

第8章 取其精华 发挥优势—继承

1、模拟人的行走、听、说、写

2、为什么需要继承

3、派生类的定义

4、基类与派生类

5、三种继承方式

6、派生类的构造与析构函数

7、点、圆、圆柱体继承设计

8、从U盘到MP3继承设计

派生类的定义

```
class 派生类名: 继承方式 基类名1, 继承方式 基类名2, ..., 继承方式 基类名n
{
    派生类增加的成员声明;
};
```

```
class sing_star:public person
{
    float salary;    //薪水
public:
    sing_star();    //无参构造函数
    sing_star(char *n,char s,char *p,int w,int h,float s1); //有参构造函数
    void change_data(char *n,char s,char *p,int w,int h,float s1); //修改数据
    void playing(char *ps); //演唱歌曲
    void print();        //输出歌星属性值
};
```

人类定义中增加有关日期时间的属性

👉 一个人一生中有很多重要日期和时刻

👉 上大学日期、入党日期

👉 出生时间

基类：日期类的定义

```
class Date
```

```
{
```

```
protected:
```

```
    int year
```

```
    int month;
```

```
    int day;
```

```
public:
```

```
    Date() { year = 1900; month = day = 1; }
```

```
    Date(int yy,int mm,int dd){                init(yy,mm,dd);                };
```

```
    void init(int,int,int );
```

```
    void print_ymd();
```

```
    void print_mdy();
```

```
};
```

基类：时间类的定义

```
class Time  
{  
    protected:  
        int hour;  
        int miniter;  
        int second;  
    public:  
        Time() { hour = miniter = second = 0; }  
        Time(int h,int m,int s){init(h,m,s);}  
        void init(int,int,int );  
        void print_time();  
};
```

派生类：人类的定义

```
class person:public Date,public Time
{
    //注意包含了基类的出身日期和出身时间
    char name[20];
    char sex;
    char pid[19];
    int weight;
    int high;
public:
    person();
    person(char *n,char s,char *p,int w,int h,int hr,int mr,int sd);
    void change_data(char *n,char s,char *p,int w,int h,int hr,int mr,int sd);
    void walking(int k);
    void hearing(char *sentence);
    void speak();
    void writing();
    void ShowMe();
};
```

派生类：人类的构造函数定义

```
person::person()  
{  
    name=new char[strlen("XXXXXX")+1];  
    strcpy(name,"XXXXXX");  
    strcpy(pid,"XXXXXXXXXXXXXXXXXXXXX");  
    sex='X';  
    weight=0;  
    high=0;  
    year=1900;  
    month=day=1;  
    hour=miniter=second=0;  
}  
person::person(char *n,char s,char *p,int w,int hh,int hr,int mr,int sd)  
{  
    change_data(n,s,p,w,hh,hr,mr,sd);  
}
```

派生类：人类的修改数据函数定义

```
void person::change_data(char *n,char s,char *p,int w,int hh,int hr,int mr,int sd)
{
    name=new char[strlen(n)];
    strcpy(name,n);
    strcpy(pid,p);
    sex=s;
    weight=w;
    high=hh;
    char temp[5];          //通过身份证号码产生出身日期
    strncpy(temp,p+6,4);
    year=atoi(temp);
    strncpy(temp,p+10,2);
    temp[2]='\0';
    month=atoi(temp);
    strncpy(temp,p+12,2);
    temp[2]='\0';
    day=atoi(temp);
    hour=hr;
    miniter=mr;
    second=sd;
}
```


派生类：人类的测试主函数

```
int main()
```

```
{
```

```
    //创建对象
```

```
    person Jack("James Chen",'M',"610103198901062493",160,180,23,34,35);
```

```
    Jack.print();                                //输出人的属性值
```

```
    system("pause");
```

```
    Jack.walking(10,4);                            //行走10步，1/4秒走一步
```

```
    Jack.hearing("You are simple");    //听英文句子
```

```
    Jack.speek(1006);                            //说出整数num的英文句子
```

```
    cout<<endl;
```

```
    Jack.writing();                                //书写汉字“曲”
```

```
    return 0;
```

```
}
```