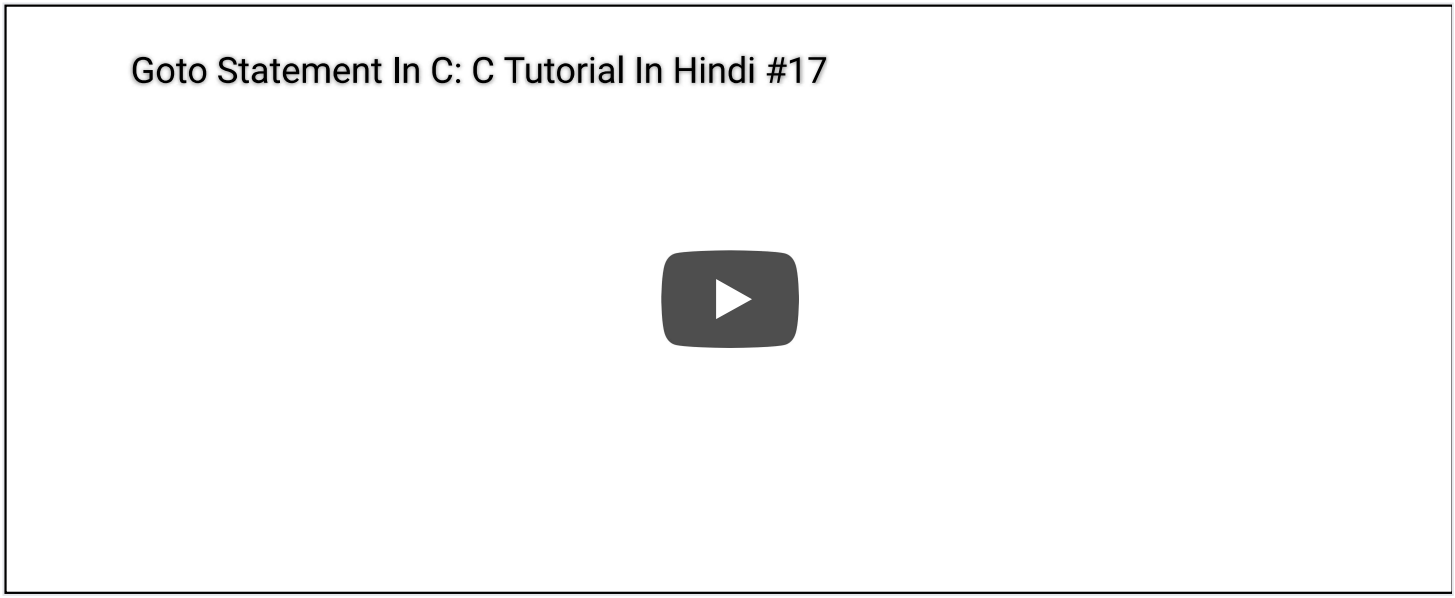
CodeWithHarry



Course: C Language Tutorials For Beginners

Overview Q&A Downloads Announcements

Goto Statement In C: C Tutorial In Hindi #17

- A goto statement in C programming language provides an unconditional jump from the 'goto' to a labeled statement in the same function.

NOTE – Use of **goto** statements is highly discouraged or avoided in any programming language because it makes difficult to trace the control flow of a program to fellow programmers, making the program hard to understand and hard to modify or manipulate. Any program which uses goto can be modified to avoid goto statements.

- These are also called 'Jump Statement'.
- It is used to transfer the control to a predefined label.
- It's use is avoided since it causes confusion for the fellow programmers in understanding code.
- goto statement is preferable when we need to break multiple loops using a single statement at the same time.

Syntax for goto statement :

```c
#include <stdio.h>
int main()
{
    int x;
    for (int i = 0;i <5; i++)
    {
        printf("\nHey Guys\n\n");
        for (int j = ;j < 3; j++)
        {
            printf("Type any No. & To Exit : Press 1\n");
            scanf("%d", &x);
            if (x == 1)
            {
                goto end; // This goto will transfer the control to end: i.e. c
            }
        }
    }
    end:
    printf("\'For\' loops are skipped as you pressed 1");
    return 0;
}
```

That's all about goto statement in C Language.

Code as described/written in the video

```c
#include <stdio.h>

int main()
{
    // label:
    //     printf("we are inside label");
    //     goto end;
    // printf("Hello World\n");
    // goto label;
    // end:
    //     printf("we are at end");
    int num;
    for(int i = 0; i < 8; i++)
    {
        printf("%d\n", i);
        for(int j = 0; j < 8; j++)
        {
            printf("Enter the number. enter 0 to exit\n");
            scanf("%d", &num);
            if(num==0){
                goto end;
            }
```

```
            }

        }
        end:

        return 0;
    }
```

Previous Next

CodeWithHarry