

5.3 类模板_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 5.3 类模板涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

3. 类模板

一、定义

对一批仅仅 ** 成员数据类型 (成员函数和成员数据)** 不同的类抽象，只要为这一批类所组成的整个类家族创建一个类模板，给出一套程序代码，就可以用来生成多种具体的类。

二、语法格式

1. 类模板定义格式

```
template <类型参数表>

class 类模板名

{

    成员函数

    成员变量

};
```

类型参数表里面声明类中需要用到的抽象类型, 如: class T1, typename T1, ...

```
template <typename T>

class SeqList

{

    public:

        SeqList(int size);

        int addData(const T &data);

        void show(void) const;

    private:

        T *buf;

        int size;

        int index;

};
```

2. 成员函数在类外定义格式

类模板中成员函数在类外实现时，需要加上类型参数列表

```
template<类型参数表>

返回值类型 类模板名<类型参数名列表>::成员函数名(函数参数)

{
    ...
}

template <typename T> SeqList<T>::SeqList(int size)

{
    buf = new T[size];

    index = 0;

    this->size = size;
}
```

注意: 类模板成员在类外定义的时候，格式比较麻烦，不推荐在类外定义，推荐直接在类内实现

```
template <typename T>

class SeqList

{
public:
    SeqList(int size);

    int addData(const T &data)
    {
        if(index >= size){
            cout << "The seqlist is full" << endl;

            return -1;
        }

        buf[index ++] = data;

        return 0;
    }

    void show(void) const;

private:
    T *buf;

    int size;

    int index;
};
```

3. 对象创建格式

类模板没有自动类型推导的使用方式, 创建对象的时候, 必须指定具体类型

类模板名<具体类型参数表> 对象名(参数)

```
SeqList<string> list(10);
```

```
list.addData("ABC");
```

```
list.show();
```

三、类模板与继承

- 子类继承的父类是一个类模板时, 子类在声明的时候, 要指定出父类中 T 的类型
- 若想灵活指出父类中 T 的类型, 子类也需变成模板

```
template<class T>
```

```
class Base{
```

```
    T m;
```

```
};
```

```
//错误, 未指定类型
```

```
class Son:public Base{
```

```
}
```

```
//必须指定类型
```

```
class Son:public Base<int>{
```

```
}
```

```
//类模板继承类模板, 可用T2指定父类中的T类型
```

```
template<class T1,class T2>
```

```
class Son:public Base<T2>{
```

```
};
```

四、任务

使用类模板描述一个顺序表, 实现顺序表的创建、插入、删除、遍历操作

- 划线

- 写笔记

学习要认真，笔记应当先



公开笔记 0/1000 提交



Sunny_SunshineX

删除 编辑

全文完

本文由 简悦 SimpRead 优化，用以提升阅读体验

使用了 全新的简悦词法分析引擎 beta，点击查看详细说明

