```
1 #include <stdio.h>
 3
   #define NUM ROWS 5
 4
   #define NUM_COLUMNS 3
 5
   int main(void)
 6
 7
       //variable declarations
 8
 9
       int iArray[NUM_ROWS][NUM_COLUMNS];
10
       int i, j;
11
12
       int *ptr iArray Row = NULL;
13
14
       //code
       // *** EVERY ROW OF A 2D ARRAY IS AN INTEGER ARRAY ITSELF COMPRISING OF
15
          'NUM COLUMNS' INTEGER ELEMENTS ***
       // *** THERE ARE 5 ROWS AND 3 COLUMNS IN A 2D INTEGER ARRAY. EACH OF THE 5 >
16
          ROWS IS A 1D ARRAY OF 3 INTEGERS.
       // *** HENCE, EACH OF THESE 5 ROWS THEMSELVES BEING ARRAYS, WILL BE THE
17
          BASE ADDRESSES OF THEIR RESPECTIVE ROWS ***
18
       for (i = 0; i < NUM ROWS; i++)
19
20
21
            ptr_iArray_Row = iArray[i]; // 'iArray[i]' IS THE BASE ADDRESS OF ith
              ROW ...
            for (j = 0; j < NUM_COLUMNS; j++)</pre>
22
                *(ptr_iArray_Row + j) = (i + 1) * (j + 1); //
23
                  'ptr_iArray_Row' (That is, 'iArray[i]' Can Be Treated As 1D Array →
                   Using Pointers) ...
24
       }
25
       printf("\n\n");
26
       printf("2D Integer Array Elements Along With Addresses : \n\n");
27
28
       for (i = 0; i < NUM ROWS; i++)
29
       {
30
            ptr_iArray_Row = iArray[i];
            for (j = 0; j < NUM_COLUMNS; j++)</pre>
31
32
            {
33
                printf("*(ptr_iArray_Row + %d)= %d \t \t At Address (ptr_iArray_Row >
                   + j) : %p\n", j, *(ptr_iArray_Row + j), (ptr_iArray_Row + j));
34
            printf("\n\n");
35
36
       }
37
38
       return(0);
39
   }
40
41
42
43
44
```