

# 2D Arrays

## Consider this array :

```
1 2 3 4 5 6
4 4 7 8 7 5
9 8 7 4 6 2
3 2 4 8 0 4
```

This array has:

- 4 rows and 6 columns

## Declaration

```
var
  arr: array[1..4, 1..6] of Integer = ((1, 2, 3, 4, 5, 6),
                                         (4, 4, 7, 8, 7, 5),
                                         (9, 8, 7, 4, 6, 2),
                                         (3, 2, 4, 8, 0, 4));
```

## Read the array left to right

```
procedure TForm1.Button1Click(Sender: TObject);
var row, col : integer;
begin
  for row := 1 to 4 do
  begin
    for col := 1 to 6 do
    begin
      redOutput.Lines.Add(inttostr(arr[row,col]));
    end;
  end;
end;
```

**Output:** 1,2, 3, 4, 5, 6, 4, 4

### **Read the array top to bottom**

```
procedure TForm1.Button1Click(Sender: TObject);
var row, col : integer;
begin
  for col := 1 to 6 do
  begin
    for row := 1 to 4 do
    begin
      redOutput.Lines.Add(inttostr(arr[row,col]));
    end;
  end;
end;
```

**Output:** 1, 4, 9, 3, 2, 4, 8, 2, 3

## **Rules**

First lets look at the basic structure of the 2D array again.

### **Left to right**

```
var
  row, col: integer;
begin
  for row:=1 to 10 do
  begin
    for col:=1 to 15 do
    begin
      end;
    end;
  end;
end;
```

1. All code is below a BEGIN STATEMENT ie: nothing must be above any BEGIN STATEMENT when writing a 2d array basic structure.

```
var
    row, col: integer;
begin
    for row:=1 to 10 do
        // nothing ever here
        begin
            for col:=1 to 15 do
                // nonthing ever here
                begin
                    end;
                end;
            end;
        end;
    end;
```

2. Where to put stuff

- a. Initialize or reset any variables or states for the new row here.
- b. Here you can access the element at (row, col) of the 2D array.  
    // For example: array2D[row][col] or however your 2D array is structured in Delphi.
- c. Here you might want to perform any final operations for the current row, like printing a newline or doing some row-specific computation.

```
var
    row, col: integer;
begin
    for row:=1 to 10 do
        // a. what happens at the start of every row
        begin
            for col:=1 to 15 do
                begin
                    // b. accessing every element of the 2D array
                end; // COLS END
            end;
            // c. what happens at the end of every row
        end; // ROWS END
    end;
```

# Exam Question

Given these two arrays:

```
arrNames: array [1 .. 10] of String = (  
  'Ruth',  
  'Nicole',  
  'Loyiso',  
  'Chris',  
  'William',  
  'Thabo',  
  'Vusi',  
  'Peter',  
  'Jenny',  
  'Tommy'  
);  
  
arrVending: array [1 .. 10, 1 .. 15] of String = (('C', "", "", "", "",  
  "", "", "", "", "", "", "", "", "", ('B', 'B', 'B', 'C', 'C', 'C',  
  'B', 'B', 'B', 'C', 'C', 'C', 'C', 'C', 'C',  
  ("", "", "", "", "", "", "", "", "", "", "", "", "", "",  
  ('C', 'C', "", "", "", "", "", "", "", "", "", "",  
  ('B', 'B', 'C', 'C', 'B', 'B', 'C', 'C', 'C', 'C', 'B', 'C', 'C', 'B',  
  ""), ('C', 'C', 'B', "", "", "", "", "", "", "", "", "",  
  ('C', 'B', "", "", "", "", "", "", "", "", "", "",  
  ('B', 'B', "", "", "", "", "", "", "", "", "", "",  
  ('C', "", "", "", "", "", "", "", "", "", "",  
  ('B', 'C', "", "", "", "", "", "", "", "", "", ""));
```

**Produce the output:**

-----	
Names	Items recycled
-----	
Ruth	C
Nicole	BBBCCCBBBCCCCC
Loyiso	
Chris	CC
William	BBCCBBCCCCBCCB
Thabo	CCB
Vusi	CB
Peter	BB
Jenny	C
Tommy	BC

## Answer

```
procedure TfrmQuestion4.btnQ4_1Click(Sender: TObject);
var
  col, row, i: integer;
  sName, sString: string;
begin
  // Provided code
  redQ4.Clear;
  redQ4.Lines.Add('-----');
  redQ4.Lines.Add('Names' + #9 + 'Items recycled');
  redQ4.Lines.Add('-----');

  for row := 1 to 10 do
  begin
    sName := arrNames[row];
    sString := '';
    for col := 1 to 15 do
    begin
      sString := sString + arrVending[row,col];
    end;
    redQ4.Lines.Add(sName + #9 + sString);
  end;
end;
```

### Review Questions

The lines:

- sName:= arrNames[row];
- sString := '';

are specifically above the for loop for the column. *Can you explain why ?*

**Secondly**, the line:

- redQ4.Lines.Add(sName + #9 + sString)

is after the for loop for the column. *Is there a reason for this ?*

**Thirdly**, why are the two for loops arranged in the order of for row:= 1 .. and then for col:= 1 and not the other way around ?