

1. **Create a virtual function in your base class. Override the virtual function in your derived classes?**

```
#include<iostream>

using namespace std;

class bas
{
public:
    virtual void print ()
    { cout<< "base class is printed" <<endl; }

    void show ()
    { cout<< "base class is shown" <<endl; }
};

class der:public bas
{
public:
    void print ()
    { cout<< "derived class is printed" <<endl; }

    void show ()
    { cout<< "derived class is shown" <<endl; }
};

int main()
{
    bas *bptr;
```

```
    der d;  
    bptr = &d;  
    bptr->print();  
    bptr->show();  
}
```

## OUTPUT

# C++ shell

```
1  #include<iostream>  
2  using namespace std;  
3  
4  class bas  
5  {  
6  public:  
7      virtual void print ()  
8      { cout<< "base class is printed" <<endl; }  
9  
10     void show ()  
11     { cout<< "base class is shown" <<endl; }  
12 };  
13  
14 class der:public bas  
15 {  
16 public:  
17     void print ()  
18     { cout<< "derived class is printed" <<endl; }  
19  
20     void show ()  
21     { cout<< "derived class is shown" <<endl; }  
22 };  
23
```

Get URL

options compilation execution

```
derived class is printed  
base class is shown
```