



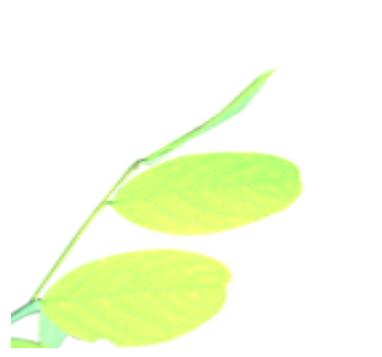
Introduction to C++ (Season 1)

Unit 4: Objects and Classes

第4单元：物以类聚－对象和类

Section 3 : More on Creating Objects

第3节：关于创建对象的更多细节



Naming Objects and Classes (为对象和类命名)

- ❖ When you declare a custom class, capitalize the first letter of each word in a class name; (声明一个自定义的类时，类名中的单词要首字母大写)
 - for example, the class names Circle, Rectangle, and Desk.
- ❖ The class names in the C++ library are named *in lowercase*. (C++标准库中的类名是小写的)
- ❖ The objects are named like variables. (对象的命名方式与变量类似)

Class is a Type (类是一种数据类型)

- ❖ use **primitive data types** to define *variables*. (用基本数据类型定义变量)
- ❖ use **class names** to declare *objects*. In this sense, a class is also a data type. (用类名定义对象)

```
Circle circle1;           // class → primitive data type  
Circle circle2(5.5);  
  
int x(4);                 // primitive data type → class  
double y(3.3);
```

3. Names representing types must be in mixed case starting with upper case.

3. 代表类型的名字必须首字母大写并且其它字母大小写混合

例如: Line, SavingsAccount

Memberwise Copy (成员拷贝)

❖ How to copy the contents from one object to the other?(如何将一个对象的内容拷贝给另外一个对象)

- use the assignment operator(使用赋值运算符): `=`
- By default, *each data field* of one object is copied to its counterpart in the other object. (默认情况下, 对象中的每个数据域都被拷贝到另一对象的对应部分)

❖ Example: circle2 = circle1;

- copies the radius in circle1 to circle2.
- After the copy, circle1 and circle2 are still two different objects, but with the same radius.

Anonymous Object (匿名对象)

- ❖ Occasionally, you may create an object and *use it only once*. (有时需要创建一个只用一次的对象)
- ❖ In this case, you don't have to name the object. (此时, 无需给对象命名) Such objects are called anonymous objects. (这种对象叫做匿名对象)
- ❖ The syntax is

```
ClassName(); //using the no-arg  
ClassName(arguments); //using the constructor with arguments
```

```
int main() {  
    Circle circle1, circle2;  
    circle1 = Circle();  
    circle2 = Circle(5);  
  
    cout << "Area is " << Circle().getArea() << endl;  
    cout << "Area is " << Circle(5).getArea() << endl;  
    return 0;  
}
```

Class Replaces struct

- ❖ The C language has the struct type for representing records. (C语言中使用结构体类型来表示一组数据)
- ❖ In C++, *class* has replaced *struct* (在C++中，结构体已被类取代)

```
struct Student
```

```
{  
    int id;  
    char firstName[30];  
    char mi;  
    char lastName[30];  
};
```

default access
control is "public"

(a)

```
class Student
```

```
{  
    public:  
    int id;  
    char firstName[30];  
    char mi;  
    char lastName[30];  
};
```

default access
control is "private"

(b)