

C++语言基础

迂者 - 贺利坚

<http://blog.csdn.net/sxhelijian/>

<http://edu.csdn.net>





本节主题：

对象成员的引用

对象成员引用的方法

- ❏ 私有(private)成员仅能在成员函数中引用，此时，直接使用成员名即可；
- ❏ 公共(public)数据成员可以在类外访问，为封装和信息隐藏，不提倡这样做；
- ❏ 公共(public)成员函数可以在类外访问，这是对象与外界进行消息传递的途径；
- ❏ 访问对象中成员的3种方法：
 - ❏ 通过对象名和成员运算符访问对象中的成员；
 - ❏ 通过指向对象的指针访问对象中的成员；
 - ❏ 通过对象的引用变量访问对象中的成员。

```
class Student
{
public:
    void display( );
private:
    int num;
    char name[20];
    char sex;
};

void Student::display( )
{... ..}

main()
{ ... .. }
```

通过对象名和成员运算符访问对象中的成员

访问对象中成员的一般形式

对象名.成员名

例：

t1.set_time();

若hour也为公用型，可以

t1.hour=24;

cout<<t2.hour;

注意：必须指定对象名

不能set_time()

```
class Time
{
public:
    void set_time( );
    void show_time( );
private:
    int hour;
    int minute;
    int sec;
};

int main( )
{
    Time t1;
    t1.set_time( );
    t1.show_time( );
    Time t2;
    t2.set_time( );
    t2.show_time( );
    return 0;
}
```

通过指向对象的指针访问对象中的成员

❏ 定义指向对象的指针

```
Time t1, *p;
```

```
p=&t1;
```

❏ 通过指针访问对象中的成员

```
p->show_time( );
```

❏ 若hour也为公用型，可以

```
p->hour=24;
```

```
cout<<p->hour;
```

```
class Time
{
public:
    void set_time( );
    void show_time( );
private:
    int hour;
    int minute;
    int sec;
};

int main( )
{
    Time t1, *p;
    p=&t1;
    t1.set_time( );
    (*p).show_time( );
    p->show_time( );
    return 0;
}
```

通过对象的引用变量访问对象中的成员

- 引用变量共占同一段存储单元的，实际上它们是同一个对象，只是用不同的名字表示而已。
- 定义对象的引用

Time t1;

Time &t2=t1;

```
class Time
{
public:
    void set_time( );
    void show_time( );
private:
    int hour;
    int minute;
    int sec;
};

int main( )
{
    Time t1;
    Time &t2=t1;
    t1.set_time( );
    t2.show_time( );
}
```

访问私有数据成员的常用方法

```
class Test
{
private:
    int x, y;
public:
    void setX(int a) {x=a;}
    void setY(int b) {y=b;}
    void printXY(void)
        {cout<<"x="<<x<<"\t"<<"y="<<y<<endl;}
};

int main()
{
    Test p1,p2;
    p1.setX(3);
    p1.setY(5);
    p1.printXY( );
}
```

1、
通过
公共函数
访问
私有成员

```
class Test
{
private:
    int x,y;
public:
    void setX(int a) {x=a;}
    void setY(int b) {y=b;}
    int getX(void) {return x;} //返回x值
    int getY(void) {return y;} //返回y值
};

int main()
{
    Test p1,p2;
    p1.setX(3);p1.setY(5);
    int a,b;
    a=p1.getX( ); b=p1.getY();
    cout<<a<<"\t"<<b<<endl;
}
```

2、(惯例)
利用
set和get函数
访问
私有数据成员

利用指针将私有数据成员的值提取到类外

```
class Test
{
private:
    int x,y;
public:
    void setX(int a) {x=a;}
    void setY(int b) {y=b;}
    void getXY(int *px, int *py)
        {*px=x; *py=y;}//提取x,y值
};

int main()
{
    Test p1;
    p1.setX(3);p1.setY(5);
    int a,b;
    p1.getXY(&a,&b); //将 a=x, b=y
    cout<<a<<'t'<<b<<endl;
}
```


利用引用将私有数据成员的值提取到类外

```
class Test
{
private:
    int x,y;
public:
    void setX(int a) {x=a;}
    void setY(int b) {y=b;}
    void getXY(int &px, int &py) //引用
        {px=x;py=y;} //提取x,y值
};

int main()
{
    Test p1;
    p1.setX(3);p1.setY(5);
    int a,b;
    p1.getXY(a, b); //将 a=x, b=y
    cout<<a<<'\\t'<<b<<endl;
}
```

THANKS

本课程由 迂者-贺利坚 提供

CSDN网站：www.csdn.net
企业服务：<http://ems.csdn.net/>
人才服务：<http://job.csdn.net/>
CTO俱乐部：<http://cto.csdn.net/>
高校俱乐部：<http://student.csdn.net/>
程序员杂志：<http://programmer.csdn.net/>

CODE平台：<https://code.csdn.net/>
项目外包：<http://www.csto.com/>
CSDN博客：<http://blog.csdn.net/>
CSDN论坛：<http://bbs.csdn.net/>
CSDN下载：<http://download.csdn.net/>

