

```

#include <iostream>
using namespace std;
/* calling of parent constructor and also implement
virtual function
*/

class Person{ // name,age
protected:
    string name;
    int age;

public:
    Person(string n,int a){
        name=n;
        age=a;
        cout<<"parameterized of parent\n";
    }
    Person(){
        name="abc";
        age=0;
        cout<<"default of parent\n";
    }

    virtual void getter(){
        cout<<"name:"<<name<<" age:"<<age<<endl;
    }

};

class Employee: public Person{ // name ,age,      eid,esal
private:
    int eid,esal;

public: // calling of parent constructor
    Employee(string n,int a,int id,int sal):Person(n,a){
        eid=id;
        esal=sal;
    }
}

```

```
        void  getter(){
            cout<<"name:"<<name<<" age:"<<age<<endl;
            cout<<"Eid:"<<eid<<" Esal:"<<esal<<endl;
        }

};

int  main(){
    Person *base;
    Employee e("ajay",23,101,34000);

    base=&e;
    base->getter();

}
```