2D Arrays

ROW
HEIGHT

myMatrix [n][m]

COLUMN
WIDTH
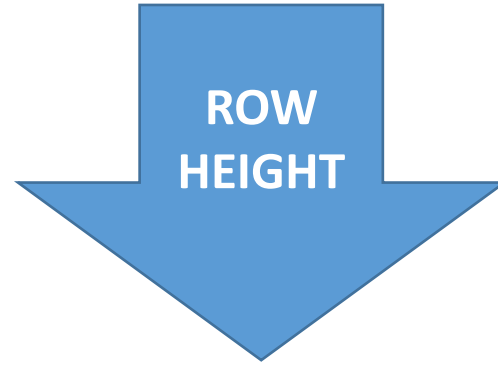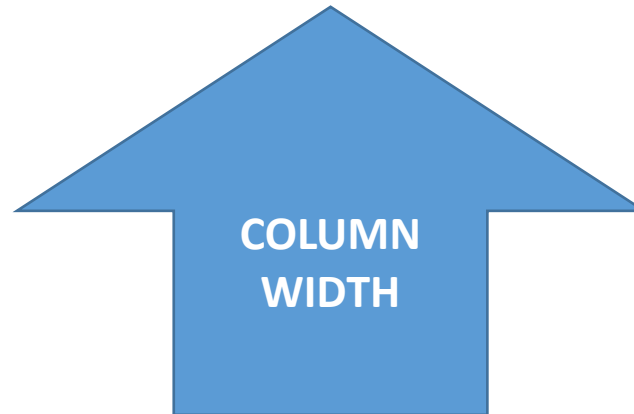
8.7 Focus on Languages: C++

- Two-Dimensional Arrays in C++
  - Here is a declaration of a two-dimensional array with three rows and four columns:

    ```
    double scores[3][4];
    ```

  - The elements in row 0 are referenced as:

    ```
    scores[0][0]
    scores[0][1]
    scores[0][2]
    scores[0][3]
    ```

  - The elements in row 1 are referenced as:

    ```
    scores[1][0]
    scores[1][1]
    scores[1][2]
    scores[1][3]
    ```

  - And so on…

Exercise 1

Develop C++ program to built the following 2-dimensional array

	0	1	2	3	4
0	1	2	3	4	5
1	2	4	6	8	10
2	3	6	9	12	15

jimmy

Exercise 1

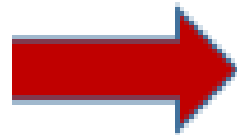|	0	1	2	3	4
0	1	2	3	4	5
1	2	4	6	8	10
2	3	6	9	12	15

jimmy

The formula

```
jimmy[n][m]=(n+1)*(m+1);
```

```cpp
#include <iostream>
#include <iomanip> // This for setw()
using namespace std;

#define WIDTH 5
#define HEIGHT 3

int jimmy [HEIGHT][WIDTH]; // Global variable
int n,m, xin; // Global variable

int main ()
{ cout<<" === Developed by ---- ==="<<endl;
    // To build the matix
    for (n=0;n<HEIGHT; n++)
      for (m=0;m<WIDTH; m++)
      {
          jimmy[n][m]=(n+1)*(m+1);
      }
```

```
for (n=0;n<HEIGHT; n++)
    {
        for (m=0;m<WIDTH; m++)
        {
            cout<<jimmy[n][m]<<" ";
        }
    cout<<endl;

    }
```

SETW() Sets the number of characters to be used as the *field width* for the next insertion operation.
#include <iomanip> *must be there to use* ***setw()***

```
for (n=0;n<HEIGHT; n++)
{
    for (m=0;m<WIDTH; m++)
    {
        cout<<setw (10)<<jimmy[n][m]<<"";
    }
    cout<<endl;
}
```

```cpp
int main ()
{ cout<<" === Developed by ---- ==="<<endl;
    // To build the matix
    for (n=0;n<HEIGHT; n++)
        for (m=0;m<WIDTH; m++)
        {
            jimmy[n][m]=(n+1)*(m+1);
        }


    // To display the matix

    cout<<"  Original Matrix "<<endl;
    for (n=0;n<HEIGHT; n++)
    {
        for (m=0;m<WIDTH; m++)
        {
            cout<<setw (10)<<jimmy[n][m]<<"";
        }
        cout<<endl;
}
```
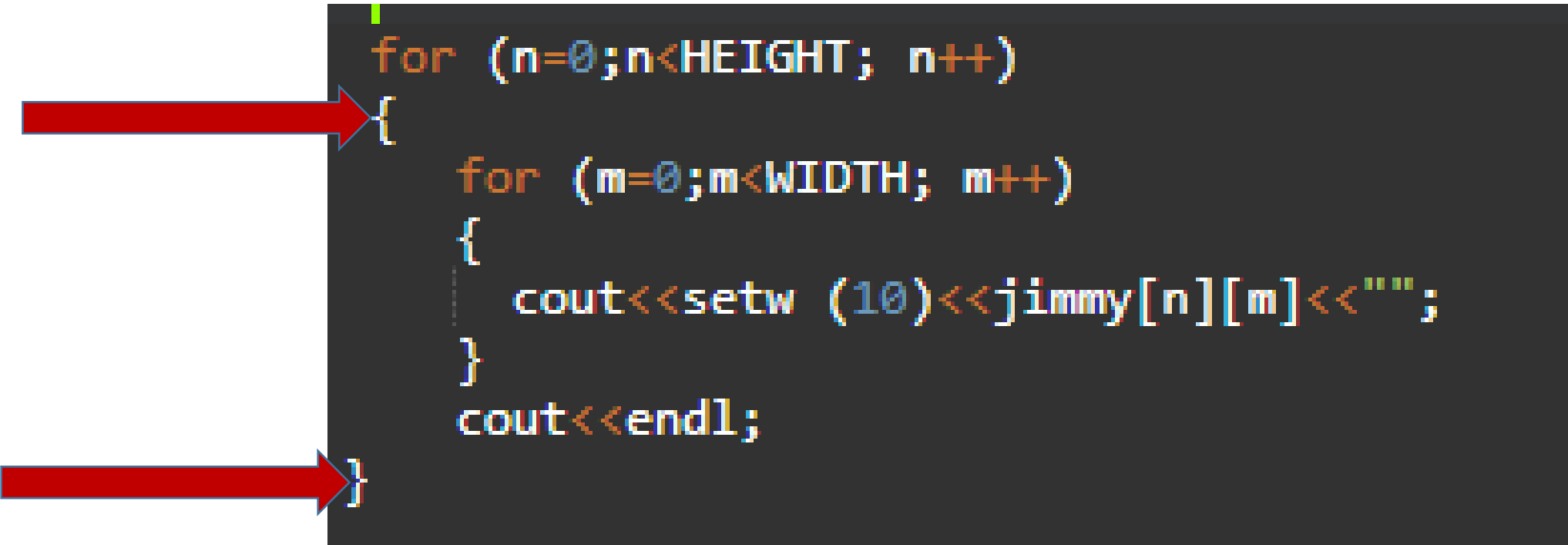
Exercise 2

Write a program that populates a two-dimensional (2X3) array where *user can enter* the values for each cell.

```cpp
#include <iostream>
#define height 2
#define width 3
using namespace std;
int x,y; // Global Variable
int main ()
{
  int myMatirx[height][width];   // 2D array defined


  for (x=0; x<height; x++)
      for (y=0; y<width; y++)
      {
        cout <<"Enter the value\n";
        cin >> myMatirx [x][y];
        cout << "\n\n";
      }
  return 0;
}
```

Assignment 1

Update the code; Generate a 3x3 2D-array where user can enter the values for each cell. And, display the matrix

```
Developed by -----
        1               2               3               4               5
        2               4               6               8              10
        3               6               9              12              15
```

Use SETW() to have nice matrix layout

DELIVERABLE :
Once you developed the c++ program, then execute/run the code. Save it LastNameW9A1.CPP

Assignment 2

Update the given exercise :

To populate 3X3 matrix. Name this one as myMatrix [3][3]

- Ask user to enter a value for each cell.

- Show this matrix on the screen

- Secondly, Ask user to enter an integer number to add all members of the developed matrix. Adding one number to each number in the whole matrix

- Add given value to all members of the matrix. This matrix should be named as myUpdatedMatrix

- Show this   new matrix on the screen

Assignment 2 | Expected output


```
=== Developed by ---- ===
 Original Matrix
        1               2               3               4               5
        2               4               6               8              10
        3               6               9              12              15
Please enter the integer value to add 88
 Updated  Matrix
       89              90              91              92              93
       90              92              94              96              98
       91              94              97             100             103
```

DELIVERABLE :
Once you developed the c++ program, then execute/run the code. Save it LastNameW9A2.CPP