```c
#include <stdio.h>

int main(void)
{
    //variable declarations
    int iArray[] = { 10, 20, 30, 40, 50, 60, 70, 80, 90, 100 }; //Integer Array
    int *ptr_iArray = NULL; //Integer Pointer

    //code

    // ######  USING ARRAY NAME AS A POINTER i.e : Value Of xth Element Of iArray : *(iArray + x) AND Address Of xth Element Of iArray : (iArray + x) ######
    printf("\n\n");
    printf("*** USING ARRAY NAME AS A POINTER i.e : Value Of xth Element Of iArray : *(iArray + x) AND Address Of xth Element Of iArray : (iArray + x) ***\n\n");
    printf("Integer Array Elements And Their Addresses : \n\n");
    printf("*(iArray + 0) = %d \t At Address (iArray + 0) : %p\n", *(iArray + 0), (iArray + 0));
    printf("*(iArray + 1) = %d \t At Address (iArray + 1) : %p\n", *(iArray + 1), (iArray + 1));
    printf("*(iArray + 2) = %d \t At Address (iArray + 2) : %p\n", *(iArray + 2), (iArray + 2));
    printf("*(iArray + 3) = %d \t At Address (iArray + 3) : %p\n", *(iArray + 3), (iArray + 3));
    printf("*(iArray + 4) = %d \t At Address (iArray + 4) : %p\n", *(iArray + 4), (iArray + 4));
    printf("*(iArray + 5) = %d \t At Address (iArray + 5) : %p\n", *(iArray + 5), (iArray + 5));
    printf("*(iArray + 6) = %d \t At Address (iArray + 6) : %p\n", *(iArray + 6), (iArray + 6));
    printf("*(iArray + 7) = %d \t At Address (iArray + 7) : %p\n", *(iArray + 7), (iArray + 7));
    printf("*(iArray + 8) = %d \t At Address (iArray + 8) : %p\n", *(iArray + 8), (iArray + 8));
    printf("*(iArray + 9) = %d \t At Address (iArray + 9) : %p\n", *(iArray + 9), (iArray + 9));

    // ASSIGNING BASE ADDRESS OF INTEGER ARRAY 'iArray' TO INTEGER POINTER 'ptr_iArray'
    // NAME OF ANY ARRAY IS ITS OWN BASE ADDRESS
    ptr_iArray = iArray; //SAME AS ... ptr_iArray = &iArray[0]

    // ######  USING POINTER AS ARRAY NAME i.e : Value Of xth Element Of iArray : ptr_iArray[x] AND Address Of xth Element Of iArray : &ptr_iArray[x]  ######
    printf("\n\n");
    printf("*** USING POINTER AS ARRAY NAME i.e : Value Of xth Element Of iArray : ptr_iArray[x] AND Address Of xth Element Of iArray : &ptr_iArray[x] ***\n\n");
    printf("Integer Array Elements And Their Addresses : \n\n");
    printf("ptr_iArray[0] = %d \t At Address &ptr_iArray[0] : %p\n", ptr_iArray
```

```
        [0], &ptr_iArray[0]);
35      printf("ptr_iArray[1] = %d \t At Address &ptr_iArray[1] : %p\n", ptr_iArray
        [1], &ptr_iArray[1]);
36      printf("ptr_iArray[2] = %d \t At Address &ptr_iArray[2] : %p\n", ptr_iArray
        [2], &ptr_iArray[2]);
37      printf("ptr_iArray[3] = %d \t At Address &ptr_iArray[3] : %p\n", ptr_iArray
        [3], &ptr_iArray[3]);
38      printf("ptr_iArray[4] = %d \t At Address &ptr_iArray[4] : %p\n", ptr_iArray
        [4], &ptr_iArray[4]);
39      printf("ptr_iArray[5] = %d \t At Address &ptr_iArray[5] : %p\n", ptr_iArray
        [5], &ptr_iArray[5]);
40      printf("ptr_iArray[6] = %d \t At Address &ptr_iArray[6] : %p\n", ptr_iArray
        [6], &ptr_iArray[6]);
41      printf("ptr_iArray[7] = %d \t At Address &ptr_iArray[7] : %p\n", ptr_iArray
        [7], &ptr_iArray[7]);
42      printf("ptr_iArray[8] = %d \t At Address &ptr_iArray[8] : %p\n", ptr_iArray
        [8], &ptr_iArray[8]);
43      printf("ptr_iArray[9] = %d \t At Address &ptr_iArray[9] : %p\n", ptr_iArray
        [9], &ptr_iArray[9]);
44      return(0);
45  }
46
47
48
```