# 第8周 实验报告册上完成的内容

1、编写一个学生和教师数据输入和显示程序，学生数据有编号、姓名、班号和成绩，教师数据有编号、姓名、职称和部门，要求将编号、姓名输入和显示设计成一个类person，并作为学生数据操作类student和教师数据操作类teacher的基类。

/\*1、编写一个学生和教师数据输入和显示程序，学生数据有编号

、姓名、班号和成绩，教师数据有编号、姓名、职称和部门，要求

将编号、姓名输入和显示设计成一个类person，并作为学生数据操

作类student和教师数据操作类teacher的基类。\*/

#include<iostream>

using namespace std;

#include<string>

class Person

{

private:

string name,ID;

public:

Person(string n,string i)

{

name=n;ID=i;

}

virtual void show()

{

cout<<"姓名："<<name<<"\t"<<"编号："<<ID<<endl;

}

};

class Student:public Person

{

private:

string room;double score;

public:

Student(string n,string i,string r,double s):Person(n,i)

{

room=r;score=s;

}

void show()

{

Person::show();

cout<<"班号："<<room<<"\t"<<"成绩："<<score<<endl;

}

};

class Teacher:public Person

{

private:

string possesion,apect;

public:

Teacher(string n,string i,string p,string a):Person(n,i)

{

possesion=p;apect=a;

}

void show()

{

Person::show();

cout<<"职称："<<possesion<<"\t"<<"部门："<<apect<<endl;

}

};

void test01()

{

Student s("张雨佳","A19210291","微机2107",100);

s.show();

Teacher t("王路明","A19210292","工程师","土木工程");

t.show();

}

int main()

{

test01();

return 0;

}

2、分别定义Teacher(教师)类和Cadre(干部)类，采用多重继承方式由这两个类派生出新类Teacher\_Cadre(教师兼干部)。要求:

① 在两个基类中都包含姓名、年龄、性别、地址、电话等数据成员。

② 在Teacher类中还包含数据成员title(职称)，在Cadre类中还包含数据成员post(职务)， 在Teacher\_Cadre类中还包含数据成员wages(工资)。

③ 对两个基类中的姓名、年龄、性别、地址、电话等数据成员用相同的名字，在引用这些数据成员时，指定作用域。

④ 在类体中声明成员函数，在类外定义成员函数。

⑤ 在派生类Teacher\_Cadre的成员函数show中调用Teacher类中的display函数，输出姓名、年龄、性别、职称、地址、电话，然后再用cout语句输出职务与工资。

#include<iostream>

using namespace std;

#include<string>

class Person

{

protected:

string name,gender,address,number;int age;

public:

Person(string n,string g,string a,string nu,int ag)

{

name=n;gender=g;address=a;number=nu;age=ag;

}

};

class Teacher:public Person

{

protected:

string title;

public:

Teacher(string n,string g,string a,string nu,int ag,string t);

void display();

};

Teacher::Teacher(string n,string g,string a,string nu,int ag,string t):Person(n,g,a,nu,ag)

{

title=t;

}

void Teacher::display()

{

cout<<"姓名："<<name<<" "<<"年龄："<<age<<" "<<"性别："<<gender<<" "

<<"职称："<<title<<" "<<"地址："<<address<<" "<<"电话："<<number<<" "<<endl;

}

class Cadre:public Person

{

protected:

string post;

public:

Cadre(string n,string g,string a,string nu,int ag,string p);

};

Cadre::Cadre(string n,string g,string a,string nu,int ag,string p):Person(n,g,a,nu,ag)

{

post=p;

}

class Teacher\_Cadre:public Teacher,public Cadre

{

private:

string wages;

public:

Teacher\_Cadre(string n,string g,string a,string nu,int ag,string t,string p,string w);

void show()

{

Teacher::display();

cout<<"职务："<<post<<"\t"<<"工资："<<wages<<endl;

}

};

Teacher\_Cadre::Teacher\_Cadre(string n,string g,string a,string nu,int ag,string t,string p,string w):Teacher(n,g,a,nu,ag,t),Cadre(n,g,a,nu,ag,p)

{

wages=w;

}

void test01()

{

Teacher\_Cadre tc("张雨佳","女","香坊区","150450到了楼下喊我名",18,"商业精英","具有超强大脑的女人","一百万");

tc.show();

}

int main()

{

test01();

return 0;

}