Practical-1

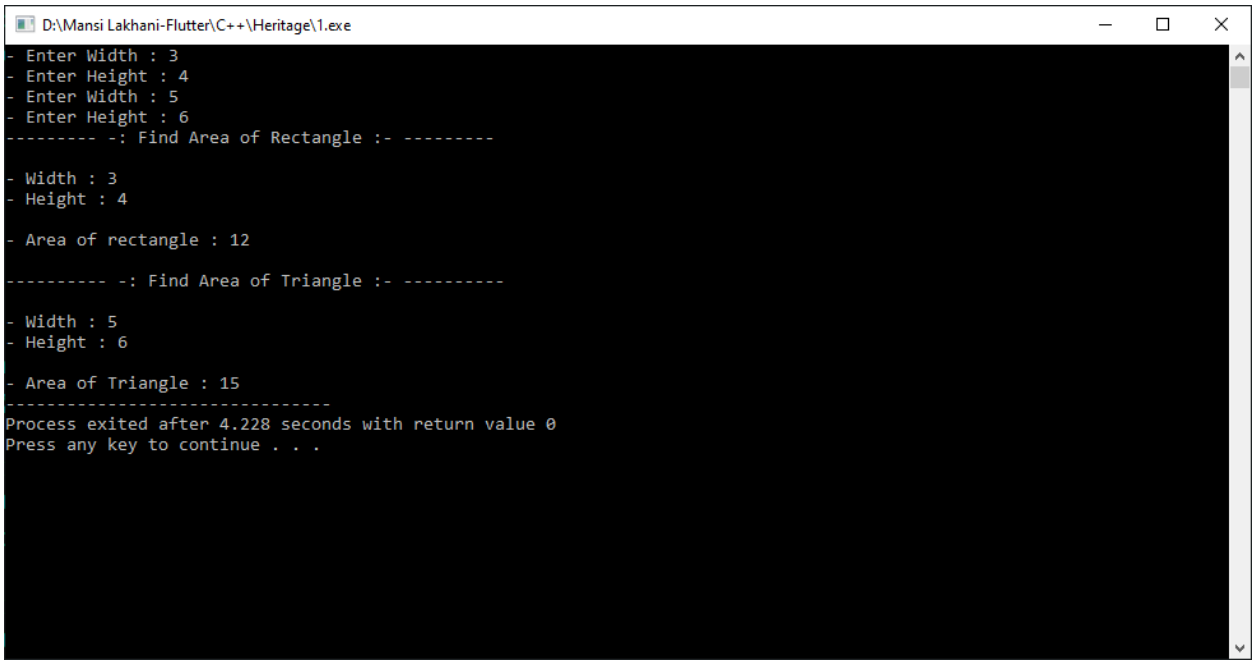
Aim: WAP that defines a shape class with a constructor that gives value to width and height. The define two sub-classes triangle and rectangle, that calculate the area of the shape area (). In the main, define two variables a triangle and a rectangle and then call the area() function in this two varibles.

Program:

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

class Shape
{
        public:
                int width;
                int height;
                int area;

        public:
                Shape()
                {
                        cout <<"- Enter Width : ";      cin >>this->width;
                        cout <<"- Enter Height : ";      cin >>this->height;
                }
                void getData()
                {
                        cout << "- Width : "<<this->width <<endl
                            << "- Height : "<<this->height <<endl;

                }
};

class Rectangle : public Shape
{
        public:
                void RectangleData()
```

```
		{
			area = width * height;
			cout <<endl<<"- Area of rectangle : "<<area;
		}
};
class Triangle : public Shape
{
	public:
		void TriangleData()
		{
			area = (width * height)/2;
			cout <<endl<< "- Area of Triangle : "<<area;
		}
};

int main()
{
	Rectangle r;
	Triangle t;

	cout<<"--------- -: Find Area of Rectangle :- ---------"<<endl<<endl;
	r.getData();
	r.RectangleData();

	cout<<endl<<endl<<"---------- -: Find Area of Triangle :- ----------"<<endl<<endl;
	t.getData();
	t.TriangleData();
	return 0;
}
```

Output:


D:\Mansi Lakhani-Flutter\C++\Heritage\1.exe
- Enter Width : 3
- Enter Height : 4
- Enter Width : 5
- Enter Height : 6
--------- -: Find Area of Rectangle :- ---------
- Width : 3
- Height : 4
- Area of rectangle : 12
---------- -: Find Area of Triangle :- ----------
- Width : 5
- Height : 6
- Area of Triangle : 15

Process exited after 4.228 seconds with return value 0
Press any key to continue . . .

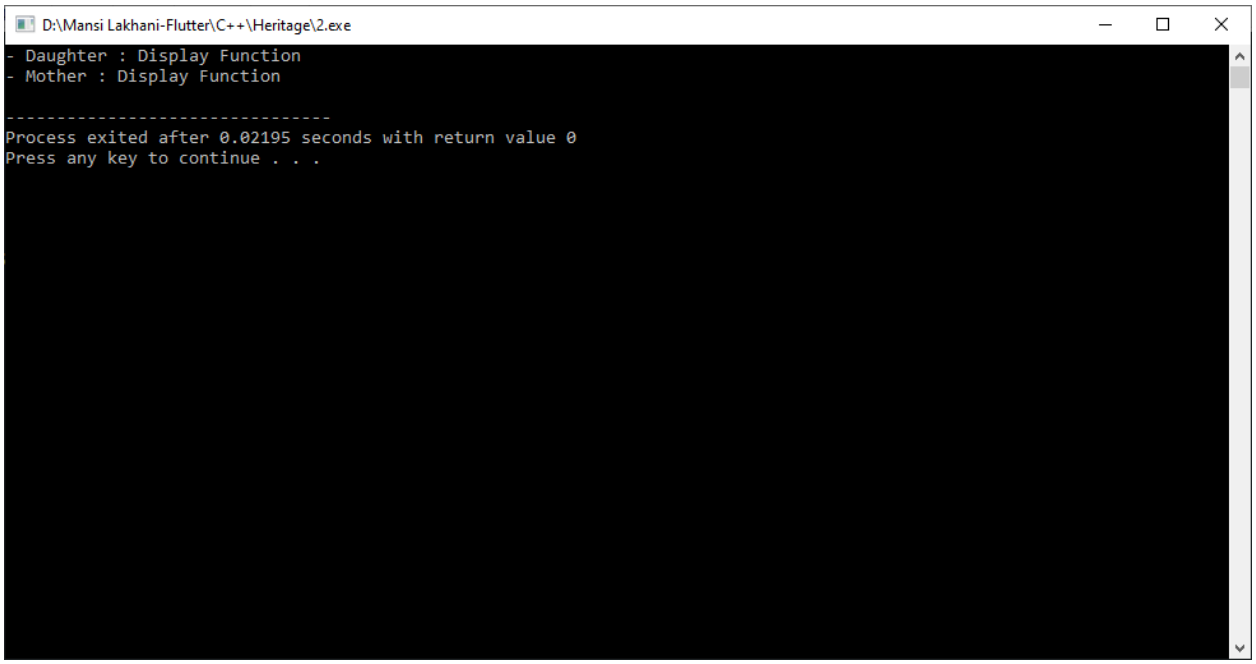
Practical-2

Aim: WAP with a mother class and an inherited daugther class.Both of them should have a method void display() that prints a message (different for mother and daugther). In the main define a daughter and call the display() method on it.

Program:

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

class Mother
{
	public :
		void M_display()
		 {
			cout << "- Mother : Display Function"<<endl;
		 }
};

class Daughter :public Mother
{
	public:
		void D_dislay()
		 {

			cout << "- Daughter : Display Function"<<endl;
		 }
};

int main()
{
	Daughter d;
	d.D_dislay();
	d.M_display();
	return 0;
```

Output:


D:\Mansi Lakhani-Flutter\C++\Heritage\2.exe
- Daughter : Display Function
- Mother : Display Function

Process exited after 0.02195 seconds with return value 0
Press any key to continue . . .

Practical-3

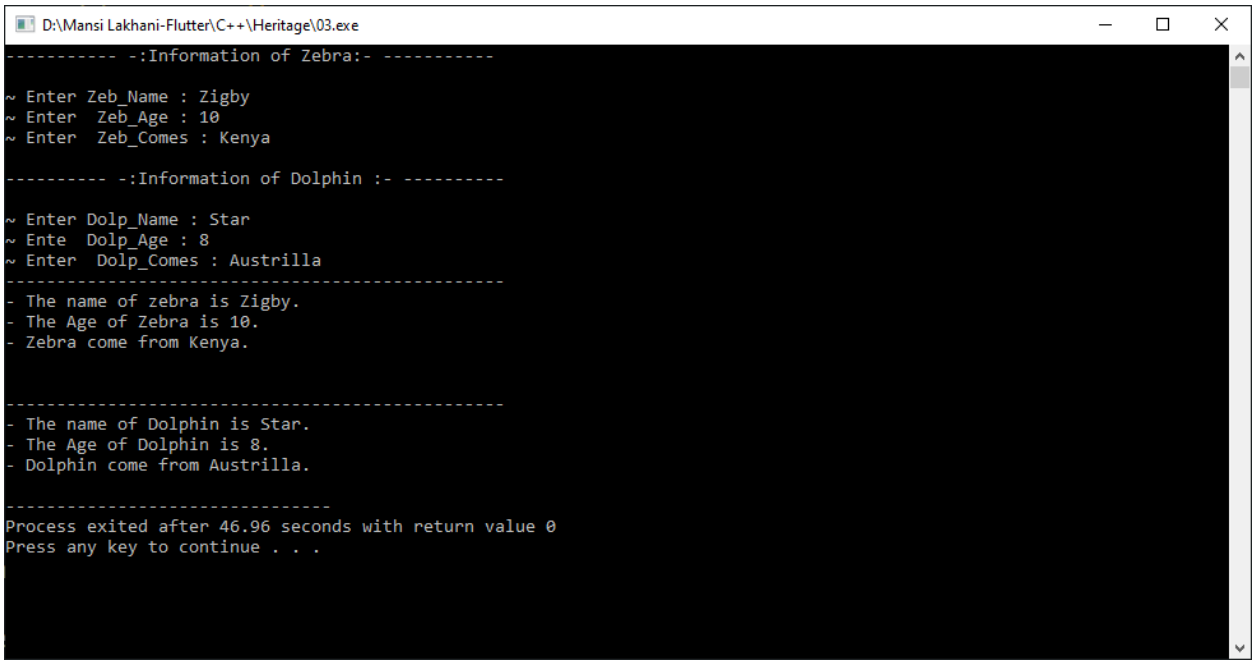
Aim:WAP with a mother class animal. Inside it define a name and an age variables, and set_value() function. Then create two bases variables Zebra and Dolphin which write a message telling the age, the name and giving some extra information (e.g. place of origin).

Program:

```
#include<iostream>
#include<string.h>
using namespace std;

class Animal
{
        public:
                int age;
                char name[100];
                char origin[100];


};

class Zebra : public Animal
{
        public:
                void setData()
                {
                        cout <<"----------- -:Information of Zebra:- -----------"<<endl<<endl;
                        cout << "~ Enter Zeb_Name : ";   cin >>name;
                        cout << "~ Enter  Zeb_Age : ";      cin >>age;
                        cout << "~ Enter  Zeb_Comes : ";    cin >>origin;
                }
                void getData()
                {

                        cout << "- The name of zebra is "<<this->name<<"."<<endl
                           << "- The Age of Zebra is "<<this->age<<"."<<endl
```

```
                        << "- Zebra come from "<<this->origin <<"." <<endl;
                }
};

class Dolphin : public Animal
{
        public:
                void setData()
                {
                        cout <<endl<<"---------- -:Information of Dolphin :-
----------"<<endl<<endl;
                        cout << "~ Enter Dolp_Name : ";   cin >>name;
                        cout << "~ Ente  Dolp_Age : ";      cin >>age;
                        cout << "~ Enter  Dolp_Comes : ";    cin >>origin;
                }
                void getData()
                {
                        cout <<endl<<endl<< "-----------------------------------------------"<<endl;
                        cout << "- The name of Dolphin is "<<this->name<<"."<<endl
                            << "- The Age of Dolphin is "<<this->age<<"."<<endl
                            << "- Dolphin come from "<<this->origin <<"." <<endl;
                }
};

int main()
{
        Zebra z;
        Dolphin d;

        z.setData();
        d.setData();
        cout << "-----------------------------------------------"<<endl;
        z.getData();
        d.getData();




        return 0;
```

Output:


D:\Mansi Lakhani-Flutter\C++\Heritage\03.exe
---------- -:Information of Zebra:- ----------
~ Enter Zeb_Name : Zigby
~ Enter Zeb_Age : 10
~ Enter Zeb_Comes : Kenya
---------- -:Information of Dolphin :- ----------
~ Enter Dolp_Name : Star
~ Ente Dolp_Age : 8
~ Enter Dolp_Comes : Austrilla
- The name of zebra is Zigby.
- The Age of Zebra is 10.
- Zebra come from Kenya.
- The name of Dolphin is Star.
- The Age of Dolphin is 8.
- Dolphin come from Austrilla.
Process exited after 46.96 seconds with return value 0
Press any key to continue . . .

Practical-4

Aim: WAP to read and print employee information using multiple inheritance.

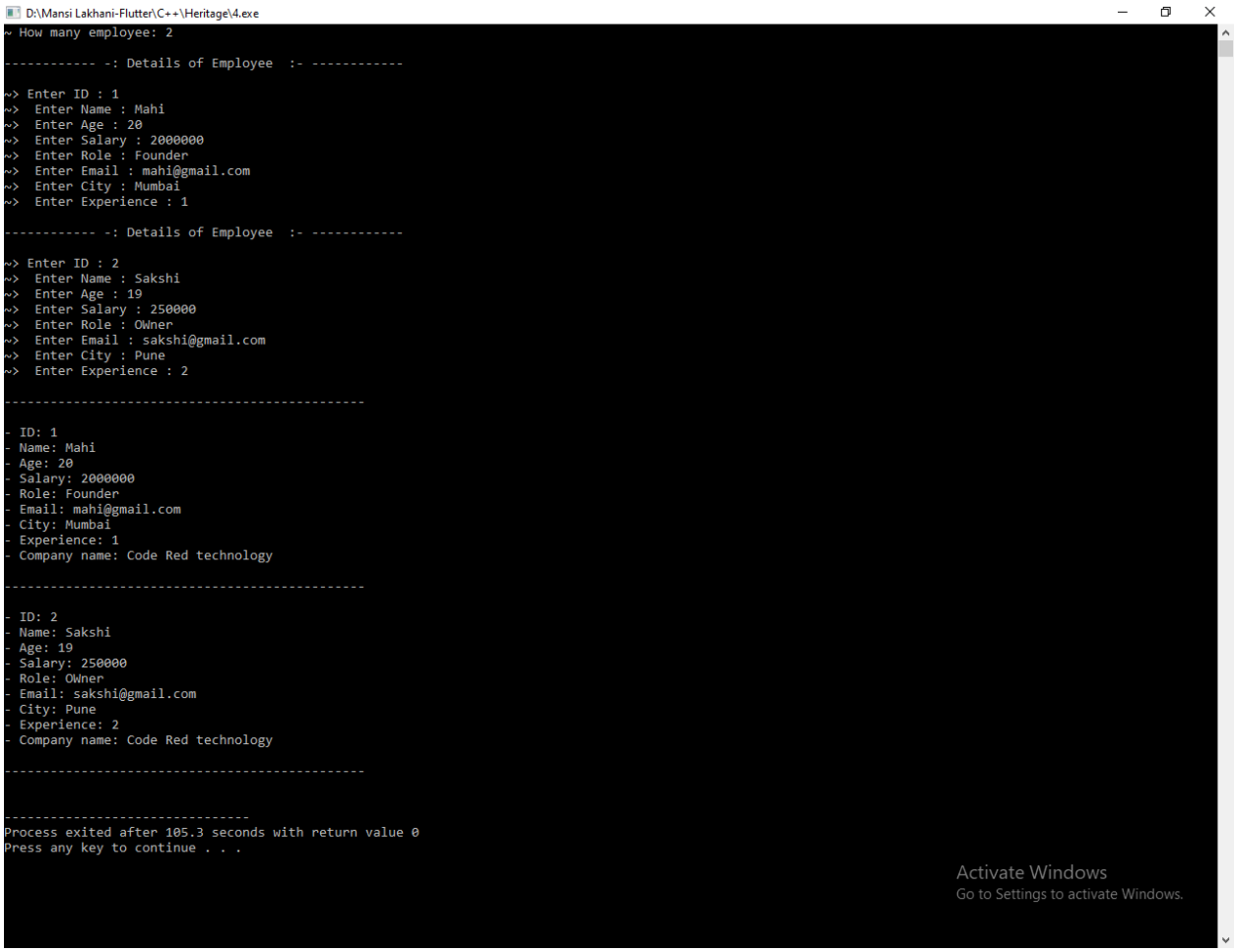
Program:

```
#include<iostream>
#include<string.h>
using namespace std;

class A
{
        public:
                int emp_id;
                char emp_name[1000];
                int emp_age;
                static char emp_company_name[100];

        public:
                void setAData()
                 {
                        cout << endl<<"------------ -: Details of Employee  :- ------------"
                              <<endl <<endl;
                        cout<<"~> Enter ID : ";                 cin >> this->emp_id;
                        cout<<"~>  Enter Name : ";                cin >> this->emp_name;
                        cout<<"~>  Enter Age : ";                 cin >> this->emp_age;
                 }
};

class B
{
         public:
                char emp_role[100];
                int  emp_salary;
                char emp_email[100];
                char emp_city[50];
                int emp_experience;
```

```
        public:
                void setBData()
                                {
                                        cout<<"~>  Enter Salary : ";
                                        cin >>      this->emp_salary;
                                        cout<<"~>  Enter Role : ";
                                        cin >> this->emp_role;
                                        cout<<"~>  Enter Email : ";
                                        cin >> this->emp_email;
                                        cout<<"~>  Enter City : ";
                                        cin >> this->emp_city;
                                        cout<<"~>  Enter Experience : ";
                                        cin >> this->emp_experience;
                                }
};

class C:public A ,public B
{
        public:

        void getAllData()
                {
                        cout    << "- ID: "<<this->emp_id <<endl
                                <<"- Name: "<<this->emp_name <<endl
                                <<"- Age: "<<this->emp_age <<endl
                                << "- Salary: "<<this->emp_salary <<endl
                                <<"- Role: "<<this->emp_role <<endl
                                <<"- Email: "<<this->emp_email <<endl
                                <<"- City: "<<this->emp_city <<endl
                                <<"- Experience: "<<this->emp_experience <<endl
                                 <<"- Company name: "<<this->emp_company_name <<endl;
                }
};

char A ::emp_company_name[100] = "Code Red technology";

int main()
{
```

```
        C c1[100];
        int i,n;
        cout <<"~ How many employee: ";
        cin >> n;


        for(i=0;i<n;i++)
        {
                c1[i].setAData();
                c1[i].setBData();

        }

        cout << endl<<"-----------------------------------------------"<<endl <<endl;

        for(i=0;i<n;i++)
        {
                c1[i].getAllData();
                cout << endl<<"-----------------------------------------------"<<endl <<endl;
        }

         return 0;
}
```

Output:


D:\Mansi Lakhani-Flutter\C++\Heritage\4.exe
~ How many employee: 2
------------ -: Details of Employee :- ------------
~> Enter ID : 1
~> Enter Name : Mahi
~> Enter Age : 20
~> Enter Salary : 2000000
~> Enter Role : Founder
~> Enter Email : mahi@gmail.com
~> Enter City : Mumbai
~> Enter Experience : 1
------------ -: Details of Employee :- ------------
~> Enter ID : 2
~> Enter Name : Sakshi
~> Enter Age : 19
~> Enter Salary : 250000
~> Enter Role : OWner
~> Enter Email : sakshi@gmail.com
~> Enter City : Pune
~> Enter Experience : 2
- ID: 1
- Name: Mahi
- Age: 20
- Salary: 2000000
- Role: Founder
- Email: mahi@gmail.com
- City: Mumbai
- Experience: 1
- Company name: Code Red technology
- ID: 2
- Name: Sakshi
- Age: 19
- Salary: 250000
- Role: OWner
- Email: sakshi@gmail.com
- City: Pune
- Experience: 2
- Company name: Code Red technology
Process exited after 105.3 seconds with return value 0
Press any key to continue . . .
Activate Windows
Go to Settings to activate Windows.

Practical-5

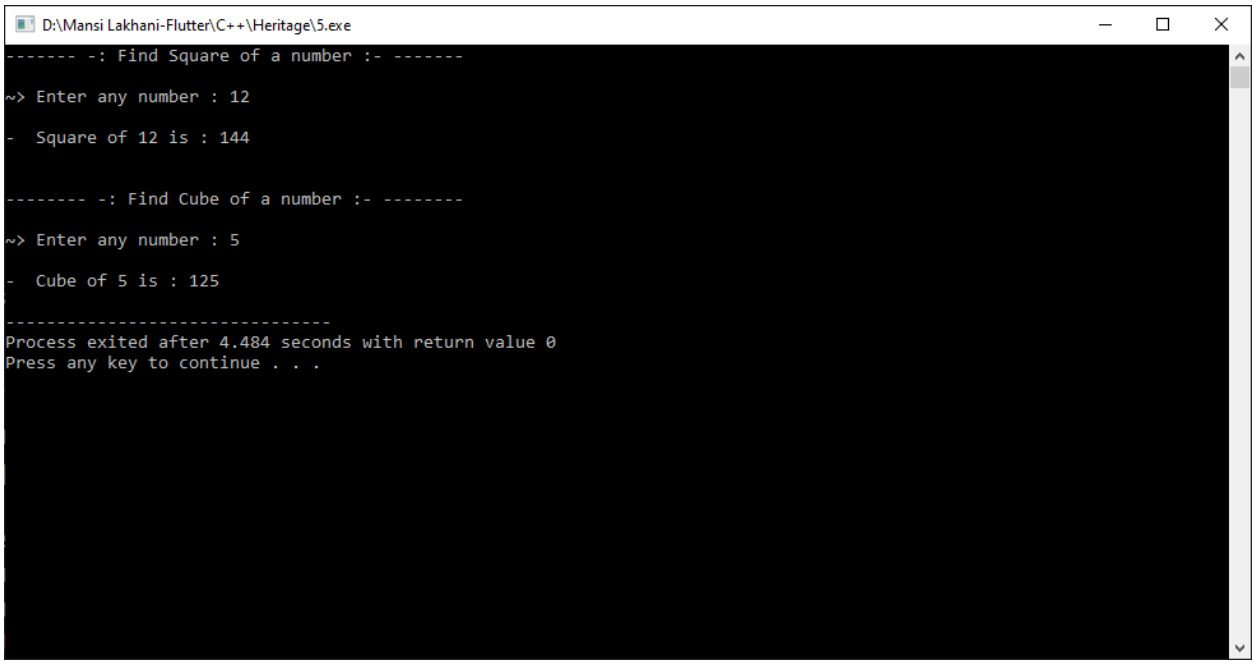
Aim: WAP to demonstrate example of hierarchical inheritance to get square and cube of a number.

Program:

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

class Number
{
        public:
                int n;
        public:
                void getNumberData()
                {
                        cout << "~> Enter any number : ";
                        cin >> n;
                }
};

class Square : public Number
{
        public:
                int sqr;

        public:
                void getSquareData()
                {
                   sqr = n*n;
                  cout <<endl<<"-  Square of "<<n << " is : "<<sqr <<endl;
                }
};

class Cube : public Number
```

```
{
	public:
		int cube;

	public:
		void getCubeData()
		{
			cube = n*n*n;
			cout <<endl<<"-  Cube of "<<n << " is : "<<cube <<endl;
		}
};

int main()
{
	Square s1;
	Cube c1;

	cout <<"------- -: Find Square of a number :- -------"<<endl<<endl;
	s1.getNumberData();
	s1.getSquareData();

	cout <<endl<<endl<<"-------- -: Find Cube of a number :- --------"<<endl<<endl;
	c1.getNumberData();
	c1.getCubeData();
	return 0;
}
```

Output:


D:\Mansi Lakhani-Flutter\C++\Heritage\5.exe
------- -: Find Square of a number :- -------
~> Enter any number : 12
- Square of 12 is : 144
-------- -: Find Cube of a number :- --------
~> Enter any number : 5
- Cube of 5 is : 125

Process exited after 4.484 seconds with return value 0
Press any key to continue . . .

Practical-6

Aim: WAP to read and print employee information with use of multilevel inheritance.

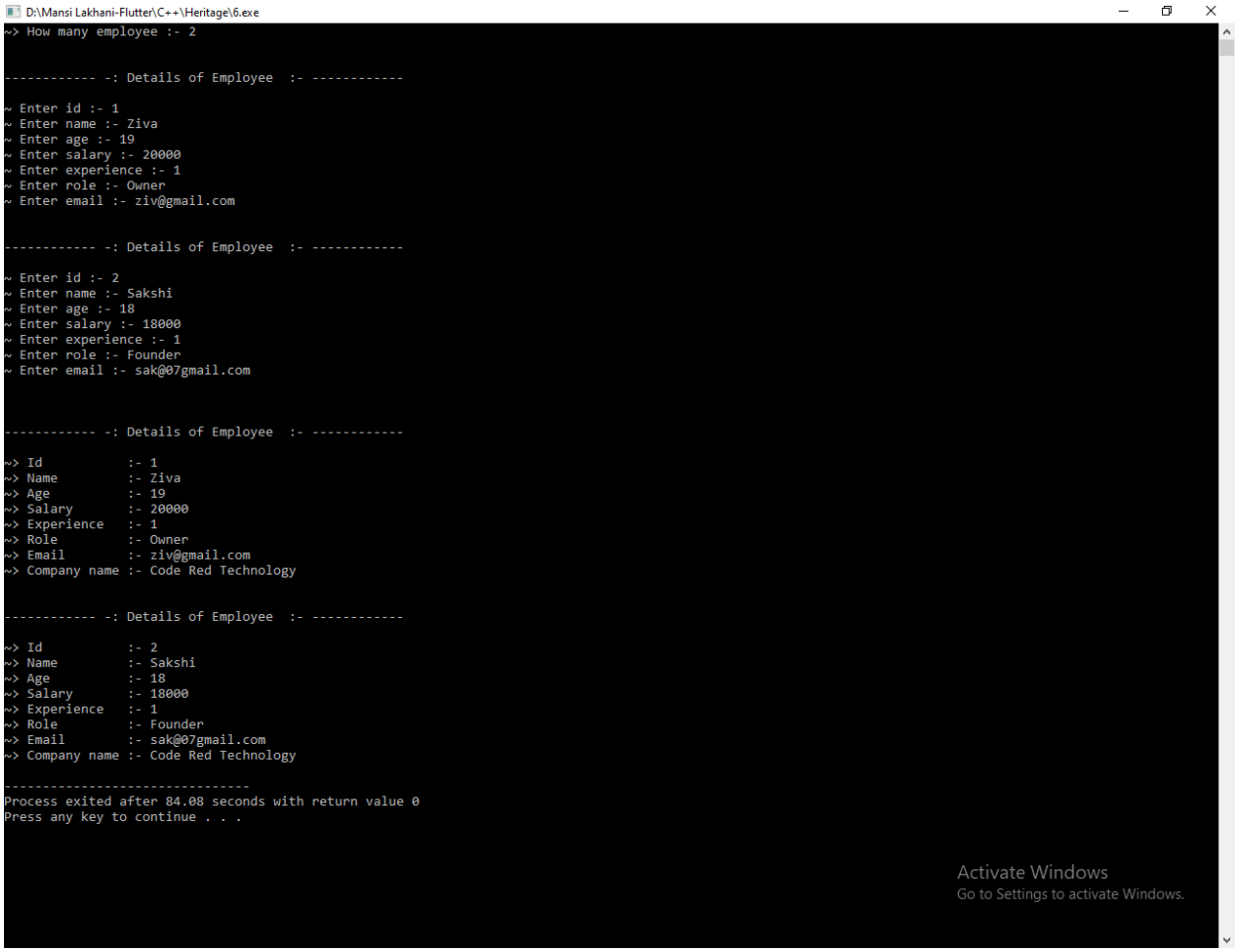
Program:

```
#include<iostream>
#include<string.h>
using namespace std;

class A
{
        public :
                int  emp_id;
                char emp_name[100];
                int  emp_age;
                int  emp_salary;
                int  emp_experience;
                char emp_role[100];
                char emp_email[100];
                static char emp_company_name[100];
};

class B : public A
{
        public :
        void setB()
        {
                cout << endl<<"------------ -: Details of Employee  :- ------------"
                      <<endl     <<endl;
                cout << "~ Enter id :- ";
                cin  >> this->emp_id;
```

```
            cout << "~ Enter name :- ";
            cin  >> this->emp_name;
            cout << "~ Enter age :- ";
            cin  >> this->emp_age;
        }
};

class C : public B
{
        public :
        void setC()
        {
            cout << "~ Enter salary :- ";
            cin  >> this->emp_salary;
            cout << "~ Enter experience :- ";
            cin  >> this->emp_experience;
            cout << "~ Enter role :- ";
            cin  >> this->emp_role;
            cout << "~ Enter email :- ";
            cin  >> this->emp_email;
            cout << endl;
        }
};

class D : public C
{
        public :
        void getAllData()
        {
            cout << endl<<endl
            <<"------------ -: Details of Employee  :- ------------"<<endl <<endl;
            cout << "~> Id           :- " << this->emp_id << endl;
            cout << "~> Name         :- " << this->emp_name << endl;
            cout << "~> Age          :- " << this->emp_age << endl;
            cout << "~> Salary       :- " << this->emp_salary << endl;
            cout << "~> Experience   :- " << this->emp_experience << endl;
            cout << "~> Role         :- " << this->emp_role << endl;
```

```
            cout << "~> Email        :- " << this->emp_email << endl;
            cout << "~> Company name :- " << this->emp_company_name << endl;
        }
};

char A :: emp_company_name[100] = "Code Red Technology";

int main()
{
        D s[100];
        int i,n;

        cout <<"~> How many employee :- "; cin >> n;
        cout << endl;

        for(i=0;i<n;i++)
        {
                s[i].setB();
                s[i].setC();
        }
        for(i=0;i<n;i++)
        {
                s[i].getAllData();
        }
        return 0;
}
```

Output:


D:\Mansi Lakhani-Flutter\C++\Heritage\6.exe
~> How many employee :- 2
------------ -: Details of Employee :- ------------
~ Enter id :- 1
~ Enter name :- Ziva
~ Enter age :- 19
~ Enter salary :- 20000
~ Enter experience :- 1
~ Enter role :- Owner
~ Enter email :- ziv@gmail.com
------------ -: Details of Employee :- ------------
~ Enter id :- 2
~ Enter name :- Sakshi
~ Enter age :- 18
~ Enter salary :- 18000
~ Enter experience :- 1
~ Enter role :- Founder
~ Enter email :- sak@07gmail.com
------------ -: Details of Employee :- ------------
~> Id :- 1
~> Name :- Ziva
~> Age :- 19
~> Salary :- 20000
~> Experience :- 1
~> Role :- Owner
~> Email :- ziv@gmail.com
~> Company name :- Code Red Technology
------------ -: Details of Employee :- ------------
~> Id :- 2
~> Name :- Sakshi
~> Age :- 18
~> Salary :- 18000
~> Experience :- 1
~> Role :- Founder
~> Email :- sak@07gmail.com
~> Company name :- Code Red Technology

Process exited after 84.08 seconds with return value 0
Press any key to continue . . .
Activate Windows
Go to Settings to activate Windows.