

1- Write a program that declares a 2D character array to store names and initializes it with three names ("Amr", "Omar", "Ali"). Without using a loop, print each name on a new line.

اكتب برنامجا يعلن عن 2D character array لتخزين الأسماء وتهيئتها بثلاثة أسماء (عمرو، عمر، علي). Without using a loop، قم بطباعة كل اسم على سطر جديد.

## Output

```
Name 1: Amr
Name 2: Omar
Name 3: Ali
```

## Solution

```
// www.gammal.tech
#include<stdio.h>
#include<string.h>

int main() {
    // Declare a 2D character array to store names and initialize it
    char names[3][5] = {"Amr", "Omar", "Ali"};

    // Print each name on a new line without using a loop
    printf("Name 1: %s\n", names[0]);
    printf("Name 2: %s\n", names[1]);
    printf("Name 3: %s\n", names[2]);

    return 0;
}
```

2- Write a program that declares a 2D character array to store names and initializes it with four names ("Amr", "Omar", "Ali", "Nada"). Using a loop, print each name on a new line.

اكتب برنامجاً يعلن عن 2D character array لتخزين الأسماء وتتهيئتها بأربعة أسماء (عمر، علي، ندى). باستخدام حلقة، قم بطباعة كل اسم على سطر جديد.

## Output

```
Amr
Omar
Ali
Nada
```

## Solution

```
// www.gammal.tech
#include<stdio.h>
#include<string.h>

int main() {
    // Declare a 2D character array to store names and initialize it
    char names[4][5] = {"Amr", "Omar", "Ali", "Nada"};

    // Print each name on a new line using a loop
    for (int i = 0; i < 4; i++) {
        printf("%s\n", names[i]);
    }

    return 0;
}
```

---

3- Write a program that declares a 2D character array to store names and initializes it with four names ("Amr", "Omar", "Ali", "Nada"). Using a loop, print the first character of each name on a new line.

اكتب برنامجاً يعلن عن 2D character array لتخزين الأسماء وتهيئتها بأربعة أسماء (عمر، عمر، علي، ندى). باستخدام حلقة، قم بطباعة الحرف الأول من كل اسم على سطر جديد.

## Output

```
A
O
A
N
```

## Solution

```
// www.gammal.tech
#include<stdio.h>
#include<string.h>

int main() {
    // Declare a 2D character array to store names and initialize it
    char names[4][5] = {"Amr", "Omar", "Ali", "Nada"};

    // Print the first character of each name on a new line using a loop
    for (int i = 0; i < 4; i++) {
        printf("%c\n", names[i][0]);
    }

    return 0;
}
```

---

4- Write a program that declares a 2D character array to store names and initializes it with four names ("Amr", "Omar", "Ali", "Nada"). Using a loop, print the ASCII value of the second character of each name on a new line.

اكتب برنامجاً يعلن عن 2D character array لتخزين الأسماء وتهيئتها بأربعة أسماء (عمر، عمر، علي، ندى). باستخدام حلقة، قم بطباعة قيمة ASCII للحرف الثاني من كل اسم على سطر جديد.

## Output

```
109
109
108
97
```

## Solution

```
// www.gammal.tech
#include<stdio.h>
#include<string.h>

int main() {
    // Declare a 2D character array to store names and initialize it
    char names[4][5] = {"Amr", "Omar", "Ali", "Nada"};

    // Print the ASCII value of the second character of each name on a new line using a loop
    for (int i = 0; i < 4; i++) {
        printf("%d\n", names[i][1]);
    }

    return 0;
}
```

5- Write a program that declares a 2D character array to store names and initializes it with four names ("Amr", "Omar", "Ali", "Nada"). Using nested loops, calculate and print the total number of characters in all the names.

اكتب برنامجاً يعلن عن 2D character array لتخزين الأسماء وتهيئتها بأربعة أسماء (عمرو، عمر، علي، ندى). باستخدام الحلقات المتداخلة، قم بحساب وطباعة العدد الإجمالي للأحرف في جميع الأسماء.

## Output

```
Total number of characters in all names: 14
```

## Solution

```
// www.gammal.tech
#include<stdio.h>
#include<string.h>

int main() {
    int counter = 0;
    char names[4][5] = {"Amr", "Omar", "Ali", "Nada"};

    // Calculate the total number of characters in all names using nested loops
    for (int i = 0; i < 4; i++) {
        for (int j = 0; names[i][j] != '\0'; j++) {
            counter++;
        }
    }

    // Print the total number of characters
    printf("Total number of characters in all names: %d", counter);

    return 0;
}
```

6- Write a program that declares a 2D character array to store names and initializes it with four names ("Amr", "Omar", "Ali", "Nada"). Using nested loops, calculate and print the number of characters in each name along with the respective names.

اكتب برنامجاً يعلن عن 2D character array لتخزين الأسماء وتهيئتها بأربعة أسماء (عمر، علي، ندى). باستخدام الحلقات المتداخلة، قم بحساب وطباعة عدد الأحرف في كل اسم بالإضافة إلى الأسماء المعنية.

## Output

```
Amr has 3 characters
Omar has 4 characters
Ali has 3 characters
Nada has 4 characters
```

## Solution

```
// www.gammal.tech
#include<stdio.h>
#include<string.h>

int main() {
    int counter = 0;
    char names[4][5] = {"Amr", "Omar", "Ali", "Nada"};

    for (int i = 0; i < 4; i++) {
        // Calculate the number of characters in each name
        for (int j = 0; names[i][j] != '\0'; j++) {
            counter++;
        }

        // Print the name along with the number of characters
        printf("%s has %d characters\n", names[i], counter);

        // Reset the counter for the next name
        counter = 0;
    }

    return 0;
}
```

7- Write a program that declares a 2D character array to store names and initializes it with four names ("Amr", "Omar", "Ali", "Nada"). Using a loop, print each name on a new line in reverse order (starting from "Nada" and ending with "Amr").

اكتب برنامجاً يعلن عن 2D character array لتخزين الأسماء وتهيئتها بأربعة أسماء (عمرو، عمر، علي، ندى). باستخدام حلقة، قم بطباعة كل اسم على سطر جديد بترتيب عكسي (يبدأ من "ندى" وينتهي بـ "عمرو").

## Output

```
Nada
Ali
Omar
Amr
```

## Solution

```
// www.gammal.tech
#include<stdio.h>
#include<string.h>

int main() {
    char names[4][5] = {"Amr", "Omar", "Ali", "Nada"};

    for (int i = 3; i >= 0; i--) {
        // Print each name on a new line in reverse order
        printf("%s\n", names[i]);
    }

    return 0;
}
```

8- Write a program that declares a 2D character array to store names and initializes it with four names ("Amr", "Omar", "Ali", "Nada"). Using nested loops, print each name in reverse order (from the last character to the first) on a single line.

اكتب برنامجاً يعلن عن 2D character array لتخزين الأسماء وتتهيئتها بأربعة أسماء (عمرو، عمر، علي، ندى). باستخدام الحلقات المتداخلة، قم بطباعة كل اسم بترتيب عكسي (من الحرف الأخير إلى الأول) على سطر واحد.

## Output

```
rmA
ramO
ilA
adaN
```

## Solution

```
// www.gammal.tech
#include<stdio.h>
#include<string.h>

int main() {
    char names[4][5] = {"Amr", "Omar", "Ali", "Nada"};

    for (int i = 0; i < 4; i++) {
        // Print each name in reverse order
        for (int j = strlen(names[i]) - 1; j >= 0; j--) {
            printf("%c", names[i][j]);
        }

        // Print a newline after each name
        printf("\n");
    }

    return 0;
}
```

9- Write a program that declares a 2D character array to store names and initializes it with four names ("Amr", "Omar", "Ali", "Nada"). Using a loop, check if the name "Omar" is present in the array. If it is found, print "found"; otherwise, do nothing.

اكتب برنامجاً يعلن عن 2D character array لتخزين الأسماء وتتهيئتها بأربعة أسماء (عمر، علي، ندى). باستخدام حلقة، تحقق من وجود اسم "عمر" في array. إذا تم العثور عليه، طباعة "وجدت"؛ خلاف ذلك، لا تفعل شيئاً.

## Output

```
found
```



## Solution

```
// www.gammal.tech
#include<stdio.h>
#include<string.h>

int main() {
    char names[4][5] = {"Amr", "Omar", "Ali", "Nada"};

    for (int i = 0; i < 4; i++) {
        // Check if the current name is "Omar"
        if (strcmp(names[i], "Omar") == 0) {
            // If found, print "found"
            printf("found\n");
            // Break out of the loop since the name is found
            break;
        }
    }

    return 0;
}
```

10- Write a program that declares a 2D character array to store names and initializes it with four names ("Amr", "Omar", "Ali", "Nada"). Using a loop, print the names that start with the letter 'A'.

اكتب برنامجاً يعلن عن 2D character array لتخزين الأسماء وتتهيئتها بأربعة أسماء ("Amr", "Omar", "Ali", "Nada"). باستخدام loop قم بطباعة الأسماء التي تبدأ بالحرف "A".

## Output

```
Amr
Ali
```

# Solution

```

    ● ● ●
// www.gammal.tech
#include<stdio.h>
#include<string.h>

int main() {
    char names[4][5] = {"Amr", "Omar", "Ali", "Nada"};

    for (int i = 0; i < 4; i++) {
        // Check if the first character of the current name is 'A'
        if (names[i][0] == 'A') {
            // If true, print the name
            printf("%s\n", names[i]);
        }
    }

    return 0;
}
```

---