



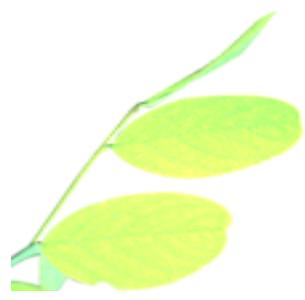
# Introduction to C++ (Season 1)

## Unit 4: Objects and Classes

### 第4单元：物以类聚－对象和类

#### Section 1 : Concepts of Class

##### 第1节：类的概念

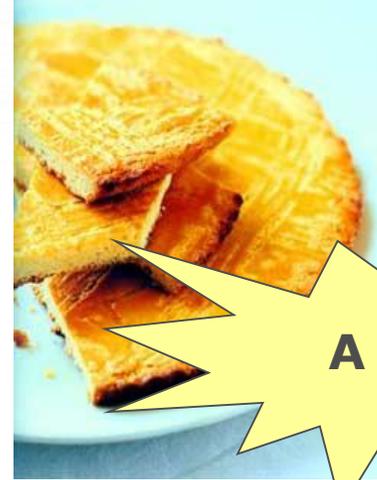


# OO Programming Concepts (面向对象编程的概念)

- ❖ Object-Oriented Programming (**OOP**) involves programming using objects.  
(OOP利用对象进行程序设计)
- ❖ An *object* represents an entity in the real world that can be distinctly identified.  
(一个对象表示现实世界中一个独一无二的实体)
  - For example, a student, a desk, a circle, a button, and even a loan can all be viewed as objects.

# Features of OO (面向对象的特征)

- ❖ Abstraction (抽象)
- ❖ Polymorphism (多态)
- ❖ Inheritance (继承)
- ❖ Encapsulation (封装)



**A PIE**

# What does an object consist of? (对象由什么构成)

- ❖ An object has a unique identity, **state**, and **behaviors**. (对象具有唯一的标识、状态和行为)
  - The *state* of an object consists of a set of ***data fields*** (also known as ***properties***) with their current values. (对象状态由数据域(也称为“属性”)及其当前值构成)
  - The *behavior* of an object is defined by a set of ***functions***. (对象的行为由一组函数定义)

# How to define Objects (如何定义对象)

- ❖ Objects of the same type are defined using a common class. (同类型对象用一个通用的类来定义)
- ❖ An object is an *instance* of a class (对象是类的实例)

```
int ia, ib; //ia, ib: 整形变量  
struct S { int m; int n; }; S sa, sb; //sa, sb: 结构变量
```

```
class C { int p; int f(); }; C ca, cb; //ca: Object
```

## 主料

千层酥皮面团 xx克

## 辅料

核桃碎 xx克      蛋液 适量

## 馅料用料

苹果 2个      细砂糖 xx克

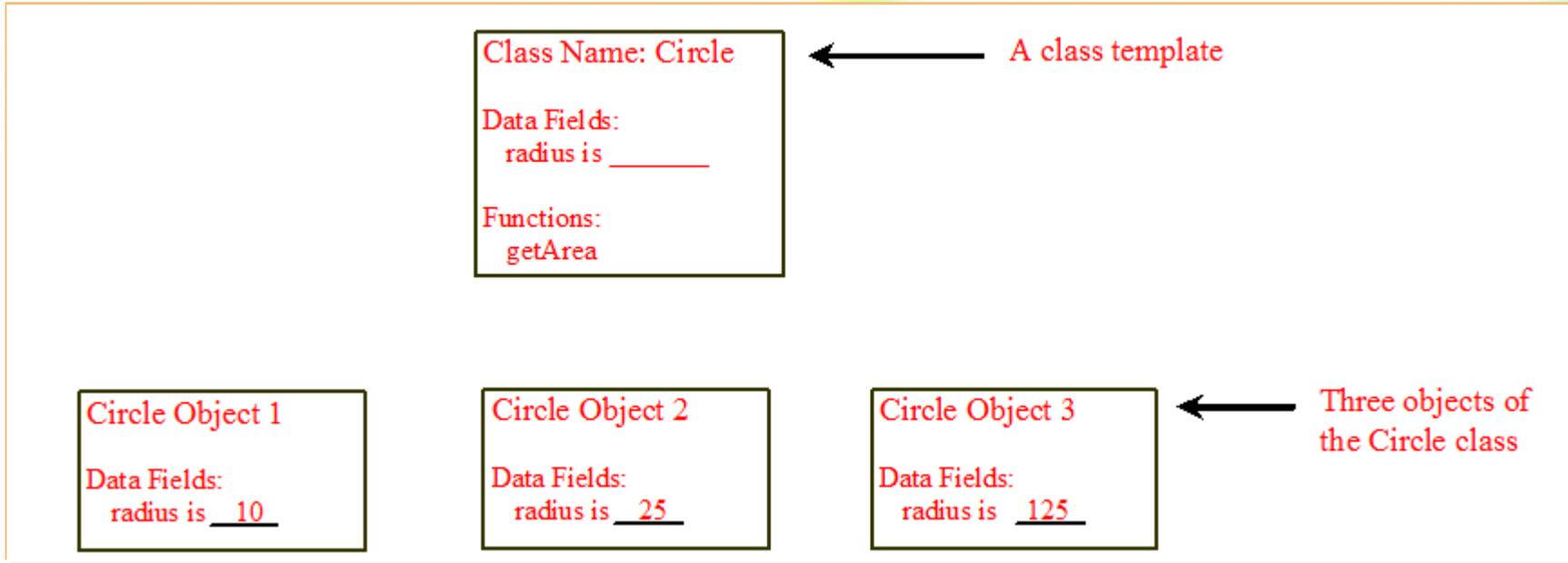
柠檬汁 10ml      酒 5ml



对象 ↔ 类

苹果派 ↔ 苹果派配方

# Examples of Classes and Objects(类与对象的例子)



# Classes

- ❖ A *class* uses *variables*( 变量) to *define data fields*( 定义数据域) and *functions*( 函数) to *define behaviors*( 定义行为).
- ❖ Additionally, a class provides a special type of functions, known as *constructors*, which are invoked to construct objects from the class. (类中有一种特殊的“构造函数”，在创建对象时被自动调用)

# Classes – an example

```
class Circle
{
public:
    // The radius of this circle
    double radius;

    // Construct a circle object
    Circle()
    {
        radius = 1;
    }

    // Construct a circle object
    Circle(double newRadius)
    {
        radius = newRadius;
    }

    // Return the area of this circle
    double getArea()
    {
        return radius * radius * 3.14159;
    }
};
```

Data field

Constructors

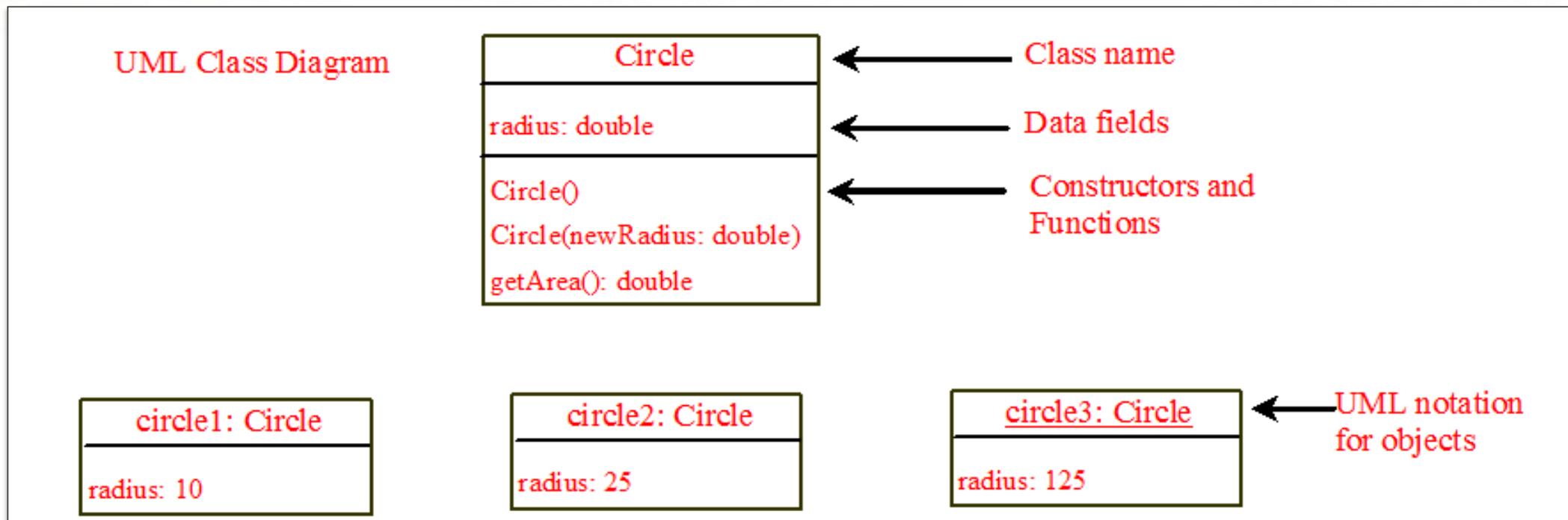
Function

# UML Class Diagram (UML类图)

❖ UML: Unified Modeling Language(统一建模语言)

## Tools:

1. StarUML (Open Source)
2. Rational Rose 7.0
3. MS Visio



## C++ Data Types

