**Practical : 1**

**Aim :** WAP to create a class Employee with setter and getter. Which have fields like id, name, role, salary, experience, address, email and contact. Get 5 records.

**Program :**

#include<iostream>

#include<string.h>

using namespace std;

class Emp

{

private :

int id;

char name[100];

char role[100];

int salary;

int exp;

char address[100];

char email[1000];

int contact;

public:

void setData()

{

cout<<endl<<"enter emp id :";

cin>>this->id;

cout<<endl<<"enter emp name :";

cin>>this->name;

cout<<endl<<"enter emp role :";

cin>>this->role;

cout<<endl<<"enter emp salary :";

cin>>this->salary;

cout<<endl<<"enter emp experience :";

cin>>this->exp;

cout<<endl<<"enter emp address :";

cin>>this->address;

cout<<endl<<"enter emp email id :";

cin>>this->email;

cout<<endl<<"enter emp contact no :";

cin>>this->contact;

}

void getData()

{

cout<<endl<<" Emp Details";

cout<<endl<<" ============";

cout<<endl<<"1.emp id :"<<this->id;

cout<<endl<<"2.emp name :"<<this->name;

cout<<endl<<"3.emp role :"<<this->role;

cout<<endl<<"4.emp salary :"<<this->salary;

cout<<endl<<"5.emp experience :"<<this->exp;

cout<<endl<<"6.emo address :"<<this->address;

cout<<endl<<"7.emp email id :"<<this->email;

cout<<endl<<"8.emp contact no :"<<this->contact;

}

};

int main()

{

Emp e[100];

int i,n;

cout<<"how many emp enter :";

cin>>n;

for(i=0;i<n;i++)

{

cout<<" Enter the employee information :";

cout<<i+1;

e[i].setData();

}

cout<<endl<<"\*\*\* The employee information is \*\*\*";

for(int i=0; i<n;i++)

{

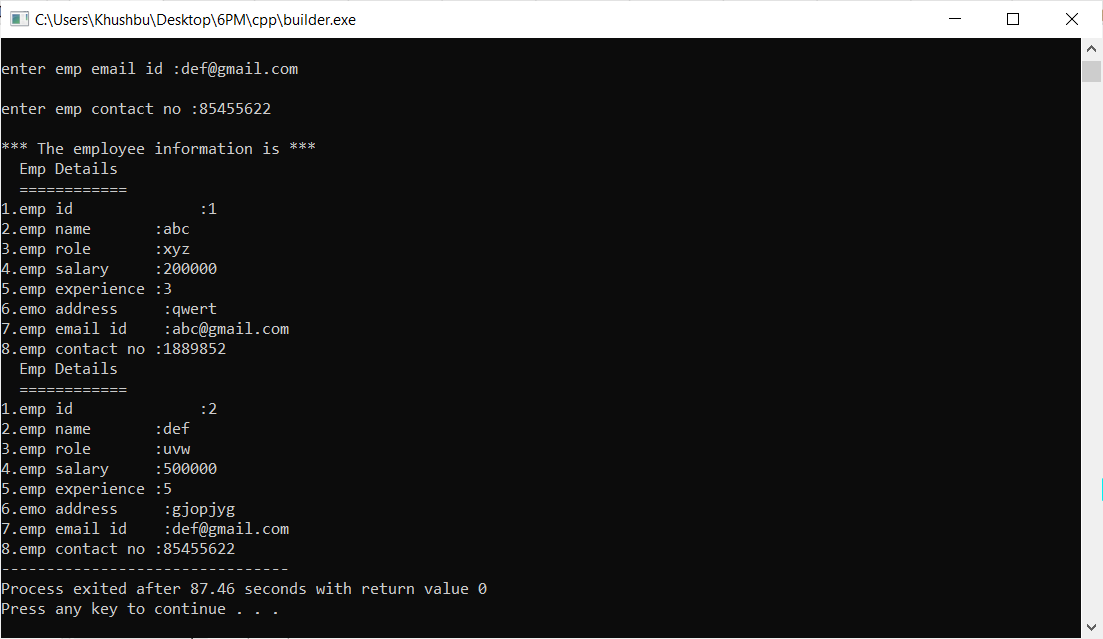
e[i].getData();

}

return 0;

}

**Output :**

****

**Practical : 2**

**Aim :** WAP to create a class which Read and print Class, Student details using Two Classes. (Make two classes, create one classe's obj in another class.)

**Program :**

#include <iostream>

#include <string.h>

using namespace std;

class student{

private:

char name[100];

int rollNo;

int age;

int contact\_no;

public:

void setStudent()

{

cout<<endl<<"enter student name :";

cin>>this->name;

cout<<endl<<"enter student roll no :";

cin>>this->rollNo;

cout<<endl<<"enter student age :";

cin>>this->age;

cout<<endl<<"enter student contact no :";

cin>>this->contact\_no;

}

void getStudent()

{

cout<<endl <<" Name : "<<this->name

<<" Roll No. : "<<this->rollNo

<<" Age : "<<this->age

<<" Contact no : "<<this->contact\_no<<endl;

}

};

class classDetails{

private:

char cname[100];

student s;//object

public:

void setClass()

{

cout<<endl<<"enter class Name :";

cin>>this->cname;

s.setStudent();

}

void getClass()

{

cout<<"class name : "<<this->cname;

s.getStudent();

}

};

int main()

{

classDetails c[100];

int i,n;

cout<<"How many student and class details u want to enter => ";

cin>>n;

cout<<endl<<" student and class information : "<<i+1<<endl;

for(i=0;i<n;i++)

{

c[i].setClass();

}

cout<<endl<<" \*\* student and class information is \*\*"<<endl;

for(i=0;i<n;i++)

{

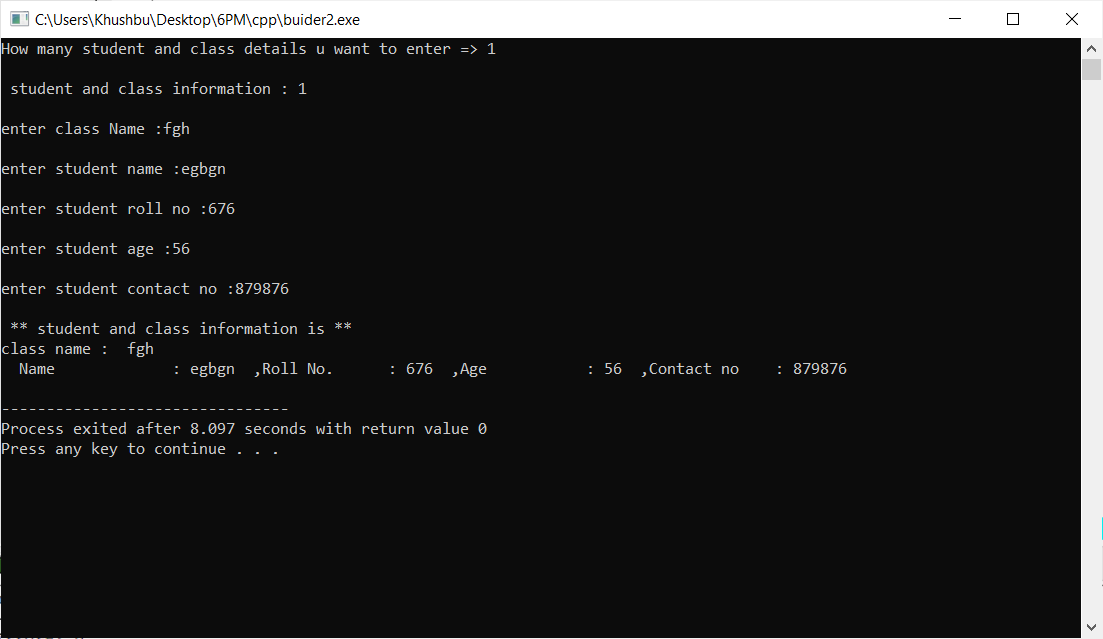
c[i].getClass();

}

return 0;

}

**Output :**



**Practical : 3**

**Aim :** WAP to create a class Hotel with fields like id, name, type, staff\_size, room\_size, establish\_year, address, rating\_type and website. Illustrate the use of encapsulation (strict encapsulation) with this keyword.

**Program :**

#include<iostream>

#include<string.h>

using namespace std;

class Hotel

{

private :

int id;

char name[100];

int staff\_size;

int room\_size;

int establish\_year;

char address[100];

int rating\_type;

char website[100];

public:

void setData()

{

cout<<"enter hotel id :";

cin>>this->id;

cout<<endl<<"enter hotel name :";

cin>>this->name;

cout<<endl<<"enter hotel staff\_size :";

cin>>this->staff\_size;

cout<<endl<<"enter hotel room\_size :";

cin>>this->room\_size;

cout<<endl<<"enter hotel establish\_year :";

cin>>this->establish\_year;

cout<<endl<<"enter hotel address :";

cin>>this->address;

cout<<endl<<"enter hotel rating\_type :";

cin>>this->rating\_type;

cout<<endl<<"enter hotel website:";

cin>>this->website;

}

void getData()

{

cout<<"id:"<<this->id

<<",name :"<<this->name

<<",staff\_size :"<<this->staff\_size

<<",room\_size:"<<this->room\_size

<<",establish\_year :"<<this->establish\_year

<<",address :"<<this->address

<<",rating\_type :"<<this->rating\_type

<<",website :"<<this->website<<endl;

}

};

int main()

{

Hotel h[100];

int i,n;

cout<<"how many hotel details enter :";

cin>>n;

cout<<" Enter the hotel information :"<<endl;

cout<<i+1<<endl;

for(i=0;i<n;i++)

{

h[i].setData();

}

cout<<endl<<"\*\*\* The hotel information is \*\*\*";

for(int i=0; i<n;i++)

{

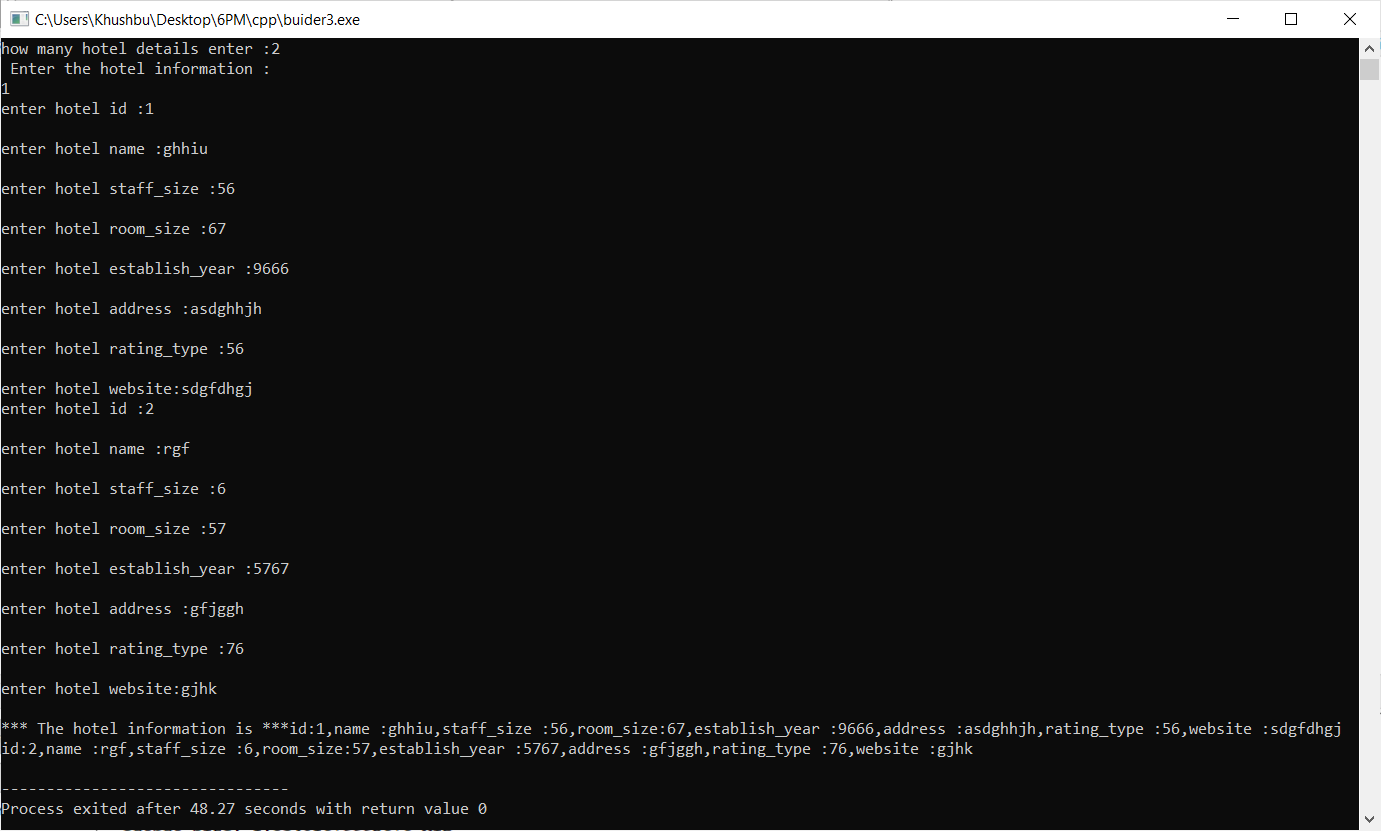
h[i].getData();

}

return 0;

}

**Output :**



**Practical : 4**

**Aim :** WAP to create two class HighSchool and College with fields like id, stu\_name, stu\_roll\_no, stu\_standard, stu\_age, stu\_contact, stu\_edu\_institute\_name and stu\_address. Make suitable setter and getter with use of static data members.

**Program :**

#include <iostream>

#include <string.h>

using namespace std;

class School{

private:

char name[100];

int rollNo;

int std;

int age;

int contact\_no;

char institute\_name[100];

char add[100];

public :

static int id;

public:

void setScl()

{

cout<<endl<<"scl infotmation";

cout<<endl<<"===============";

cout<<endl<<"enter student name :";

cin>>this->name;

cout<<endl<<"enter student roll no :";

cin>>this->rollNo;

cout<<endl<<"enter student standerd :";

cin>>this->std;

cout<<endl<<"enter student age :";

cin>>this->age;

cout<<endl<<"enter student contact no :";

cin>>this->contact\_no;

cout<<endl<<"enter student address :";

cin>>this->add;

}

void getScl()

{

cout<<endl <<" ,Id :" <<this->id

<<" ,Name : "<<this->name

<<" ,Roll No. : "<<this->rollNo

<<" ,Standerd : "<<this->std

<<" ,Age : "<<this->age

<<" ,Contact no : "<<this->contact\_no

<<" ,Institute name: "<<this->institute\_name

<<" ,Address :" <<this->add<<endl;

}

};

class Clg{

private:

char name[100];

int rollNo;

char course[100];

int age;

int contact\_no;

char institute\_name[100];

char add[100];

School s;//object

public :

static int id;

public:

void setClg()

{

cout<<endl<<"clg infotmation";

cout<<endl<<"===============";

cout<<endl<<"enter student name :";

cin>>this->name;

cout<<endl<<"enter student roll no :";

cin>>this->rollNo;

cout<<endl<<"enter student course :";

cin>>this->course;

cout<<endl<<"enter student age :";

cin>>this->age;

cout<<endl<<"enter student contact no :";

cin>>this->contact\_no;

cout<<endl<<"enter student institute name :";

cin>>this->institute\_name;

cout<<endl<<"enter student address :";

cin>>this->add;

s.setScl();

}

void getClg()

{

cout<<endl <<" Id :" <<this->id

<<" ,Name : "<<this->name

<<" ,Roll No. : "<<this->rollNo

<<" ,Course : "<<this->course

<<" ,Age : "<<this->age

<<" ,Contact no : "<<this->contact\_no

<<" ,Institute name: "<<this->institute\_name

<<" ,Address :" <<this->add<<endl;

s.getScl();

}

};

int School :: id = 4546;

int Clg :: id = 6778;

int main()

{

Clg c[100];

int i,n;

cout<<"School Id is : "<<School :: id<<endl;

cout<<"college Id is : "<<Clg :: id<<endl;

cout<<"How many school and clg details u want to enter => ";

cin>>n;

for(i=0;i<n;i++)

{

cout<<endl<<" school and clg information : "<<i+1<<endl;

c[i].setClg();

}

cout<<endl<<" \*\* school and clg information is \*\*"<<endl;

for(i=0;i<n;i++)

{

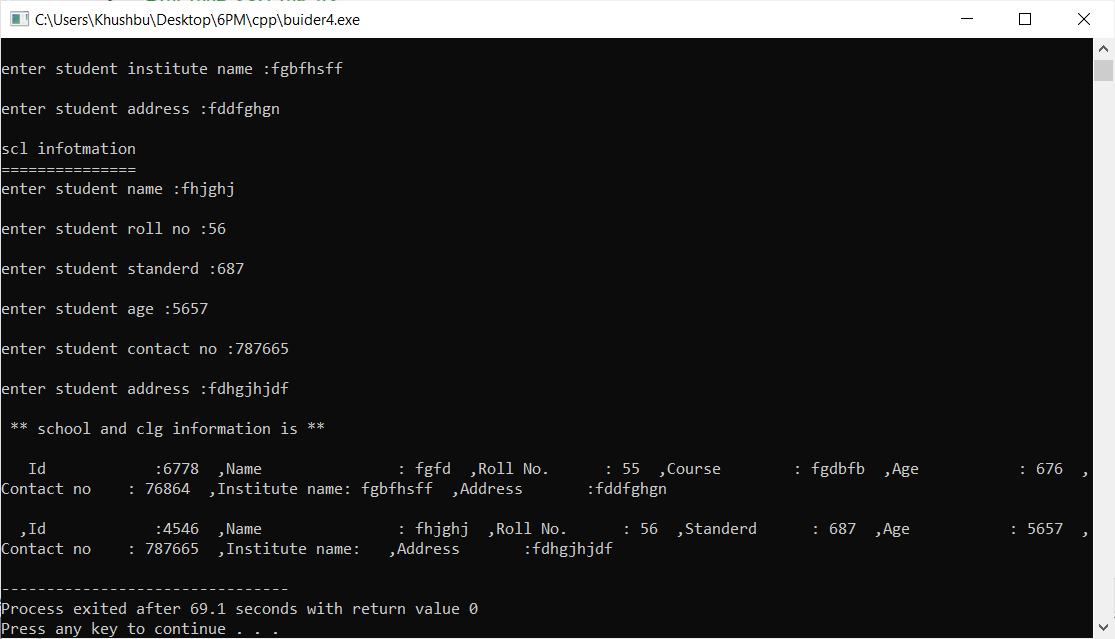
c[i].getClg();

}

return 0;

}

**Output :**



**Practical : 5**

**Aim :** WAP to create a class which have both static data member and static member function. That class gives details of all states in India.

**Program :**

#include <iostream>

#include <string.h>

using namespace std;

class State

{

private:

char state\_name[100];

int area;

char zone[100];

char largest\_city[100];

public :

static char official\_language[100];

public:

static void language()

{

strcpy(official\_language,"Hindi");

}

void setState()

{

cout<<endl<<"India State infotmation";

cout<<endl<<"=======================";

cout<<endl<<"enter state\_name :";

cin>>this->state\_name;

cout<<endl<<"enter area :";

cin>>this->area;

cout<<endl<<"enter zone :";

cin>>this->zone;

cout<<endl<<"enter largest\_city :";

cin>>this->largest\_city;

}

void getState()

{

cout<<endl <<" state\_name :" <<this->state\_name

<<" , area : "<<this->area

<<" , zone : "<<this->zone

<<" , largest\_city : "<<this->largest\_city

<<" , official\_language : "<<this->official\_language<<endl;

}

};

char State :: official\_language[100] = "Hindi";

int main()

{

State s[100];

int i,n;

State :: language() ;

cout<<"How many State details u want to enter => ";

cin>>n;

for(i=0;i<n;i++)

{

cout<<endl<<" State information : "<<i+1<<endl;

s[i].setState();

}

cout<<endl<<" \*\* State information is \*\*"<<endl;

for(i=0;i<n;i++)

{

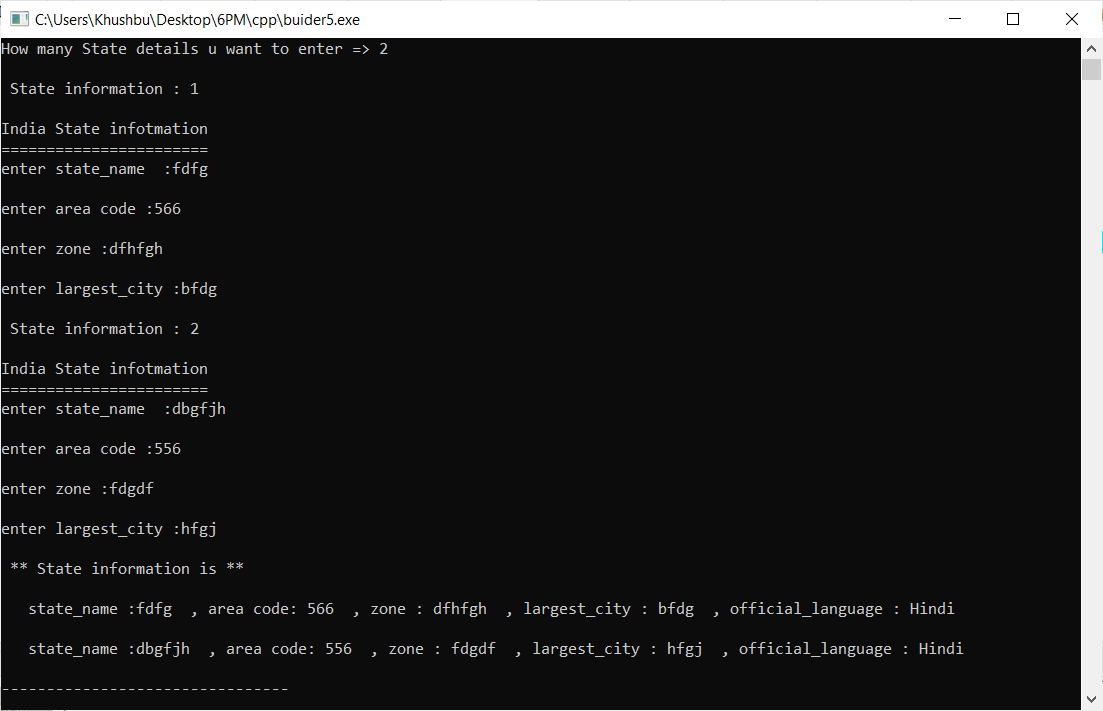
s[i].getState();

}

return 0;

}

**Output :**



**Practical : 6**

**Aim :** WAP to demonstrate example of default constructor or no argument constructor.

**Program :**

#include <iostream>

using namespace std;

class DefaultConstructor

{

private:

int num1, num2 ;

public:

DefaultConstructor()

{

num1 = 10;

num2 = 20;

}

void display()

{

cout<<"num1 = "<< num1 <<endl;

cout<<"num2 = "<< num2 <<endl;

}

};

int main()

{

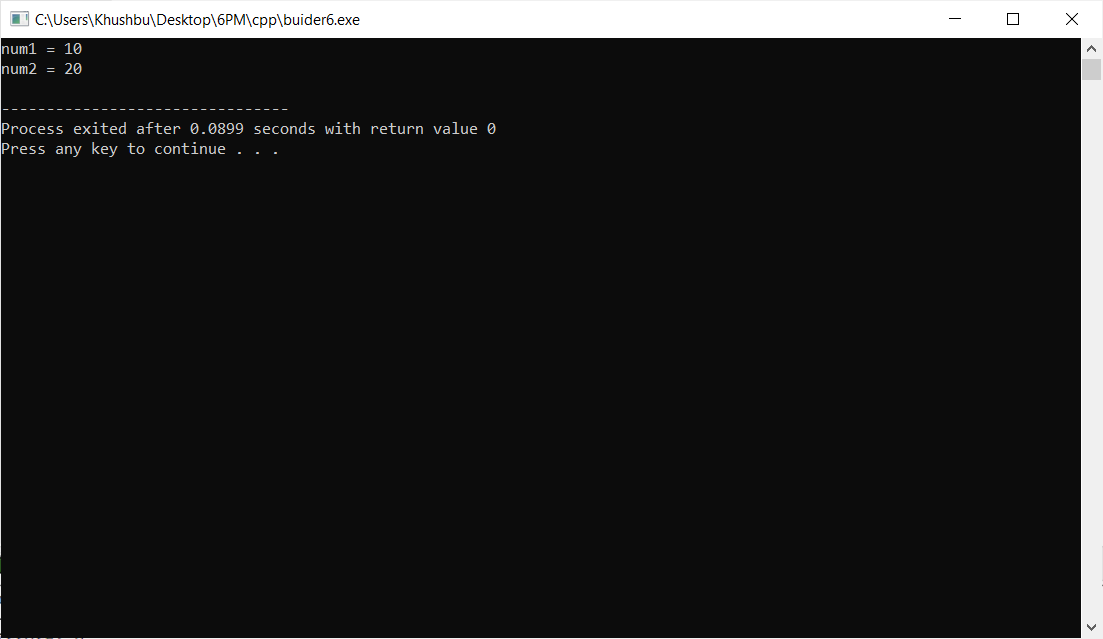
DefaultConstructor de;

de.display();

return 0;

}

**Output :**



**Practical : 7**

**Aim :** WAP to demonstrate example of parameterized constructor.

**Program :**

#include <iostream>

using namespace std;

class A

{

private:

int b, c;

public:

A (int b1, int c1)

{

b = b1;

c = c1;

}

int getX ()

{

return b;

}

int getY ()

{

return c;

}

};

int main ()

{

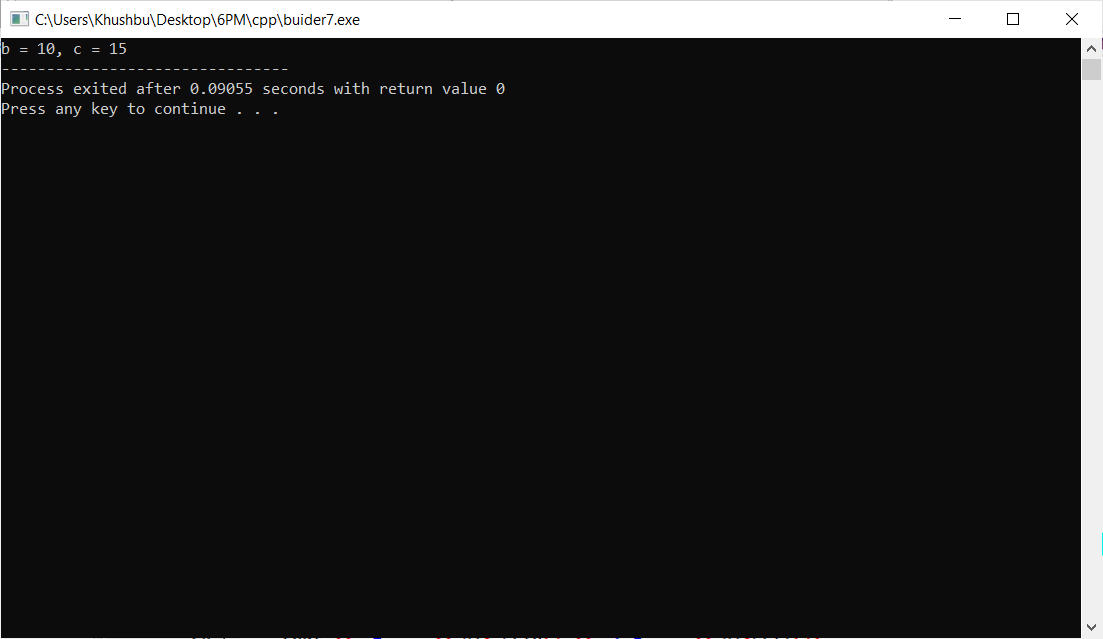
A p1(10, 15);

cout << "b = " << p1. getX() << ", c = " << p1.getY();

return 0;

}

**Output :**



**Practical : 8**

**Aim :**  WAP to demonstrate example of copy constructor.

**Program :**

#include<iostream>

using namespace std;

class Point

{

private:

int x, y;

public:

Point(int x1, int y1)

{

x = x1;

y = y1;

}

Point(const Point &p1)

{

x = p1.x;

y = p1.y;

}

int getX()

{

return x;

}

int getY()

{

return y;

}

};

int main()

{

Point p1(10, 15);

Point p2 = p1;

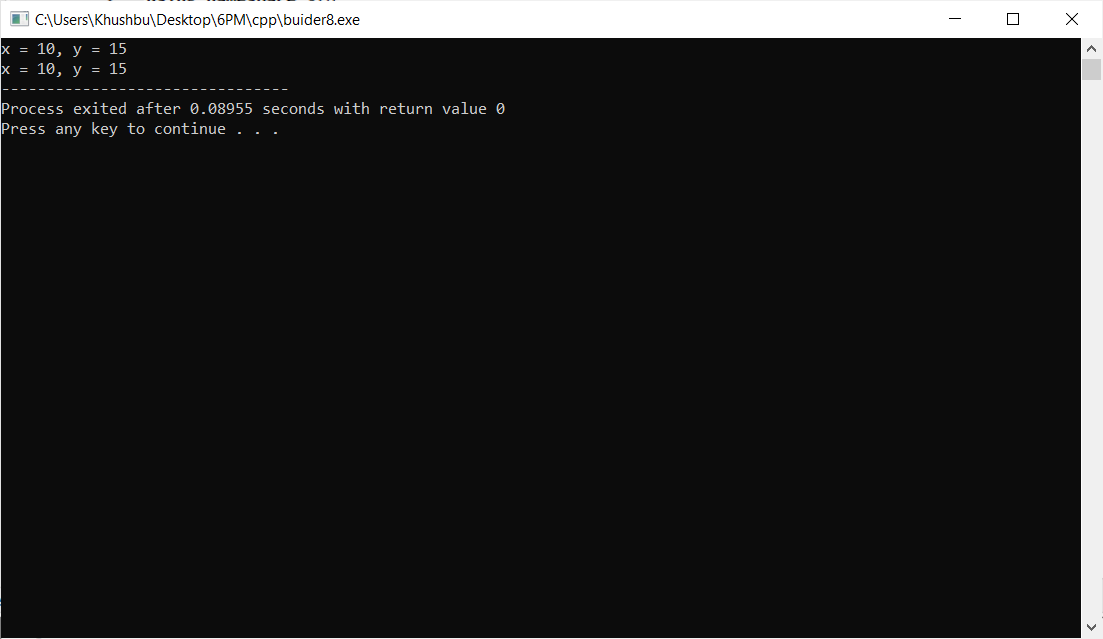
cout << "x = " << p1.getX() << ", y = " << p1.getY();

cout << "\nx = " << p2.getX() << ", y = " << p2.getY();

return 0;

}

**Output :**



**Practical : 9**

**Aim :** WAP to demonstrate example of destructors.

**Program :**

#include <iostream>

using namespace std;

class A {

public:

int roll;

apple()

{

roll=5;

cout << "Roll is " << roll << endl;

};

~A()

{

roll=5;

cout << "Roll is " << roll << endl;

}

};

int main()

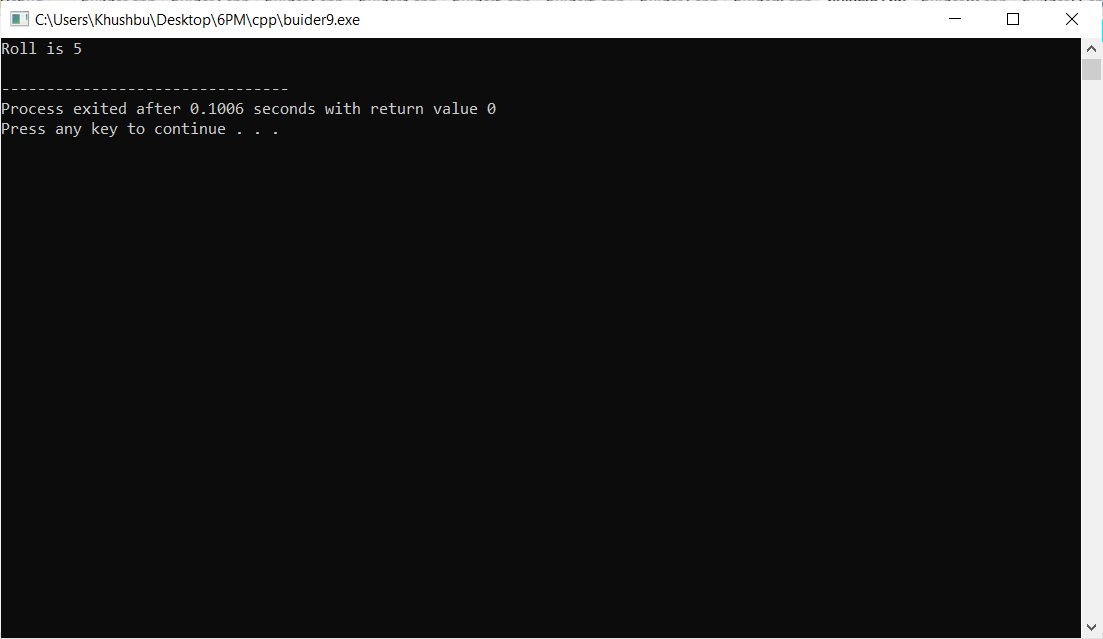
{

A obj1;

return 0;

}

**Output :**

****

**Practical : 10**

**Aim :** WAP to create a class which Set values of data members using default, parameterized and copy constructor.

**Program :**

#include <iostream>

#include <string.h>

using namespace std;

class A{

private:

char name[30];

int age;

public:

//default constructor

A(){

strcpy(name,"None");

age = 0;

}

//parameterized constructor

A(char n[], int a){

strcpy(name, n);

age = a;

}

//copy constructor

A(A &p){

strcpy(name, p.name);

age =p.age;

}

void getData()

{

cout<<name<<" is "<<age<<" years old."<<endl;

}

};

int main(){

A p1;

A p2("Khushbu Khakhriya",21);

A p3(p2);

cout<<"default constructor..."<<endl;

p1.getData();

cout<<"parametrized constructor..."<<endl;

p2.getData();

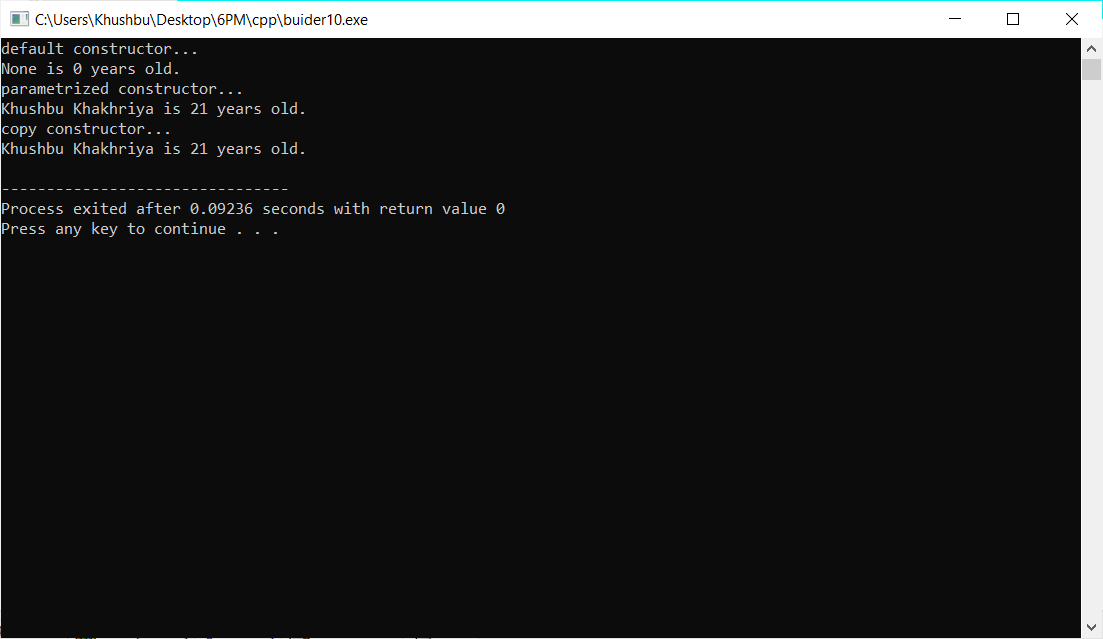
cout<<"copy constructor..."<<endl;

p3.getData();

return 0;

}

**Output :**

****

**Practical : 11**

**Aim :** WAP to create a class which illustrate the use of constant object.

**Program :**

#include <iostream>

using namespace std;

class A {

int id;

public:

A(int v = 0)

{

id = v;

}

int getValue() const

{

return id;

}

};

int main()

{

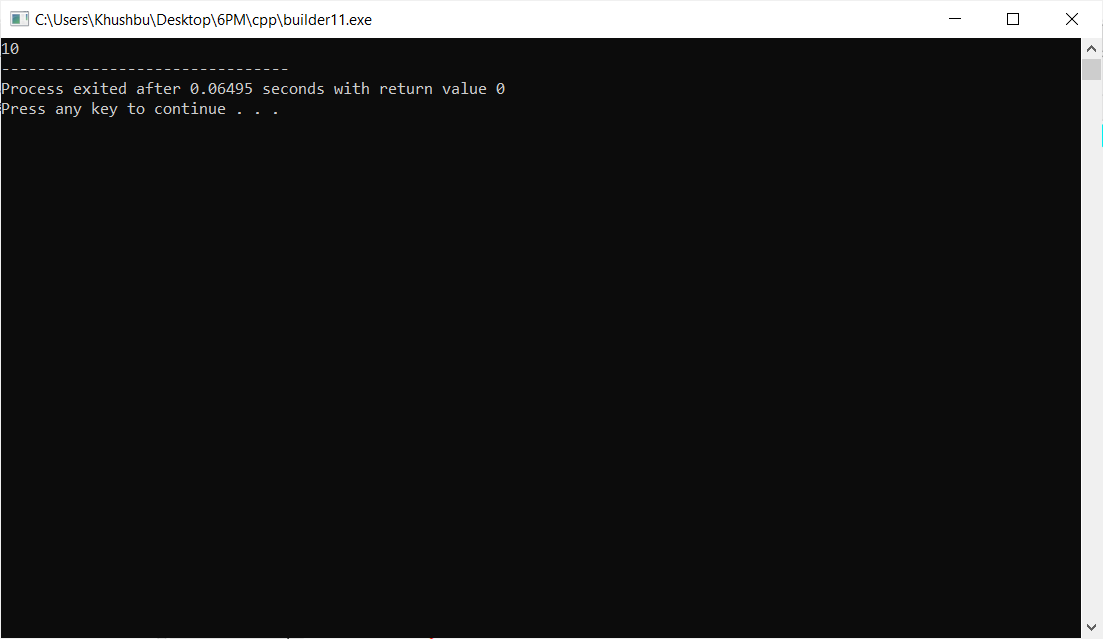
A t(10);

cout << t.getValue();

return 0;

}

**Output :**

****

**Practical : 12**

**Aim :** WAP to create a class for student to get and print details of N students. (with use of array of objects)

**Program :**

#include <iostream>

#include<string.h>

using namespace std;

class student

{

private:

char name[30];

int rollNo;

int std;

int address[100];

public:

void setDetails()

{

cout << "Enter name: " ;

cin >> this->name;

cout << "Enter roll number: ";

cin >> this->rollNo;

cout << "Enter standered ";

cin >> this->std;

}

void getDetails()

{

cout<<"name:"<<this->name

<<",roll no :"<<this->rollNo

<<",standered :"<<this->std<<endl;

}

};

int main()

{

student s[100]; //array of objects creation

int n,i;

cout << "Enter total number of students: ";

cin >> n;

for(i=0;i< n; i++){

cout << "Enter details of student " << i+1 << ":\n";

s[i].setDetails();

}

cout << endl;

for(i=0;i< n; i++){

cout << "Details of student "<< ":\n";

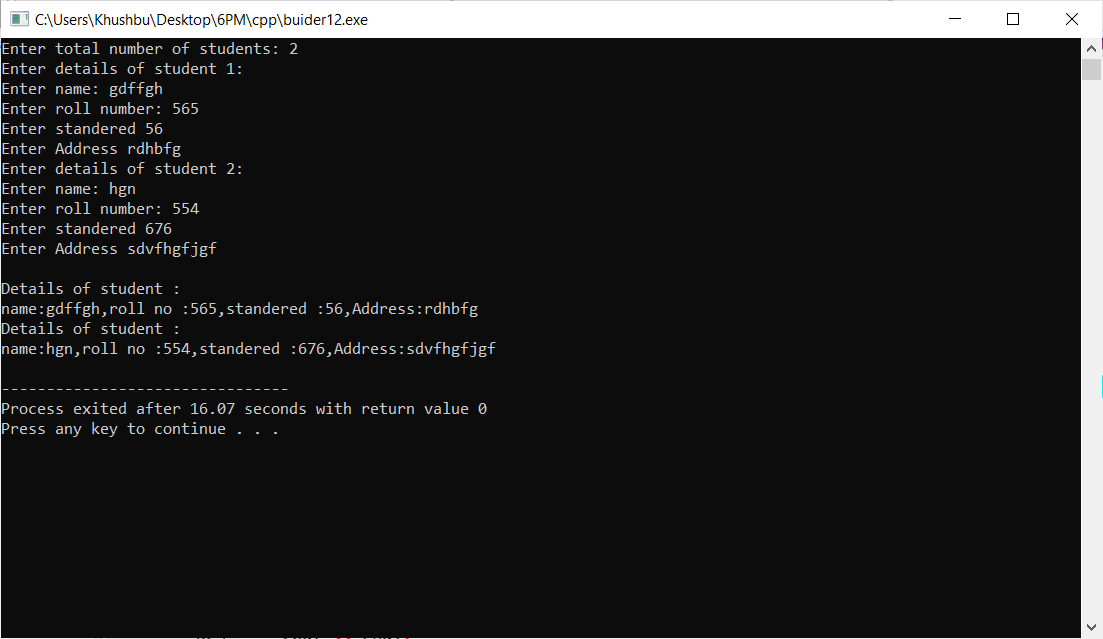
s[i].getDetails();

}

return 0;

}

**Output :**

****

**Practical : 13**

**Aim :** WAP to make Railway Reservation System.

Requirements:

(A) User Input Train Number , Train Name , Source , Destination , Train Time.

(B) Display Record By Search Train Number.

(C) Minimum 3 Input Train Record.

**Program :**

#include<iostream>

using namespace std;

class train

{

private:

int train\_number;

char train\_name[100];

char train\_destination[100];

int train\_time;

public:

void input()

{

cout<<endl<<"\t\tEnter Train Details :-"<<endl;

cout<<"\tTrain Number : "; cin>>this->train\_number;

cout<<"\tTrain Name : "; cin>>this->train\_name;

cout<<"\tTrain Destination : "; cin>>this->train\_destination;

cout<<"\tTrain Time : "; cin>>this->train\_time;

}

void display()

{

cout<<endl<<"\t\t Train Details :-"<<endl;

cout<<"\tTrain Number : "<<this->train\_number<<endl;

cout<<"\tTrain Name : "<<this->train\_name<<endl;

cout<<"\tTrain Destination : "<<this->train\_destination<<endl;

cout<<"\tTrain Time : "<<this->train\_time<<endl;

}

int disp(train y)

{

return y.train\_number;

}

};

int main()

{

int ch;

int a=0;

int b;

train t[100];

while(1)

{

cout<<" Menu"<<endl;

cout<<" \t ====="<<endl;

cout<<" 1. Input"<<endl;

cout<<" 2. Search"<<endl;

cout<<" 3. Display"<<endl;

cout<<" enter your choice :";

cin>>ch;

switch(ch)

{

case 1:

t[a].input();

a++;

break;

case 2:

cout<<"\t\tEnter Train Number :"; cin>>b;

for(int i=0;i<a;i++)

{

if(t[i].disp(t[i])==b)

{

t[i].display();

}

}

break;

case 3:

for(int i=0;i<a;i++)

{

t[i].display();

}

break;

default:

cout<<"invalid input "<<endl;

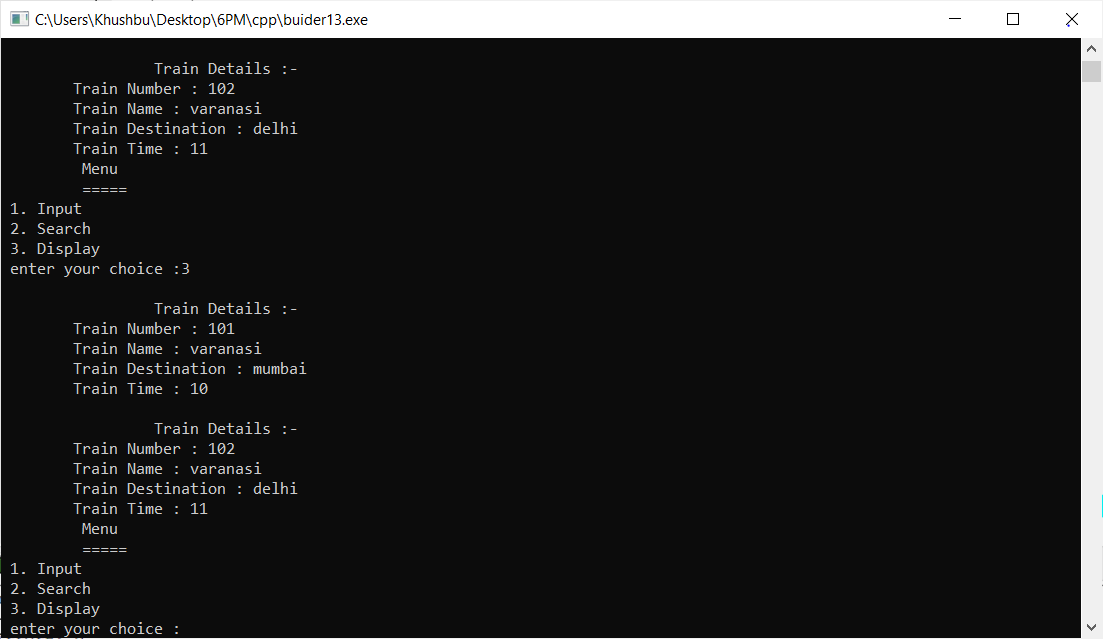
}

}

return 0;

}

**Output :**

****

**Practical : 14**

**Aim :** WAP to make Supermarket Billing System.

Requirements:

(A) Verify User Id And Password

(B) User Input Item Number , Item Name , Quantity , Tax , Discount.

(C) Display All Record In Ascending order (by item Number)

**Program :**

#include<iostream>

using namespace std;

class Login

{

private:

int userid;

int password;

int uid;

int pass;

public:

void singup()

{

cout<<"\t\t\tSingUp :- "<<endl;

cout<<"\tUser Id : "; cin>>userid;

cout<<"\tPassWord : "; cin>>password;

}

int login()

{

cout<<"\t\t\tLogIn :- "<<endl;

cout<<"\tUser Id : "; cin>>uid;

cout<<"\tPassWord : "; cin>>pass;

if(userid==uid && password==pass)

{

return 1;

}

else

{

return 0;

}

}

};

class Billing

{

private:

char name[100];

int item\_number;

char item\_name1[100];

int item\_prize;

int quantity;

int tex;

int discout;

int total;

public:

void input()

{

cout<<"\t\t\tCreate Bill :- "<<endl<<endl;

cout<<"\tCoustomer name :- "; cin>>name;

cout<<"\tItem Number :- "; cin>>item\_number;

cout<<"\tItem Name :- "; cin>>item\_name1;

cout<<"\tItem Prize :- "; cin>>item\_prize;

cout<<"\tQuantity :- "; cin>>quantity;

cout<<"\tTex :- "; cin>>tex;

cout<<"\tDiscout :- "; cin>>discout;

total=(item\_prize\*quantity);

total=total-((total\*discout)/100);

total=total+((total\*tex)/100);

}

void display()

{

cout<<"\t\t\tBill Details :- "<<endl;

cout<<"\tCoustomer name :- "<<name<<endl;

cout<<"\tItem Number :- "<<item\_number<<endl;

cout<<"\tItem Name :- "<<item\_name1<<endl;

cout<<"\tItem Prize :- "<<item\_prize<<endl;

cout<<"\tQuantity :- "<<quantity<<endl;

cout<<"\tTex :- "<<tex<<endl;

cout<<"\tDiscout :- "<<discout<<endl;

cout<<"\t -----------"<<endl;

cout<<"\tTotal BILL :- "<<total<<endl;

}

int item()

{

return item\_number;

}

};

int main()

{

int x,ch;

Billing temp;

Billing s[100];

int a=0;

Login l;

l.singup();

again:

x=l.login();

if(x)

{

while(1)

{

system("cls");

cout<<"\t\t\tEnter 1 For Create Bill "<<endl;

cout<<"\t\t\tEnter 2 For Display all BILL"<<endl;

cout<<"\t\t\tEnter 3 For Logout"<<endl;

cout<<"\t\t\t\t"; cin>>ch;

system("cls");

switch(ch)

{

case 1:

s[a].input();

a++;

system("cls");

break;

case 2:

for(int i=0;i<a-1;i++)

{

for(int j=i;j<a-1 ;j++)

{

if(s[i].item()>s[j+1].item())

{

temp=s[i];

s[i]=s[j+1];

s[j+1]=temp;

}

}

}

case 3:

goto again;

break;

default:

cout<<"\t\t\tEnter Valid Choise :"<<endl;

}

}

}

else

{

cout<<"\tEnter Valid UserId And PassWord"<<endl;

}

return 0;

}

**Practical : 15**

**Aim :**  WAP which perform Addition of members of two different classes using friend Function.

**Program :**

#include<iostream>

using namespace std;

class Sum {

private:

int x, y, z;

public:

void input() {

cout << "Enter 1st numbers:";

cin >> x;

cout << "Enter 2nd numbers:";

cin>>y;

}

friend void add(Sum &s);

void display() {

cout << "sum is:" << z;

}

};

void add(Sum &s) {

s.z = s.x + s.y;

}

int main() {

Sum s;

s.input();

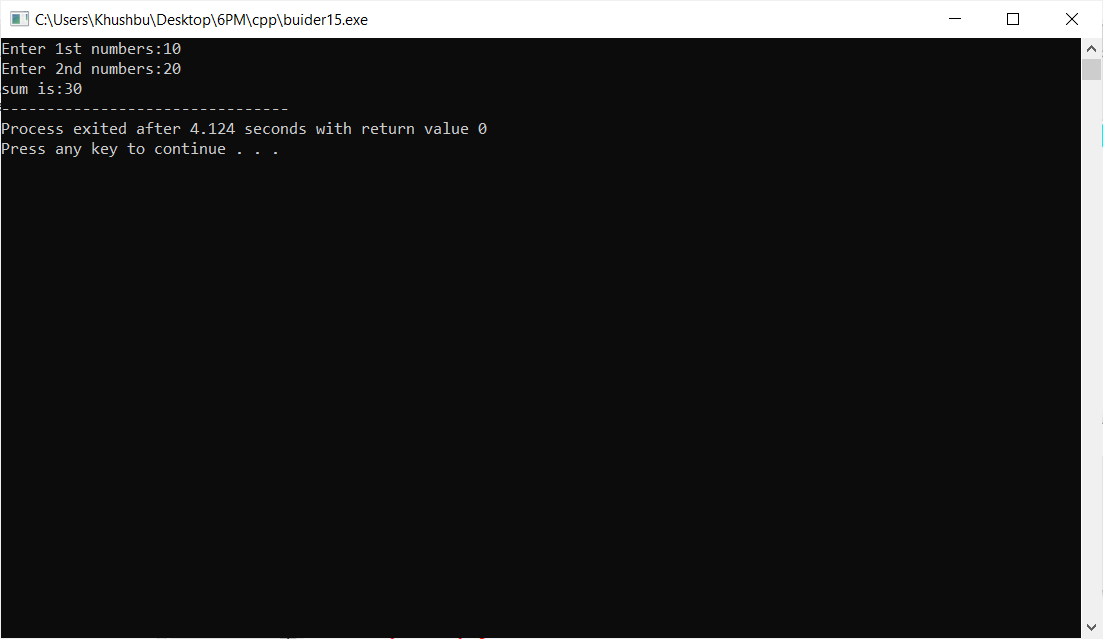
add(s);

s.display();

return 0;

}

**Output :**

****

**Practical : 16**

**Aim :** WAP to make Bank Management System. (like shown in below images)

**Program :**

#include<iostream>

#include<stdio.h>

#include<string.h>

using namespace std;

class Administrator

{

private:

int id;

char name[100];

char branch[100];

public:

void setData()

{

cout<< " Enter Your ID : ";

cin >> this->id;

cout<< " Enter Your Name : ";

cin >> this->name;

cout<< " Enter Your Branch : ";

cin >> this->branch;

}

void getData()

{

cout <<"\t\t--------------------------------------------------------------------------------\t\t\t\t\t";

cout <<"\t\t Your ID :" << this->id <<endl;

cout <<"\t\t\t\t Your Name :" << this->name <<endl;

cout <<"\t\t\t\t Your Branch :" << this->branch <<endl;

cout <<"\t\t--------------------------------------------------------------------------------\t\t\t\t\t"<<endl<<endl;

cout <<"\t\t:\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t\t\t\t\t"<<endl;

cout << "\t\t\t\t\t login as Administrator Sucesfull..."<<endl<<endl;

cout <<"\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t\t\t\t\t"<<endl;

}

};

class Staff

{

private:

int id;

char name[100];

char email[100];

int number;

public:

void setData()

{

cout<< " Enter Your ID : ";

cin >> this->id;

cout<< " Enter Your Name : ";

cin >> this->name;

cout<< " Enter Your Email : ";

cin >> this->email;

cout<< " Enter Your Phone no. : ";

cin >> this->number;

}

void getData()

{

cout <<"\t\t--------------------------------------------------------------------------------\t\t\t\t\t";

cout <<"\t\t# Your ID : " << this->id <<endl;

cout <<"\t\t\t\t# Your Name : " << this->name <<endl;

cout <<"\t\t\t\t# Your Email : " << this->email <<endl;

cout <<"\t\t\t\t# Your Phone No. : " << this->number <<endl;

cout <<"\t\t--------------------------------------------------------------------------------\t\t\t\t\t";

}

void withdraw()

{

int m,t=10000,total;

cout <<"\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t\t\t\t\t"<<endl;

cout << "\t\t\t\t\* Enter Withdrawal Money :";

cin >> m;

total=t-m;

cout <<endl<<"\t\t\t\t\* Congratulation Withdraw successfully..."<<endl<<endl;

cout <<"\t\t\t\t\* Final amount in your wallet = "<<total<<endl<<endl;

cout <<"\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t\t\t\t\t"<<endl;

}

void deposite()

{

int m,t=10000,total;

cout <<"\t\t:\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t\t\t\t\t"<<endl;

cout << "\t\t\t\t\* Enter Deposite Money :";

cin >> m;

total=t+m;

cout <<endl<<"\t\t\t\t\* Congratulation Deposite Sucessful..."<<endl<<endl;

cout <<"\t\t\t\t\* Final amount in your wallet = "<<total<<endl<<endl;

cout <<"\t\t::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::\t\t\t\t\t"<<endl;

}

void balance()

{

cout <<"\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t\t\t\t\t"<<endl;

cout << "\t\t\t\t\tTotal Balanace = 10000/-"<<endl<<endl;

cout <<"\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t\t\t\t\t"<<endl;

}

};

class Bank

{

Administrator a1;

Staff s1;

public:

void info()

{

cout <<endl<<endl;

cout <<"\t\t=====================================================================================\t\t\t\t\t";

cout <<endl;

cout << "\t\t\t\t\t -: WELCOME :-\t\t\t\t\t\t ";

cout <<endl;

cout <<"\t\t====================================================================================\t\t\t\t\t";

cout <<endl;

cout <<"\t\t====================================================================================\t\t\t\t\t";

cout <<endl;

cout << "\t\t\t\t -: Designed By : khushbu khakhriya :-\t\t\t\t\t\t ";

cout <<endl;

cout <<"\t\t====================================================================================\t\t\t\t\t";

}

void category()

{

int d,n;

cout <<"\t\t--------------------------------------------------------------------------------\t\t\t\t\t";

cout << "\t\t\t Choose Your Category -\t\t\t\t\t\t "<<endl;;

cout<< "\t\t\t\t\t => Enter [1] For Administrator."<<endl;

cout<< "\t\t\t\t\t => Enter [2] For Customer."<<endl;

cout <<"\t\t--------------------------------------------------------------------------------\t\t\t\t\t"<<endl;

cout<< " Enter Your Choise : ";

cin >> d;

switch(d)

{

case 1:

{

cout <<"\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t\t\t\t\t";

cout << "\t\t\t ADMINISTRATOR \t\t\t\t\t\t\t "<<endl;

cout <<"\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t\t\t\t\t"<<endl;

a1.setData();

a1.getData();

break;

}

case 2:

{

cout <<"\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t\t\t\t\t";

cout << "\t\t\t CUSTOMER \t\t\t\t\t\t\t "<<endl<<endl;

cout <<"\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t\t\t\t\t"<<endl;

s1.setData();

s1.getData();

cout<< "\t\t\t Enter [1] For Withdraw Money."<<endl;

cout<< "\t\t\t\t\t Enter [2] For Deposite Money."<<endl;

cout<< "\t\t\t\t\t Enter [3] For Check Balance."<<endl;

cout<< " Enter Your Choise : ";

cin >> n;

cout <<endl<<endl;

switch(n)

{

case 1:

{

s1.withdraw();

break;

}

case 2:

{

s1.deposite();

break;

}

case 3:

{

s1.balance();

break;

}

default:

{

cout <<"invalid choice";

}

}

break;

}

default:

{

cout << "Invalid Choise..";

break;

}

}

}

};

int main()

{

char i[100];

Bank s1;

s1.info();

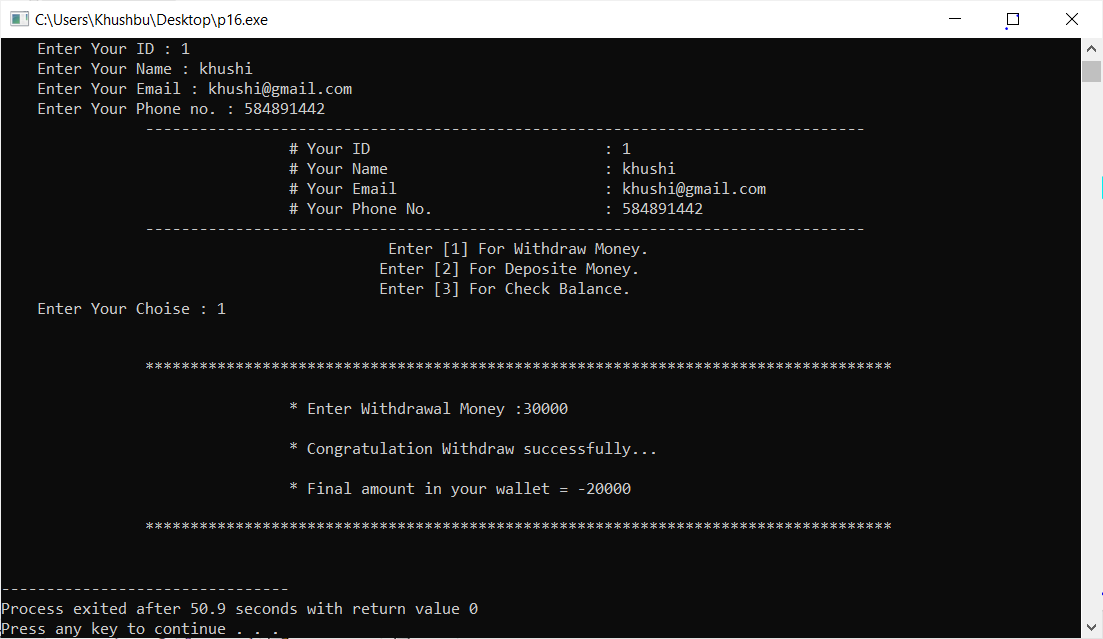
cout <<endl<< " \t\t\tPress Any Key For Continue... ";

cin >> i;

s1.category();

}

**Output :**

****