

Կուրսորի դիրքավորումը կոնսոլի պատուհանի վրա

```
#include <windows.h>

void positioningCursor(int x, int y)
{
    HANDLE hOut;
    COORD Position;
    hOut = GetStdHandle(STD_OUTPUT_HANDLE);

    Position.X = x;
    Position.Y = y;
    SetConsoleCursorPosition(hOut, Position);
}

void main() //կարող է լինել main ֆունկցիան լինել որպես հաճախորդ
{
    cout<<"\nSome info...\nSome another info....";
    positioningCursor (rand()%20, rand()%50);
    cout<<"Finally....\n\n";
}
```

```
#include <conio.h>
{
    cout<<“Some Info....\n;
    _getch();
    system("CLS")); ;
}
```

The Cls method resets the CurrentX and CurrentY properties to 0.

Ցուցադրում Էկրանի Կենտրոնական Մասում

```
#include <windows.h>
#define WIDTH 80
#define HEIGHT 22
int main()
{
    HANDLE hCon;
    COORD cPos;

    int countOfStringCharacters=16; //Սա ցուցադրվելիք նախադասության նիշերի քանակն է
    hCon = GetStdHandle(STD_OUTPUT_HANDLE);
    cPos.Y = HEIGHT / 2;
    cPos.X = (WIDTH - countOfStringCharacters) / 2;
    SetConsoleCursorPosition(hCon, cPos);
    cout << "Show Must Go On!" << endl;
    return 0;
}
```

```
system("Color 1A"); std::cout << "\t\t\t Hello World" << std::endl;  
system("Color 2B"); std::cout << "\t\t\t Hello World" << std::endl;  
system("Color 3C"); std::cout << "\t\t\t Hello World" << std::endl;  
system("Color 4D"); std::cout << "\t\t\t Hello World" << std::endl;  
system("Color 5E"); std::cout << "\t\t\t Hello World" << std::endl;  
system("Color 6F"); std::cout << "\t\t\t Hello World" << std::endl;  
system("Color A1"); std::cout << "\t\t\t Hello World" << std::endl;  
system("Color B2"); std::cout << "\t\t\t Hello World" << std::endl;  
system("Color C3"); std::cout << "\t\t\t Hello World" << std::endl;  
system("Color D4"); std::cout << "\t\t\t Hello World" << std::endl;  
system("Color E5"); std::cout << "\t\t\t Hello World" << std::endl;  
system("Color F6"); std::cout << "\t\t\t Hello World" << std::endl;
```

Փորձարկել տարբեր համադրույթներ....

ԳոԼՆափոխման այլ իրականացում

```
someClientFunction()  
{  
    HANDLE hConsole = GetStdHandle(STD_OUTPUT_HANDLE);  
    // կարելի է փոխել k-ի ավելի մեծ արժեքների համար  
    for(int k = 1; k < 255; k++)  
    {  
        SetConsoleTextAttribute(hConsole, k);  
        cout << k << " Life is beautiful!" << endl;  
    }  
}
```

ԳոԼՆափոխումներ և վերադարձ՝ նախկին ձևաչափին

```
int main()
```

```
{
```

```
    HANDLE hConsoleHandle = GetStdHandle(STD_OUTPUT_HANDLE);
```

```
    CONSOLE_SCREEN_BUFFER_INFO *ConsoleInfo = new  
CONSOLE_SCREEN_BUFFER_INFO();
```

```
    GetConsoleScreenBufferInfo(hConsoleHandle, ConsoleInfo);
```

```
    WORD OriginalColors = ConsoleInfo->wAttributes;
```

```
    cout<<"Original Colors";
```

```
    cout<<"Press Enter to Start with...";
```

```
    _getch();
```

```
    SetConsoleTextAttribute(hConsoleHandle, FOREGROUND_GREEN);
```

```
    cout<<"GREEN TEXT";
```

```
    cout<<"Press Enter to change colors again";
```

```
    _getch();
```

ԳԻԼՆԱԿԻՊԻՄԻՆԱԿԵՐ Ա ՎԵՐԱԴԻՄԱՐԺ ՆԱԽԱԿԻՆ ՃԼԱՋԱԿԻՆ

```
SetConsoleTextAttribute(hConsoleHandle, FOREGROUND_RED);  
cout<<"RED TEXT";  
cout<<"Press Enter to change colors again";  
_getch();
```

```
SetConsoleTextAttribute(hConsoleHandle, FOREGROUND_BLUE  
|FOREGROUND_INTENSITY|BACKGROUND_GREEN|BACKGROUND_INT  
ENSITY);  
cout<<"BLUE TEXT OVER A GREEN BACKGROUND";  
cout<<"Press a key to restore the original colors";  
_getch();
```

```
SetConsoleTextAttribute(hConsoleHandle, OriginalColors);  
cout<<"We're Back to original colors";  
cout<<"Press a key to Finish with....";  
_getch();
```

```
return 0;  
}
```

Character Attributes	Meaning
FOREGROUND_BLUE	Text color contains blue.
FOREGROUND_GREEN	Text color contains green.
FOREGROUND_RED	Text color contains red.
FOREGROUND_INTENSITY	Text color is intensified.
BACKGROUND_BLUE	Background color contains blue.
BACKGROUND_GREEN	Background color contains green.
BACKGROUND_RED	Background color contains red.
BACKGROUND_INTENSITY	Background color is intensified.
COMMON_LVB_LEADING_BYTE	Leading byte.
COMMON_LVB_TRAILING_BYTE	Trailing byte.
COMMON_LVB_GRID_HORIZONTAL	Top horizontal.
COMMON_LVB_GRID_LVERTICAL	Left vertical.
COMMON_LVB_GRID_RVERTICAL	Right vertical.
COMMON_LVB_REVERSE_VIDEO	Reverse foreground and background attributes.
COMMON_LVB_UNDERSCORE	Underscore.