



# 2D ARRAY

# 2D ARRAYS

- 2D array can be defined as an **array of arrays**.
- The 2D array is organized as matrices which can be represented as the collection of rows and columns.
- However, 2D arrays are created to implement a relational database look alike data structure.



# HOW TO DECLARE 2D ARRAY

```
int arr[max_rows][max_columns];
```

	0	1	2	.....	n-1
0	a[0][0]	a[0][1]	a[0][2]	.....	a[0][n-1]
1	a[1][0]	a[1][1]	a[1][2]	.....	a[1][n-1]
2	a[2][0]	a[2][1]	a[2][2]	.....	a[2][n-1]
3	a[3][0]	a[3][1]	a[3][2]	.....	a[3][n-1]
4	a[4][0]	a[4][1]	a[4][2]	.....	a[4][n-1]
.	.	.	.	.....	.
.	.	.	.	.....	.
n-1	a[n-1][0]	a[n-1][1]	a[n-1][2]	.....	a[n-1][n-1]

a[n][n]



## INITIALIZING AN ARRAY

There are two ways to initialize a two Dimensional arrays during declaration.

```
int disp[2][4] = {  
    {10, 11, 12, 13},  
    {14, 15, 16, 17}  
};
```

OR

```
int disp[2][4] = { 10, 11, 12, 13, 14, 15, 16, 17};
```



# TWO-DIMENSIONAL ARRAY EXAMPLE IN C

```
#include <stdio.h>

int main(){
int i=0,j=0;
int arr[4][3]={{1,2,3},{2,3,4},{3,4,5},{4,5,6}};

//traversing 2D array
for(i=0;i<4;i++){
for(j=0;j<3;j++){
printf("arr[%d] [%d] = %d \n",i,j,arr[i][j]);
}
}
return 0;
}
```

## Output

```
arr[0][0] = 1
arr[0][1] = 2
arr[0][2] = 3
arr[1][0] = 2
arr[1][1] = 3
arr[1][2] = 4
arr[2][0] = 3
arr[2][1] = 4
arr[2][2] = 5
arr[3][0] = 4
arr[3][1] = 5
arr[3][2] = 6
```

# 2D ARRAY EXAMPLE: STORING ELEMENTS IN A MATRIX AND PRINTING IT

```
#include <stdio.h>
void main ()
{
    int arr[3][3],i,j;
    for (i=0;i<3;i++)
    {
        for (j=0;j<3;j++)
        {
            printf("Enter a[%d][%d]: ",i,j);
            scanf("%d",&arr[i][j]);
        }
    }
    printf("\n printing the elements ....\n");
```

```
for(i=0;i<3;i++)
{
    printf("\n");
    for (j=0;j<3;j++)
    {
        printf("%d\t",arr[i][j]);
    }
}
```



## Output

```
Enter a[0][0]: 56
Enter a[0][1]: 10
Enter a[0][2]: 30
Enter a[1][0]: 34
Enter a[1][1]: 21
Enter a[1][2]: 34

Enter a[2][0]: 45
Enter a[2][1]: 56
Enter a[2][2]: 78

printing the elements ....

56      10      30
34      21      34
45      56      78
```



## 2D ARRAY PRINTING AND CALCULATING SUM

```
1 #include<stdio.h>
2 void main()
3 {
4 int a[2][3],i,j,sum;
5 printf("Enter the matrix values:");
6 for (i=0;i<2;i++)
7 {
8 for(j=0;j<3;j++)
9 {
10 scanf("%d",&a[i][j]);
11 }
12 }
13 for(i=0;i<2;i++)
14 {
15 for(j=0;j<3;j++)
16 {
17 printf("%d\t",a[i][j]);
18 }
19 printf("\n");
20 }
21 for(i=0;i<2;i++)
22 {
23 sum=0;
24 for(j=0;j<3;j++)
25 {
26 sum=sum+a[i][j];
27 }
28 }
29 }
30 printf("sum is %d",sum);
31 }
32
```





# MATRIX TRANSPOSE

```
1 #include<stdio.h>
2 void main()
3 {
4     int a[2][3],i,j;
5     printf("Enter the matrix values:");
6     for (i=0;i<2;i++)
7     {
8         for(j=0;j<3;j++)
9         {
10            scanf("%d",&a[i][j]);
11        }
12    }
13    for(i=0;i<2;i++)
14    {
15        for(j=0;j<3;j++)
16        {
17            printf("%d\t",a[i][j]);
18        }
19        printf("\n");
20    }
21    printf("The transpose of matrix is :\n");
22    for(i=0;i<3;i++)
23    {
24        for(j=0;j<2;j++)
25        {
26            printf("%d\t",a[j][i]);
27        }
28    }
29    printf("\n");
30 }
```

```
ubuntu@ubuntu-Lenovo-ideapad-300-15ISK:~/Desktop$ ./a.out
Enter the matrix values:1 2 3 4 5 6
1      2      3
4      5      6
The transpose of matrix is :
1      4
2      5
3      6
ubuntu@ubuntu-Lenovo-ideapad-300-15ISK:~/Desktop$
```

```
#include<stdio.h>
void main()
{
int a[2][3],b[3][2],i,j,c[3][2];
printf("Enter the matrix values:");
for (i=0;i<2;i++)
{
for(j=0;j<3;j++)
{
scanf("%d",&a[i][j]);
}
}
for(i=0;i<2;i++)
{
for(j=0;j<3;j++)
{
printf("%d\t",a[i][j]);
}
printf("\n");
}
printf("The transpose of matrix is :\n");
/*Find the transpose of matrix*/
for(i=0;i<2;i++)
{
for(j=0;j<3;j++)
{
c[j][i]=a[i][j];
}
printf("\n");
}
/*To print the transpose of matrix*/
for(i=0;i<3;i++)
{
for(j=0;j<2;j++)
{
printf("%d\t",c[i][j]);
}
printf("\n");
}
}
```



# MATRIX ADDITION

```
1 #include<stdio.h>
2 void main()
3 {
4 int a[50][50],b[50][50],c[50][50],i,j;
5 printf("Enter the first matrix values:");
6 for (i=0;i<2;i++)
7 {
8 for(j=0;j<3;j++)
9 {
10 scanf("%d",&a[i][j]);
11 }
12 }
13 printf("\nThe first matrix is:\n");
14 for(i=0;i<2;i++)
15 {
16 for(j=0;j<3;j++)
17 {
18 printf("%d\t",a[i][j]);
19 }
20 printf("\n");
21 }
22 printf("Enter the second matrix values:");
23 for (i=0;i<2;i++)
24 {
25 for(j=0;j<3;j++)
26 {
27 scanf("%d",&b[i][j]);
28 }
29 }
30 printf("\nThe Second matrix is:\n");
31 for(i=0;i<2;i++)
32 {
33 for(j=0;j<3;j++)
34 {
35 printf("%d\t",b[i][j]);
36 }
37 printf("\n");
38 }
39 printf("\nThe sum of two matrix is :\n");
40
41 for(i=0;i<2;i++)
42 {
43 c[i][j]=0;
44 for(j=0;j<3;j++)
45 {
46 c[i][j]=a[i][j]+b[i][j];
47 printf("%d\t",c[i][j]);
48 }
49 printf("\n");
50 }
51
52 }
```

```
ubuntu@ubuntu-Lenovo-ideapad-300-15ISK:~/Desktop$ ./a.out
```

```
Enter the first matrix values:1 2 3 4 5 6
```

```
The first matrix is:
```

```
1      2      3
```

```
4      5      6
```

```
Enter the second matrix values:1 2 3 4 5 6
```

```
The Second matrix is:
```

```
1      2      3
```

```
4      5      6
```

```
The sum of two matrix is :
```

```
2      4      6
```

```
8      10     12
```



# MATRIX MULTIPLICATION

```
1 #include<stdio.h>
2 void main()
3 {
4 int a[100][100],b[100][100],c[100][100],i,j,m,n,p,q,k;
5 printf("Enter rows and columns of first matrix");
6 scanf("%d%d",&m,&n);
7 printf("Enter the first matrix values:");
8 for (i=0;i<m;i++)
9 {
10 for(j=0;j<n;j++)
11 {
12 scanf("%d",&a[i][j]);
13 }
14 }
15 printf("\nThe first matrix is:\n");
16 for(i=0;i<m;i++)
17 {
18 for(j=0;j<n;j++)
19 {
20 printf("%d\t",a[i][j]);
21 }
22 printf("\n");
23 }
24 printf("Enter rows and columns of second matrix");
25 scanf("%d%d",&p,&q);
26 printf("Enter the second matrix values:");
27 for (i=0;i<p;i++)
28 {
29 for(j=0;j<q;j++)
30 {
31 scanf("%d",&b[i][j]);
32 }
33 }
```

```
34 printf("\nThe Second matrix is:\n");
35 for(i=0;i<p;i++)
36 {
37 for(j=0;j<q;j++)
38 {
39 printf("%d\t",b[i][j]);
40 }
41 printf("\n");
42 }
43 if (n!=p)
44 {
45 printf("\nThe multiplication is not possible :\n");
46 }
47 else
48 {
49 printf("\n The product of matrices is:\n");
50 for(i=0;i<m;i++)
51 {
52 c[i][j]=0;
53 for(j=0;j<q;j++)
54 {
55 for(k=0;k<m;k++)
56 {
57 c[i][j]=c[i][j]+a[i][k]*b[k][j];
58 }
59 }
60 printf("%d\t",c[i][j]);
61 }
62 printf("\n");
63 }
64 }
```

```
ubuntu@ubuntu-Lenovo-ideapad-300-15ISK:~/Desktop$ ./a.out
```

```
Enter rows and columns of first matrix 3 3
```

```
Enter the first matrix values: 1 2 3 1 2 1 4 0 2
```

```
The first matrix is:
```

```
1 2 3
```

```
1 2 1
```

```
4 0 2
```

```
Enter rows and columns of second matrix 3 2
```

```
Enter the second matrix values:1 2 3 1 1 0
```

```
The Second matrix is:
```

```
1 2
```

```
3 1
```

```
1 0
```

```
The product of matrices is:
```

```
10 4
```

```
8 4
```

```
6 8
```

```
ubuntu@ubuntu-Lenovo-ideapad-300-15ISK:~/Desktop$ ./a.out
```

```
Enter rows and columns of first matrix 3 3
```

```
Enter the first matrix values:1 2 3 1 2 1 4 0 2
```

```
The first matrix is:
```

```
1 2 3
```

```
1 2 1
```

```
4 0 2
```

```
Enter rows and columns of second matrix 2 3
```

```
Enter the second matrix values: 1 2 3 4 5 6
```

```
The Second matrix is:
```

```
1 2 3
```

```
4 5 6
```

```
The multiplication is not possible :
```