



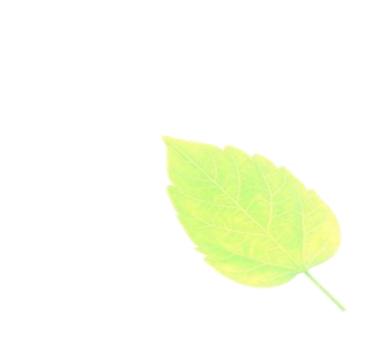
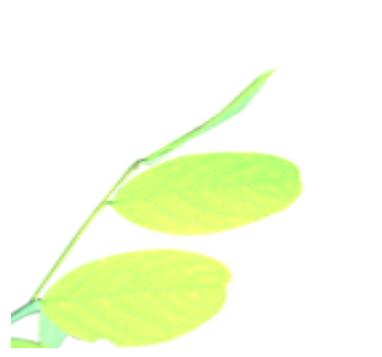
Introduction to C++ (Season 1)

Unit 4: Objects and Classes

第4单元：物以类聚－对象和类

Section 8 : The Scope of Variables & "this" pointer

第8节：变量作用域与this指针



The Scope of Variables – Review (变量作用域-回顾)

❖ C语言的“函数”章节，介绍了3种变量作用域

❖ Global variables (全局变量)

- are declared outside all functions and are accessible to all functions in its scope. (在所有函数外面声明并在其作用域内可被所有函数访问)
- The scope starts from its declaration and *continues to the end of the program*. (作用域起于声明，止于程序结束)

❖ Local variables (局部变量)

- are defined inside functions. (在函数内定义)
- The scope starts from its declaration and *continues to the end of the block that contains the variable*. (作用域起于声明，止于包含该变量的块尾)

❖ Static local variables (静态局部变量)

- permanently stored in the program.
- can be used in the next call of the function

The Scope of Data Fields in Class (类中数据域的作用域)

❖ The data fields

- are declared as variables inside class (被定义为变量形式)
- are accessible to all constructors and functions in the class.(可被类内所有函数访问)

❖ Data fields and functions can be **declared in any order** in a class. (数据域与函数可按任意顺序声明)

```
class Circle
{
public:
    Circle();
    Circle(double);
    double getArea();
    double getRadius();
    void setRadius(double);

private:
    double radius;
};
```

(a)

```
class Circle
{
public:
    Circle();
    Circle(double);

private:
    double radius;

public:
    double getArea();
    double getRadius();
    void setRadius(double);
};
```

(b)

```
class Circle
{
private:
    double radius;

public:
    double getArea();
    double getRadius();
    void setRadius(double);

public:
    Circle();
    Circle(double);
};
```

(c)

Hidden by same name (同名屏蔽)

- ❖ If a local variable has the same name as a data field: (若成员函数中的局部变量与某数据域同名)
 - the local variable takes precedence (局部变量优先级高)
 - the data field with the same name is hidden. (同名数据域在函数中被屏蔽)



Note

为避免混淆，不要在类中多次声明同名变量，除了函数参数

The this Pointer

❖ How do you reference a class's hidden data field in a function? (如何在函数内访问类中被屏蔽的数据域)

❖ *this* keyword

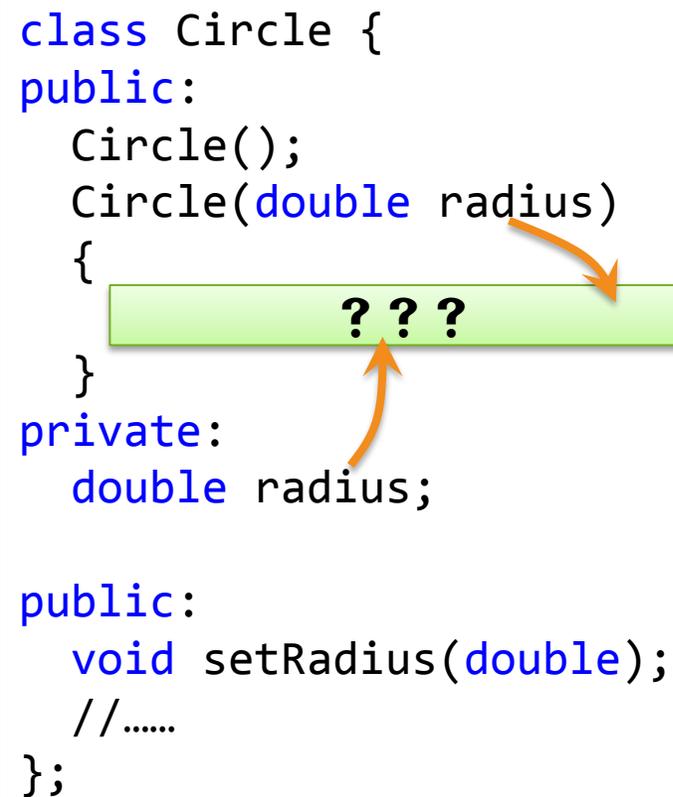
- a special built-in pointer (特殊的内建指针)
- references to the calling object. (引用当前函数的调用对象)

Screen类中有两个函数move()和set(),如何设计这两个函数的返回值,才能使得下面语句是合法的?

```
myScreen.move(4,0).set('#');
```

```
class Circle {
public:
    Circle();
    Circle(double radius)
    {
        ???
    }
private:
    double radius;

public:
    void setRadius(double);
    //.....
};
```



Simple way to avoid name hidden (避免重名屏蔽的简单方法)

```
class Circle {  
public:  
    Circle();  
    Circle(double radius)  
    {  
        //this->radius = radius;  
        radius_ = radius;  
    }  
private:  
    double radius_;  
  
public:  
    void setRadius(double);  
    //.....  
};
```

11. Private class variables should have underscore suffix.

11. 私有类成员变量名应有下划线后缀

例:

```
class SomeClass {  
private:  
    int length_;  
}
```

代码中容易区分类成员变量及函数局部变量

也有些规范中使用下划线前缀。但使用后缀让名字可读性更好