

# C语言 02 printf 和 scanf

---

printf

scanf

## printf

四种用法

```
printf("字符串");
```

```
printf("输出控制符", 输出参数);
```

```
int a = 1;
```

```
printf("%d\n", a);
```

```
printf("输入控制符1 输入控制符2", 输入参数, 输出参数);
```

```
int a = 1;
```

```
int b = 2;
```

```
printf("%d\n%d\n", a, b);
```

```
printf("输入控制符, 非输入控制符", 输入参数);
```

输出控制符包含如下

%d —— 》 整数int

%ld —— 》 长整数long

%#X (%x, %X, %#x) —— 》 十六进制

%o —— 》 八进制

%f —— 》 浮点数

%lf —— 》 长浮点数

%c —— 》 字符

非输出控制符就是其他的东西，比如“i = ”

```
printf("a = %d\n", a);
```

## 额外十六进制的几种printf方法

```
int x = 78;
```

```
printf("%x\n", x); ——》 4e
```

```
printf("%X\n", x); ——》 4E
```

```
printf("%#x\n", x); ——》 0x4e
```

```
printf("%#X\n", x); ——》 0X4E, 推荐使用！！
```

## scanf

通过键盘将数据输入到变量中

### 两种用法

- scanf("输入控制符", 输入参数的地址);

将用户输入的数据转化为输入控制符相应的数据类型，并存放在以输入参数的值为地址 (&i) 所占的内存中

```
int i;
```

```
scanf("%c", &i);
```

```
printf("i = %d", i);
```

- scanf("非输入控制符+输入控制符", 输入参数的地址);

用户必须先【原样输入】非输入控制符，再输入数据

尽量少加非输入控制符，比如 scanf("%d\n", &i);

- 给多个变量赋值
- 编写代码对用户的非法输入做适当的处理

```
char ch;  
while ((ch = getchar()) != '\n')  
    continue;
```

scanf一个字符——》scanf (" %c", &ch), %c前必须加一个空格, 即【空白符】