```c
#include <stdio.h>

#define NUM_ROWS 5
#define NUM_COLUMNS 3

int main(void)
{
    //variable declarations
    int iArray[NUM_ROWS][NUM_COLUMNS];
    int i, j;

    //code
    // *** NAME OF AN ARRAY ITSELF IS ITS BASE ADDRESS ***
    // *** HENCE, 'iArray' IS BASE ADDRESS OF 2D INTEGER ARRAY iArray[][]

    // iArray[5][3] => iArray IS A 2D ARRAY HAVING 5 ROWS AND 3 COLUMNS. EACH OF THESE 5 ROWS IS A 1D INTGER ARRAY OF 3 INTEGERS ***
    // iArray[0] => IS THE 0TH ROW ... HENCE, IS THE BASE ADDRESS OF ROW 0
    // iArray[1] => IS THE 1ST ROW ... HENCE, IS THE BASE ADDRESS OF ROW 1
    // iArray[4] => IS THE 4TH ROW ... HENCE, IS THE BASE ADDRESS OF ROW 4

    // (iArray[0] + 0) => ADDRESS OF 0th INTEGER FROM BASE ADDRESS OF 0th ROW (iArray[0][0])
    // (iArray[0] + 1) => ADDRESS OF 1ST INTEGER FROM BASE ADDRESS OF 0th ROW (iArray[0][1])
    // (iArray[0] + 2) => ADDRESS OF 2ND INTEGER FROM BASE ADDRESS OF 0th ROW (iArray[0][2])

    // (iArray[1] + 0) => ADDRESS OF 0th INTEGER FROM BASE ADDRESS OF 1ST ROW (iArray[1][0])
    // (iArray[1] + 1) => ADDRESS OF 1ST INTEGER FROM BASE ADDRESS OF 1ST ROW (iArray[1][1])
    // (iArray[1] + 2) => ADDRESS OF 2ND INTEGER FROM BASE ADDRESS OF 1ST ROW (iArray[1][2])

    // iArray[0], iArray[1] ... ARE 1D INTEGR ARRAYS AND HENCE CAN BE TREATED AS 1D INTEGER ARRAYS USING POINTERS ...
    // 'iArray' IS THE NAME AND BASE ADDRESS OF 2D INTEGER ARRAY ***
    // (*(iArray + 0) + 0) = (iArray[0] + 0) = ADDRESS OF 0TH ELEMENT FROM BASE ADDRESS OF 0TH ROW = (iArray[0] + 0) = (iArray[0][0])
    // (*(iArray + 0) + 1) = (iArray[0] + 1) = ADDRESS OF 1ST ELEMENT FROM BASE ADDRESS OF 0TH ROW = (iArray[0] + 1) = (iArray[0][1])
    // (*(iArray + 0) + 2) = (iArray[0] + 2) = ADDRESS OF 2ND ELEMENT FROM BASE ADDRESS OF 0TH ROW = (iArray[0] + 2) = (iArray[0][2])

    // (*(iArray + 1) + 0) = (iArray[1] + 0) = ADDRESS OF 0TH ELEMENT FROM BASE ADDRESS OF 1ST ROW = (iArray[1] + 0) = (iArray[1][0])
    // (*(iArray + 1) + 1) = (iArray[1] + 1) = ADDRESS OF 1ST ELEMENT FROM BASE ADDRESS OF 1ST ROW = (iArray[1] + 1) = (iArray[1][1])
    // (*(iArray + 1) + 2) = (iArray[1] + 2) = ADDRESS OF 2ND ELEMENT FROM BASE ADDRESS OF 1ST ROW = (iArray[1] + 2) = (iArray[1][2])


    for (i = 0; i < NUM_ROWS; i++)
    {
        for (j = 0; j < NUM_COLUMNS; j++)
```

```c
            *(*(iArray + i) + j) = (i + 1) * (j + 1); // 'iArray[i]' Can Be Treated As 1D Array Using Pointers ...
    }

    printf("\n\n");
    printf("2D Integer Array Elements Along With Addresses : \n\n");
    for (i = 0; i < NUM_ROWS; i++)
    {
        for (j = 0; j < NUM_COLUMNS; j++)
        {
            printf("*(*(iArray + %d) + %d)= %d \t \t At Address (*(iArray + %d) + %d) : %p\n", i,  j, *(*(iArray + i) + j), i, j, (*(iArray + i) + j));
        }
        printf("\n\n");
    }

    return(0);
}
```