

输出位数前加0

`#include<iomanip>`这个头文件提供了一种输出方式，可以输出时在前面加零

```
1. #include<iostream>
2. #include<iomanip>
3. using namespace std;
4. int main()
5. {
6.     int a=10,b=1;
7.     cout<<setfill('0')<<setw(8)<<a<<endl;
8.     cout<<setfill('0')<<setw(8)<<b<<endl;
9. }
```

输出8位数，自动用0填满8位，显示的时候还不错。

如果你要显示一个10位的整数，前面显示0，

`int n;`

`printf("%010d", n);`

即可。其中10表示宽度，0表示前面添充0。

///
1. 补位，域宽问题

```
1. #include<iostream>
2. using namespace std;
3. int main()
4. {
5.     int num;
6.     num=5;
7.     cout.fill('0');//设置填充字符
8.     cout.width(5);//设置域宽
9.     cout<<num<<endl;
10.    return 0;
11. }
```

结果：00005

///
2. 小数点以及精度问题

```
1. #include <iostream>
2. #include <iomanip> ///头文件很重要
3. using namespace std;
```

```

4. |
5. const double value = 12.3456789;
6. int main()
7. {
8.     cout.precision(5); //输出 5 位数字 超出原数字位数会 输出原数字 (会四舍五
    入)
9. |
10.    cout << value << endl;
11. |
12.    cout.precision(6);
13.    cout.setf(ios::fixed); //输出 6 位小数 超出原数字小数位数会 补0 (会四舍
    五入)
14. |
15.    cout << value << endl;
16. |
17.    cout<<setiosflags(ios::fixed)<<setprecision(6)<<value<<endl; //同上
18. |
19.    return 0;
20. }

```

结果：

12. 345

12. 345679 （四舍五入）

12. 345679 （四舍五入）