

StringGrid^{스트링그리드} 콘트롤

데이터를 표 형식으로 표현하고 계산할 수 있게 됩니다.

StringGrid^{스트링그리드}는 *String*^{스트링, 문자열} 데이터를 모아 *Grid*^{그리드, 격자} 형태로 표현한다(한글로는 ‘스트링그리드’ 또는 그냥 ‘그리드’라고 하겠다).

*StringGrid*는 툴 팔레트의 *Additional* 그룹 안에 들어있다. *StringGrid* 컴포넌트의 이름을 지을 때 *sgd* 를 접두사로 붙인다. 우리는 *sgdMy* 라는 이름을 쓰기로 한다.

*StringGrid*에는 고정 셀이 있다. 대체로, 가장 왼쪽 열과 가장 위의 행에 있는 셀이다. 이것은 일반 셀의 헤더가 되고, 실행 시 사용자가 마우스로 고정 셀의 크기를 조정하면, 그리드 안에 있는 일반 셀의 높이와 너비를 변경할 수 있다.

고정 열과 고정 행의 개수를 직접 지정하려면, *FixedCols*와 *FixedRows* 프로퍼티를 사용한다. 만약 모두 0으로 지정되면, 이 그리드에는 고정 셀이 없게 된다. 일반적인 경우라면,

```
sgdMy.FixedCols:= 1; //고정 열의 개수 = 1
sgdMy.FixedRows:= 1; //고정 행의 개수 = 1
```

그리드에서 고정 셀이 아닌 열과 행에는 줄을 얼마든지 넣을 수 있고, 줄의 개수는 프로그램에서 변경할 수도 있다. 만약 그리드가 표시되는 화면보다 그리드의 내용이 더 많으면

자동으로 알맞은 스크롤이 생긴다. 그리드에서 스크롤을 사용하면 고정 행과 고정 열은 스크롤에 관계없이 항상 표시되지만, 내용은 스크롤되어 넘어가게 된다.

`Cells`^[셀스]는 `StringGrid`에서 중심이 되는 프로퍼티이다. `Cells` 프로퍼티는 `Cell`^[셀]들의 모음이고 각 `Cell`들에는 `Text`^[텍스트]가 들어갈 수 있다. `Cells`에는 좌표 2개가 있다. — `Cell`이 있는 열의 번호와 행의 번호이다. 첫 번째 행은 0번 행이고 첫 번째 열은 0번 열이다.

`Cells`는 `String` 타입을 가진다.

실행 중에 손으로 `Cells` 안에 값을 넣을 수 있다. 이렇게 하려면 오브젝트 인스펙터에서 `Options` 프로퍼티의 + 표를 눌러서 펼치고 `goEditing` 프로퍼티를 `True` 로 설정한다.

`Cell`의 값은 프로그램에서 할당 연산자를 사용하여 넣을 수도 있다. `Cell`에 들어가려면 인덱스번호를 사용한다. 첫 번째가 열 번호이고, 두 번째가 행 번호인 점을 기억하자.

예문

```
sgdMy.Cells[1,1] := '고정 셀이 아닌 것 중 가장 왼쪽 위의 셀';
sgdMy.Cells[0,0] := '번호: ';
```

‘번호’라는 문자열이 그리드의 가장 첫 번째 셀 즉, 고정 영역의 첫 번째 셀에 표시된다.

`ColCount`와 `RowCount` 프로퍼티의 값은 그리드의 규모를 정한다.

`ColCount` 와 `RowCount` 는 프로그램이 시작되거나 실행되는 도중에 변경될 수 있다. 하지만, `FixedCols` 와 `FixedRows` 의 값보다는 최소한 하나 이상 더 커야 한다.

`sgdMy` 그리드에서 열의 개수는 3으로, 행의 개수는 5로 지정해보자.

```
sgdMy.ColCount := 3;
sgdMy.RowCount := 5;
```

`FixedColor` 프로퍼티는 고정 셀들의 색상을 지정하고 `Color` 프로퍼티는 나머지 일반 셀들의 색상을 지정한다.

예제 프로그램을 작성해보자. 폼에는 행, 열, 고정 행, 고정 열의 개수를 받을 텍스트박스 4개와 버튼 3개, 그리드가 1개가 있다. 첫 번째 버튼을 클릭하면 텍스트박스에 입력된 설정값대로 *StringGrid*가 표시된다. 두 번째 버튼을 클릭하면 고정 셀들이 녹색으로 칠해진다. 세 번째 버튼을 클릭하면 나머지 일반 셀들이 빨강으로 칠해진다. (주의, 델파이가 오래된 버전이 아니라면, *StringGrid*의 *DrawingStyle* 프로퍼티가 *gdsClassic*로 설정되어 있어야 색상 변경이 적용될 수 있다)

```

procedure GetGridParam(var n1, n2, n3, n4: Integer);
begin
    n1:= StrToInt(frmGrid.edtLine.Text);
    n2:= StrToInt(frmGrid.edtStolb.Text);
    n3:= StrToInt(frmGrid.edtFline.Text);
    n4:= StrToInt(frmGrid.edtFstolb.Text);
end;
procedure CreateGrid(n1, n2, n3, n4: Integer);
begin
    frmGrid.sgdMy.RowCount:= n2;
    frmGrid.sgdMy.ColCount:= n1;
    frmGrid.sgdMy.FixedCols:= n4;
    frmGrid.sgdMy.FixedRows:= n3;
end;
procedure TfrmGrid.btnTablClick(Sender: TObject);
var
    n1, ns, nfl, nfs: Integer;
begin
    GetGridParam(n1, ns, nfl, nfs); //그리드를 만들기 위한 파라미터를 받는다
    CreateGrid(n1, ns, nfl, nfs); //정해진 파라미터에 맞게 그리드를 만든다
end;
procedure TfrmGrid.btnCellRedClick(Sender: TObject);
begin
    frmGrid.sgdMy.Color:= clRed;
end;
procedure TfrmGrid.btnFCGreenClick(Sender: TObject);
begin
    frmGrid.sgdMy.FixedColor:= clGreen;
end;

```

유용한 StringGrid 프로퍼티 몇가지

프로퍼티	설명
BorderStyle: TBorderStyle;	그리드 테두리의 모습을 명시한다: bsNone — 테두리 없음; bsSingle — 테두리 두께 1 pixel.
ColCount: LongInt;	그리드의 열 개수를 넣는다
DefaultColWidth: Integer;	그리드의 열 너비 기본값을 넣는다
RowCount: LongInt;	그리드의 행 개수를 넣는다
DefaultRowHeight: Integer;	그리드의 행 높이 기본값을 넣는다.
Color: TColor;	셀들의 색상을 조정한다.
FixedCols: Integer;	고정열의 개수를 조정한다.
FixedRows: Integer;	고정행의 개수를 조정한다.
FixedColor: TColor;	고정 셀의 색상을 조정한다.
GridHeight: Integer;	(읽기 전용) 그리드 높이를 가지고 있다.
GridWidth: Integer;	(읽기 전용) 그리드 너비를 가지고 있다.
GridLineWidth:	그리드 줄의 두께를 명시한다.

실습

Exercise 1.

StringGrid 하나를 만든다.

텍스트박스에서 다음 값을 받는다:

- 행의 개수
- 열의 개수
- 고정행의 개수
- 고정열의 개수

버튼 3개를 만들고 고정 셀의 색상을 (버튼 마다 다른 색이 반영되도록) 바꿔보자.

버튼 3개를 만들고 일반 셀의 색상을 (버튼 마다 다른 색이 반영되도록) 바꿔보자.

텍스트박스 하나와 버튼 2개를 만든다. 그리드의 셀들을 무작위 숫자들로 채운다.

첫 번째 버튼을 클릭하면, 텍스트박스에 있는 번호를 가진 열의 내용을 메모에 표시한다.

두 번째 버튼을 클릭하면, 텍스트박스에 있는 번호를 가진 행을 메모에 표시한다.

Exercise 2.

StringGrid 하나를 만든다.

학생들이 물리학 실험실에서 실험을 하고 있다. 학생들은 아래와 같이 실험 횟수, 실험의 가중치, 그리고 측정된 값들을 얻는다.

실험 횟수와 가중치를 텍스트박스에 넣는다.

측정된 값들은 키보드로 그리드 안에 입력한다. 실험 결과를 모두 그리드에 넣고 나면, 프로그램에서 결과값을 계산하고 최대값과 최소값을 찾아내보자.

예를 들어, 실험 횟수가 5회이고, 가중치가 10이라면 표는 다음과 같다.

실험 횟수	가중치	측정된 값	결과값	최대값/최소값
1	10	83	830	최대값
2	10	51	510	
3	10	67	670	
4	10	49	490	최소값
5	10	75	750	