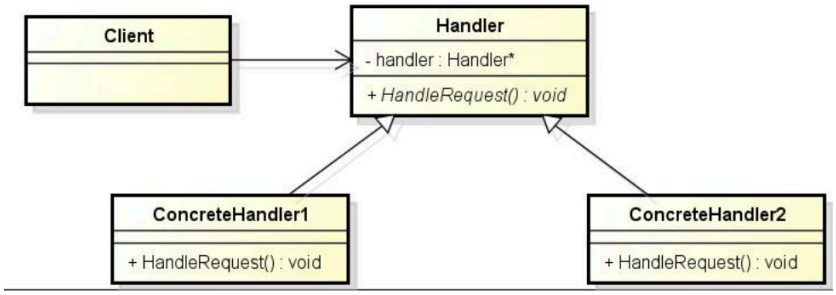
定义:使多个对象都有机会处理请求，从而避免请求的发送者和接收者之间的耦合关系。将这些对象连成一条链，并沿着这条链传递该请求，直到有一个对象处理它为止。

抽象处理者(Handler)角色：定义出一个处理请求的接口。如果需要，接口可以定义出一个方法，以设定和返回对下家的引用。这个角色通常由一个抽象类或接口实现。

具体处理者(ConcreteHandler)角色：具体处理者接到请求后，可以选择将请求处理掉，或者将请求传给下家。由于具体处理者持有对下家的引用，因此，如果需要，具体处理者可以访问下家。




```cpp
#include <iostream>
#include <string>
using namespace std;
class Request{
public:
    int m_nNumber;
};
class Manager{
protected:
    Manager *m_pManager;
    string m_strName;
public:
    Manager(string name){
        m_strName = name;
    }
    virtual ~Manager(){}
    void SetHandler(Manager *pM){
        m_pManager = pM; //设置处理人
    }
    virtual void GetRequest(Request *pRequest) = 0;
};
class ManagerC : public Manager{
public:
    ManagerC(string name):Manager(name){}
    void GetRequest(Request * pRequest){
        if(pRequest->m_nNumber > 0 && pRequest->m_nNumber < 1000 ){
            cout << m_strName << "处理了: " << pRequest->m_nNumber << endl;
        }else{
            m_pManager->GetRequest(pRequest);
        }
    }
};
class ManagerB : public Manager{
public:
```

```
    ManagerB(string name):Manager(name){}
    void GetRequest(Request * pRequest){
        if(pRequest->m_nNumber > 1000 && pRequest->m_nNumber < 5000 ){
            cout << m_strName << "处理了: " << pRequest->m_nNumber << endl;
        }else{
            m_pManager->GetRequest(pRequest);
        }
    }
};
class ManagerA : public Manager{
public:
    ManagerA(string name):Manager(name){}
    void GetRequest(Request * pRequest){
        if(pRequest->m_nNumber >= 5000){
            cout << m_strName << "处理了: " << pRequest->m_nNumber << endl;
        }
    }
};
int main(){
    Manager* pMC = new ManagerC("主管");
    Manager* pMB = new ManagerB("经理");
    Manager* pMA = new ManagerA("总监");
    pMC->SetHandler(pMB);
    pMB->SetHandler(pMA);
    Request *pR = new Request();
    pR->m_nNumber = 100;
    pMC->GetRequest(pR);
    pR->m_nNumber = 1500;
    pMC->GetRequest(pR);
    pR->m_nNumber = 6500;
    pMC->GetRequest(pR);
    delete pMC;delete pMB; delete pMA; delete pR;
    return 0;
}
```

```
[192:DesignPattnsStudy weishichun$ g++ -o ChianOfResponsible_1.out ChianOfRespons]
ible_1.cpp
[192:DesignPattnsStudy weishichun$ ./ChianOfResponsible_1.out                    ]
主管处理了: 100
经理处理了: 1500
总监处理了: 6500
192:DesignPattnsStudy weishichun$
```