

## 07、C 常量

常量是固定值，在程序执行期间不会改变。这些固定的值，又叫做**字面量**。

常量可以是任何的基本数据类型，比如整数常量、浮点常量、字符常量，或字符串字面值，也有枚举常量。

**常量**就像是常规的变量，只不过常量的值在定义后不能进行修改。

### 整数常量

整数常量可以是十进制、八进制或十六进制的常量。前缀指定基数：`0x` 或 `0X` 表示十六进制，`0` 表示八进制，不带前缀则默认表示十进制。

整数常量也可以带一个后缀，后缀是 `U` 和 `L` 的组合，`U` 表示无符号整数（`unsigned`），`L` 表示长整数（`long`）。后缀可以是大写，也可以是小写，`U` 和 `L` 的顺序任意。

下面列举几个整数常量的实例：

```
plaintext
212          /* 合法的 */
215u         /* 合法的 */
0xFeeL       /* 合法的 */
078          /* 非法的：8 不是八进制的数字 */
032UU        /* 非法的：不能重复后缀 */
```

以下是各种类型的整数常量的实例：

```
plaintext
85           /* 十进制 */
0213         /* 八进制 */
0x4b         /* 十六进制 */
30           /* 整数 */
30u          /* 无符号整数 */
30l          /* 长整数 */
30ul         /* 无符号长整数 */
```

### 浮点常量

浮点常量由整数部分、小数点、小数部分和指数部分组成。您可以使用小数形式或者指数形式来表示浮点常量。

当使用小数形式表示时，必须包含小数点、指数，或同时包含两者。当使用指数形式表示时，必须包含整数部分、小数部分，或同时包含两者。带符号的指数是用 e 或 E 引入的。

下面列举几个浮点常量的实例：

plaintext

```
3.14159      /* 合法的 */
314159E-5L   /* 合法的 */
510E         /* 非法的：不完整的指数 */
210f         /* 非法的：没有小数或指数 */
.e55        /* 非法的：缺少整数或分数 */
```

## 字符常量

字符常量是括在单引号中，例如，'x' 可以存储在 char 类型的简单变量中。

字符常量可以是一个普通的字符（例如 'x' ）、一个转义序列（例如 't' ），或一个通用的字符（例如 'u02C0' ）。

在 c 中，有一些特定的字符，当它们前面有反斜杠时，它们就具有特殊的含义，被用来表示如换行符（n）或制表符（t）等。下表列出了一些这样的转义序列码：

转义序列	含义
\\	\ 字符
\'	' 字符
\"	" 字符
?	? 字符
\a	警报铃声
\b	退格键

转义序列	含义
\f	换页符
\n	换行符
\r	回车
\t	水平制表符
\v	垂直制表符
\ooo	一到三位的八进制数
\xhh . . .	一个或多个数字的十六进制数

下面的实例显示了一些转义序列字符：

```

#include <stdio.h>
int main() {
    printf("Hello\tWorld\n\n");
    return 0;
}

```

当上面的代码被编译和执行时，它会产生下列结果：

```

Hello      World

```

字符串面值或常量是括在双引号 `"` 中的。一个字符串包含类似于字符常量的字符：普通的字符、转义序列和通用的字符。

您可以使用空格做分隔符，把一个很长的字符串常量进行分行。

下面的实例显示了一些字符串常量。下面这三种形式所显示的字符串是相同的。

```

"hello, dear"

"hello,

```

```
dear"  
  
"hello, " "d" "ear"
```

在 C 中，有两种简单的定义常量的方式：

1. 使用 **#define** 预处理器。
2. 使用 **const** 关键字。

## define 预处理器

下面是使用 **#define** 预处理器定义常量的形式：

```
#define identifier value
```

具体请看下面的实例：

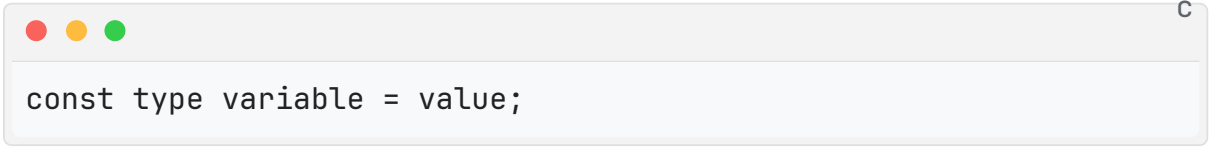
```
#include <stdio.h>  
  
#define LENGTH 10  
#define WIDTH 5  
#define NEWLINE '\n'  
  
int main() {  
    int area;  
    area = LENGTH * WIDTH;  
    printf("value of area : %d", area);  
    printf("%c", NEWLINE);  
    return 0;  
}
```

当上面的代码被编译和执行时，它会产生下列结果：

```
value of area : 50
```

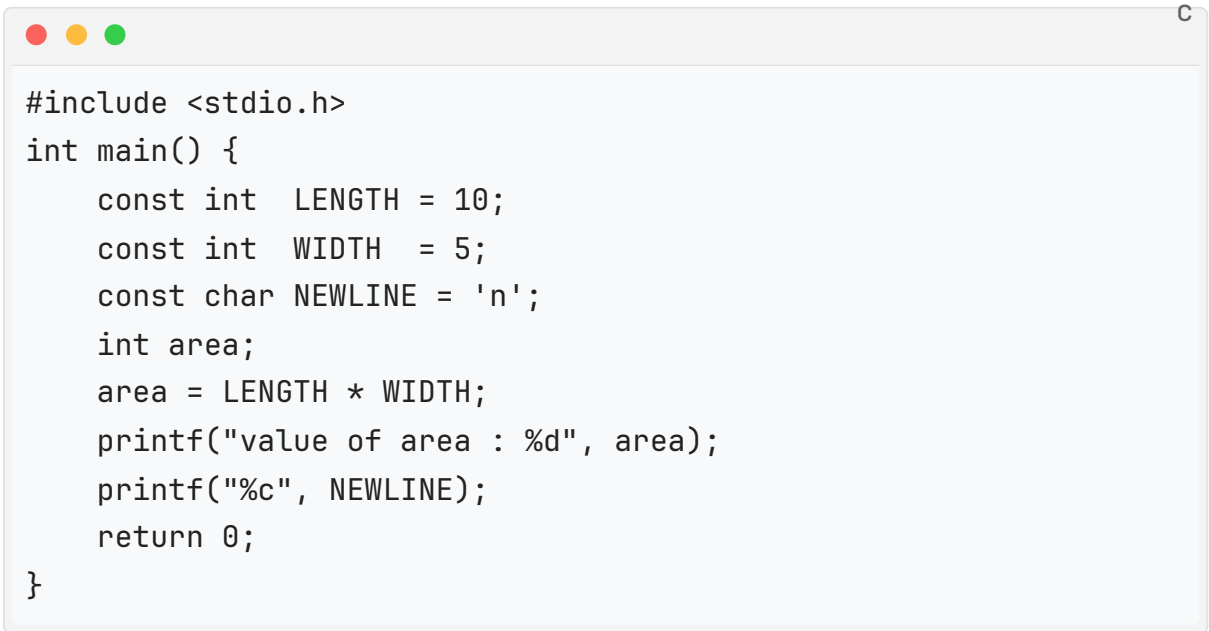
## const 关键字

您可以使用 **const** 前缀声明指定类型的常量，如下所示：



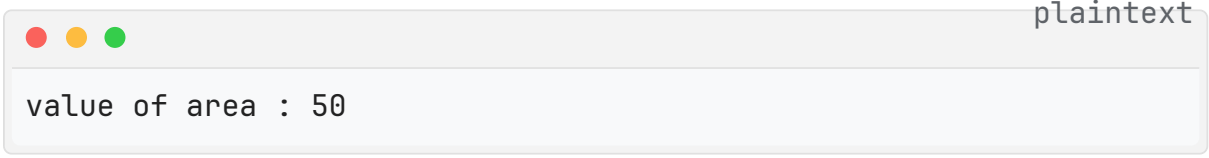
```
const type variable = value;
```

具体请看下面的实例：



```
#include <stdio.h>
int main() {
    const int  LENGTH = 10;
    const int  WIDTH  = 5;
    const char NEWLINE = 'n';
    int area;
    area = LENGTH * WIDTH;
    printf("value of area : %d", area);
    printf("%c", NEWLINE);
    return 0;
}
```

当上面的代码被编译和执行时，它会产生下列结果：



```
value of area : 50
```

请注意，把常量定义为大写字母形式，是一个很好的编程实践。