**Experiment :-8**

Program To Implement Class Hierarchy

**ALGORITHM**

Step 1: Start

Step 2: Declare and define class “manager” with public functions “getdata” which prints the employee’s information, “setbonus” which prints the department and bonus and private variables “name”, “dept”, “emDid” and “bonus”.

Step 3: Declare and define class “prodmanager” inheriting from main class “manager” with public functions “managersprod” which prints the supplies, “display” which prints the required information of the products and bonus and define variable “noofsup”.

Step 4: Declare and define class “salesman” inheriting from main class “manager” with public functions “managersales” which prints the sales, “display” which prints the required information of the sales and bonus and define variable “noofsalesman”.

Step 5: call “getdata”, function through salesman class by inheriting from main class manager to print the information required. Similary for prodmanger class.

Step 6: Print the Number of Supplies by calling a member function “managerprod” from class “prodmanager”.

Step 7: Print the Number of Sales by calling a member function “managersales” from class “salesman”.

Step 6: call “setbonus”, function through salesman class by inheriting from main class manager to display the information required. Similary for prodmanger class.

Step 7: call “display”, function through salesman class by inheriting from main class manager to display the information required. Similary for prodmangerclass..

Step 8: Stop

**Program**

#include <iostream>

using namespace std;

class manager//base class

{

protected://access only for base class and derived class

string name;

string dept;

int emDId;

int bonus;

public:

void getdata()//getter function

{

cout<<"\n Enter name :";

cin>>name;

cout<<"\n Enter dept :";

cin>>dept;

cout<<"\n Enter ID :";

cin>>emDId;

}

void setbonus (int b)//setter function for bonus

{

cout<<"\n In setbonus()";

bonus = b;

cout<<"\t Dept :"<<dept<<"\t Bonus:"<<bonus;

}

};

class prodmanager:public manager //derived class prodmanager of base class manager

{

int noofsup;//number of supervisor

public:

void managerprod(int n)

{

cout <<" \n \n In manage prod ";

noofsup=n;

cout <<"\t No of sup :" << noofsup;

}

void display()

{

cout <<"\n Name :" << name;

cout <<"\n Sept :" << dept;

cout <<"\n No of Sup :" << noofsup;

cout <<"\n Bonus :" << bonus;

}

};

class salesman:public manager //derived class salesmanager of base class manager

{

int noofsalesman;//number of sales man for a sales manager

public:

void managersales(int n)

{

cout <<" \n \n In manage sale ";

noofsalesman=n;

cout <<"\t No of salesman :" << noofsalesman;

}

void display()

{

cout <<"\n Name :" << name;

cout <<"\n Dept :" << dept;

cout <<"\n No of salesman :" << noofsalesman;

cout <<"\n Bonus :" << bonus;

}

};

int main()

{

prodmanager p;// instantiate object of production manager

salesman s;// instantiate object of sales manager

cout<<"sales manager\n";

s.getdata();//input data for sales manager

cout<<"\nproduction manager\n";

p.getdata();//input data for production manager

p.managerprod(1000);//set number of supervisor for production manager

s.managersales(1000);//set number of salesman for sales manager

s.setbonus(5000);//set bonus for sales manager

p.setbonus(4000);//set bonus for production manager

s.display();//display sales manager

p.display();//display production manager

return 0;

}

**Sample Input Output**

