

```
//一：隐式类型转换：系统自动进行，不需要程序开发人员介入。  
//int m = 3 + 45.6; //48.600000000 ，把小数部分截掉，也属于隐式类型转换的一种行为。  
//double n = 3 + 45.6;
```

```
//二：显式类型转换（强制类型转换）  
//int k = 5 % 3.2; //语法错  
//int k = 5 % (int)3.2; //OK, C语言风格的强制类型转换  
//int k = 5 % int(3.2); //函数风格的强制类型转换(C语言风格的强制类型转换)  
  
//C++强制类型转换分为4种：我们现在写程序应该使用C++风格的强制类型转换。  
//这4种强制类型转换，分别用于不同的目的，每一个都有一个不同的名字。提供4种类的目的：提供更丰富的含义和功能，更好的类型检查  
//a) static_cast  
//b) dynamic_cast  
//c) const_cast  
//d) reinterpret_cast  
//这四个强制类型转换都被称呼为 “命名的强制类型转换（因为他们每一个都有一个名字并且名字各不相同）；  
  
//通用形式  
//强制类型转换名<type>(express):  
//强制类型转换名，是上边的a, b, c, d四个名字之一。  
//type: 转换的目标类型  
//express: 你要转换的值（你要转换的是谁）；  
//（2.1）static_cast：静态转换，大家就理解成“正常转换”，编译的时候就会进行类型转换的检查。  
//代码中要保证转换的安全性和正确性，static_cast含义跟C语言中的强制类型转换这种感觉差不多。  
//C风格的强制类型转换，以及编译器能够进行的隐式类型转换，都可以用static_cast类显式完成。  
//可用于：  
//a) 相关类型转换，比如整型和实型之间的转换：  
double f = 100.34f;  
int i = (int)f; //C风格的  
int i2 = static_cast<int>(f); //b) 后续学习类的时候子类转成父类类型（继承关系），也能用这个static_cast赚。
```