```
#include<iostream>
using namespace std;
class sample
{
public:
virtual void example()=0;
void show()
{
cout<<"\nThis is sample abstract class";
}
};
class derived1:public sample
{
public:
void example()
{
cout<<"C++";
}
};
class derived2:public sample
{
public:
void example()
{
cout<<"\nC";
}
};
int main()
{
sample *ptr;//Base pointer
//sample obj;//Compile time error(Creating object of abstract class)
derived1 obj1;
derived2 obj2;
ptr=&obj1;//Currently pointer is pointing towards derived1
ptr->example();
ptr->show();
ptr=&obj2;//Currently pointer is pointing towards derived2
ptr->example();
ptr->show();
return 0;
}
```

```
#include <iostream>
using namespace std;
class shape
{
   public:
   virtual void area()=0;
};
```

```
class circle:public shape
{
   private:
   float radius,area1;
   public:
   void input()
   {
      cout<<"\nEnter radius:";
      cin>>radius;
   }
   void area()
   {
      area1=3.14*radius*radius;
      cout<<"\nArea of circle is:"<<area1;
   }
};
class rectangle:public shape
{
   private:
   int length,breadth,area2;
   public:
   void input()
   {
      cout<<"\nEnter length and breadth:";
      cin>>length>>breadth;
   }
   void area()
   {
      area2=length*breadth;
      cout<<"\nArea of rectangle is:"<<area2;
   }
};
int main()
{
   shape *bptr;
   circle obj1;
   bptr=&obj1;
   obj1.input();
   bptr->area();
   rectangle obj2;
   bptr=&obj2;
   obj2.input();
   bptr->area();
   return 0;
}
---------------------------------------
#include <iostream>
using namespace std;
class shape
```

```
{
   public:
   virtual void area()=0;
};
class circle:public shape
{
   private:
   float radius,area1;
   public:
   void input()
   {
      cout<<"\nEnter radius:";
      cin>>radius;
   }
   void area()
   {
      area1=3.14*radius*radius;
      cout<<"\nArea of circle is:"<<area1;
   }
};
class rectangle:public shape
{
   private:
   int length,breadth,area2;
   public:
   void input()
   {
      cout<<"\nEnter length and breadth:";
      cin>>length>>breadth;
   }
   void area()
   {
      area2=length*breadth;
      cout<<"\nArea of rectangle is:"<<area2;
   }
};
int main()
{
   shape *bptr;
   circle obj1;
   bptr=&obj1;
   //obj1.input();
   ((circle*)bptr)->input();
   bptr->area();
   rectangle obj2;
   bptr=&obj2;
   //obj2.input();
   ((rectangle*)bptr)->input();
   bptr->area();
```

```
    return 0;
}
```