다.

실행 결과 :





## 타이머 [1/3]

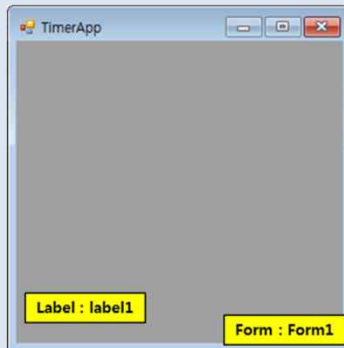
- 주기적인 간격으로 이벤트를 발생시키는 컴포넌트
  - 배경작업을 처리할 때 주로 사용
  - 일정한 간격에 따라 **Tick 이벤트**를 발생
    - **Interval 프로퍼티**를 통해 간격을 설정
    - **밀리 초**(millisecond, 1/1000초)를 사용
  - 주기적으로 발생시키기 위해서는 **Enable 프로퍼티**를 참으로 설정
  - 항상 Interval 프로퍼티의 간격에 따라 Tick 이벤트가 발생하는 것은 아님
    - Tick 이벤트가 다른 이벤트에 비해 우선순위가 낮기 때문
  - 타이머 컴포넌트의 추가
    - **【도구상자】➡【Timer】**를 선택하여 폼에 추가



## 타이머 [2/3]

예제 11.8 [w6\_ex11\_8\_xxxx] : 타이머

### 1) 디자인



컴포넌트 : (Name)	프로퍼티	값
Timer : timer1	Enable	True
	Interval	100
ImageList : imageList1	Images	frame-1.png frame-2.png frame-3.png frame-4.png frame-5.png frame-6.png frame-7.png frame-8.png

\* 이미지 경로:

<http://opengameart.org/content/game-character-blue-flappy-bird-sprite-sheets>

컨트롤 : (Name)	프로퍼티	값
Form : Form1	Text	TimerApp
Label : label1	Text	
	Dock	Fill
	BackColor	ButtonShadow
컴포넌트 : (Name)	이벤트	메소드명
Timer : timer1	Tick	timer1_Tick()



## 타이머 [3/3]

2) 코드

```
private int index = 0;  
private void timer1_Tick(object sender, EventArgs e) {  
    index %= imageList1.Images.Count;  
    label1.Image = imageList1.Images[index++];  
}
```

실행 결과 :

