

virt1.cpp

```
#include <iostream>
using namespace std;

class base {
    int content;
public:
    base(int n=1){ content=n; }
    virtual void print(){ cout<<"base: " <<content<<endl; }
};

class derived1: public base {
    int content;
public:
    derived1(int n=1){ content=n; }
    virtual void print(){ cout<<"derived1: " <<content<<endl; }
};

class derived2: public base {
    int content;
public:
    derived2(int n=1){ content=n; }
    virtual void print(){ cout<<"derived2: " <<content<<endl; }
};

class derived3: public base {
    int content;
public:
    derived3(int n=1){ content=n; }
};

class run {
public: static void printinfo(base &b){ b.print(); }
};

void main(){
    base b;
    derived1 d1(11);
    derived2 d2(22);
    derived3 d3(33);

    run::printinfo(b);      // base: 1
    run::printinfo(d1);     // derived1: 11
    run::printinfo(d2);     // derived2: 22
    run::printinfo(d3);     // base: 1
}
```