```
//Assignment 1
#include <iostream>
using namespace std;
class bank_acc{
  long long int acc_no,wd,bal,depo, limit = 10000;
  int n;
  string name ;
  public:
  void read(){
      cout<<"enter account number ";
      cin>>acc_no;
      cout<<"enter your name ";
      cin>>name;
      cout<<"enter your balance ";
      cin>>bal;
      if(bal<10000){
          cout<<"you need to keep minimum balance of 10000"<<endl;
          exit(0);
      }
      cout<<"\n\n";
  }
  void display(){
     cout<<"account number is "<<acc_no<<endl;
      cout<<"your name "<< name<<endl;
      cout<<"your balance"<< bal<<endl;
      cout<<"\n";
  }
  void deposit (){
      cout<<"enter the amount that you want to deposit"<<endl;
      cin>>depo;
      bal= bal+ depo;
```

```
   cout<<"current balnce = "<<bal<<endl;
 }
 void withdraw(){
   cout<<"enter the amount that you want to withdraw "<<endl;
   cin>>wd;
   if(wd>bal){
     cout<<"insufficient balance"<<endl;
     cout<<"do you want to take loan?";

     exit(0);
   }
   if(bal - wd <= limit ){
     cout<<"you cant withdraw "<< wd <<"as you need to maintain minimum balance of 10000"<< endl;
   }
   else{
     cout<<wd << "debited";
   }
   bal= bal - wd;
   cout<<"current balnce = "<<bal<<endl;
 }
  int getacc_no(){
   return acc_no;
 }

//  ~bank_acc(){
//    acc_no=0;
//    wd=0;
//    bal=0;
//    depo=0;
//    name = "NULL";
//    // cout<<"\n Destructor executed"<<endl;
//    // display();
//  }
//  bank_acc(){
//    //default ocnstructor
//  }
//  bank_acc(bank_acc &a){
//    acc_no = a.acc_no;
//    wd = a.wd;
//    bal = a.bal;
//    depo = a.depo;
//    //cop constructor
//  }
};
   int find(bank_acc o[1000],int acc){
     int l,i,n;
     for(int i=0;i<n;i++){
         if(acc==o[i].getacc_no()){
             l=i;
         }
     }
     return l;
  }

int main()
{
   bank_acc o[1000];
   int n;
   cout<<"enter number of user"<<endl;
   cin>>n;
   for(int i=0;i<n;i++){
```

```
   o[i].read();

   }

      int acc;

      cout<<"enter account number that you want to search"<<endl;

      cin>>acc;

      int search_result = find(o,acc);

      cout<<"account number is present at "<<search_result<< " position"<<endl<<endl;

   int ch;

   do{

   cout<<"do you want to display info, deopsit money, withdraw money or exit "<<endl;

   cout<<"enter 0 to display information, 1 to deposit money, 2 to withdraw money and 3 to
exit"<<endl;

   cin>>ch;

   if(ch==0){

      o[search_result].display();

   }

   else if(ch==1){

      o[search_result].deposit();

   }

   else if(ch==2){

      o[search_result].withdraw();

   }

   else if(ch==3){

      exit(0);

   }

   else{

      cout<<"An unexpected error occured"<<endl;

   }

   }while(1);

   return 0;

}
```

```
//Assignment 2
#include<iostream>
using namespace std;
class student{
    string name, div, blood_grp, address;
    int  roll_no, Class, dob, telephone_no;
    public:

    inline void read(){
      cout<<"Enter name: ";
      cin>>name;
      cout<<"Enter address: ";
      cin>>address;
      cout<<"Enter class: ";
      cin>>Class;
      cout<<"Enter D.O.B: ";
      cin>>dob;
      cout<<"Enter blood group: ";
      cin>>blood_grp;
      cout<<"Enter phone no: ";
      cin>>telephone_no;
      cout<<"Enter division: ";
      cin>>div;
    }

    inline void print(){
    cout<<"Name: "<<name<<endl;
    cout<<"Address: "<<address<<endl;
    cout<<"Class: "<<Class<<endl;
    cout<<"Division: "<<div<<endl;
    cout<<"Roll No: "<<roll_no<<endl;
```

```
  cout<<"D.O.B: "<<dob<<endl;
  cout<<"Blood Group: "<<blood_grp<<endl;
  cout<<"Phone No: "<<telephone_no<<endl;
  }

  student(){
   roll_no = 0;
   name = "null";
   address = "null";
   Class = 0;
   dob = 0;
   blood_grp = "null";
   div = "null" ;
   telephone_no = 0;
  }
  ~student(){
   roll_no = 0;
   name = "null";
   address = "null";
   Class = 0;
   dob = 0;
   blood_grp = "null";
   div = "null" ;
   telephone_no = 0;
  }
};


int main()
{
  cout<<"\n # Student Database System # \n"<<endl;
```

```
    static int n;
    cout<<"enter number of students ";
    cin>>n;
    student o[n];
    for(int i=0;i<n;i++)
    {
      o[i].read();
    }
    for(int i=0;i<n;i++)
    {
      o[i].print();
    }
return 0;
}
```

```
//Assignment 3
#include <iostream>
using namespace std;

class vehicle{
   public:
   int milege,price;
};


class car:public vehicle{
   public:
   int ownership_cost, warranty , seating_capacity;
   string fuel_type;
};
class bike:public vehicle{
   public:
   int no_cylinders, no_gears;
   string cooling_type, wheel_type;
   int fuel_tank_size;
};


class audi:public car{
   public:
   string model_type;
   void read_audi();
   void display_audi();
};
class ford:public car{
   public:
    string model_type;

    void read_ford();

    void display_ford();

};



class bajaj:public bike{

    public:

    string make_type;

};

class tvs:public bike{

    public:

    string make_type;

};



void audi :: read_audi(){

    cout<<"welcome to Audi"<<endl;

    cout<<"enter model type: ";

    cin>>model_type;

    cout<<"enter owernship cost";

    cin>>ownership_cost;

    cout<<"enter warrenty in year";

    cin>>warranty;

    cout<<"enter seating capacity";

    cin>>seating_capacity;

    cout<<"enter fuel type (diesel or petrol)";

    cin>>fuel_type;

    cout<<"enter milege";

    cin>>milege;

    cout<<"enter price of the vehicle";
    cin>>price;
}

void ford :: read_ford(){
    cout<<endl;
    cout<<"welcome to ford "<<endl;
    cout<<"enter model type: ";
    cin>>model_type;
    cout<<"enter owernship cost";
    cin>>ownership_cost;
    cout<<"enter warrenty in year";
    cin>>warranty;
    cout<<"enter seating capacity";
    cin>>seating_capacity;
    cout<<"enter fuel type (diesel or petrol)";
    cin>>fuel_type;
    cout<<"enter milege";
    cin>>milege;
    cout<<"enter price of the vehicle";
    cin>>price;
}

void audi :: display_audi(){
    cout<<endl;
    cout<<"model type is "<<model_type<<endl;
    cout<<"owernship cost is "<<ownership_cost<<endl;
    cout<<"warrenty in year is "<<warranty<<endl;
    cout<<"seating capacity is "<<seating_capacity<<endl;
    cout<<"fuel type (diesel or petrol) is "<<fuel_type;
    cout<<"milege is "<<milege<<endl;
    cout<<"price of the vehicle is "<<price<<endl;
```

```
}

void ford :: display_ford(){
   cout<<endl;
   cout<<"model type is "<<model_type<<endl;
   cout<<"owernship cost is "<<ownership_cost<<endl;
   cout<<"warrenty in year is "<<warranty<<endl;
   cout<<"seating capacity is "<<seating_capacity<<endl;
   cout<<"fuel type (diesel or petrol) is "<<fuel_type;
   cout<<"milege is "<<milege<<endl;
   cout<<"price of the vehicle is "<<price<<endl;
}

int main()
{
   audi o1;
   ford o2;
   o1.read_audi();
   o1.display_audi();
   o2.read_ford();
   o2.display_ford();



   return 0;
}
```

```
//Assignment 4

#include <iostream>

using namespace std;

class complex{

   int real,imag;

   public:


   complex(){

      real=0;

      imag=0;

   }

   complex(int r,int i){

      real=r;

      imag=i;

   }



   complex operator+(complex c1){

      complex ret;

      ret.real=real+c1.real;

      ret.imag=imag+c1.imag;

      return ret;

   }

   complex operator*(complex c1){

      complex ret;

      ret.real=real*c1.real - imag*c1.imag;

      ret.imag=real*c1.imag + imag*c1.real;

      return ret;

   }

   friend ostream & operator<< (ostream& out,  complex& c);

   friend istream & operator>> (istream& din,  complex& c);
};
   ostream & operator<< (ostream& out,  complex& c)
   {
   out<<"("<<c.real<<"+"<<c.imag<<"i)";
   return out;
   }


   istream& operator>> (istream &din, complex &c){
   cout<<"Enter :";
   din>>c.real;
   din>>c.imag;
   return din;
   }
int main()
{
   complex c1,c2,c3,c4;
   cin>>c1;
   cin>>c2;


   c3=c1+c2;
   c4=c1*c2;


   cout<<endl<<"Addition :"<<c3<<endl;
   cout<<"Multiplication :"<<c4;




   return 0;
}
```

```
//assignment 5

#include<iostream>

using namespace std;

class Shape

{

public:

double length,breadth;



Shape()

{

length=0;

breadth=0;

}



void get_data()

{

cout<<"\nEnter length :"<<endl;

cin>>length;

cout<<"\nEnter Breadth :"<<endl;

cin>>breadth;

}



virtual void display_area()

{

}

};
```

```
class Triangle : public Shape
{
public:

void getdata(){
  Shape::get_data();
}

void display_area(){
cout<<"\nArea of Triangle = "<<(length*breadth)/2<<endl;
}
};


class Rectangle : public Shape
{
public:

void getdata(){
     Shape::get_data();
}

void display_area(){
cout<<"\nArea of Rectangle = "<<(length*breadth)<<endl;
}
};


int main(){
    int ch;
    Shape *base_ptr;
    Triangle tr;
    Rectangle rec;



    cout<<"\n1.triangle\n2.rectangle\n";
    cout<<"Enter your choice:"<<endl;
    cin>>ch;

    switch(ch){

        case 1:
        base_ptr=&tr;
        tr.get_data();
        tr.display_area();
        break;

         case 2:
        base_ptr=&rec;
        rec.get_data();
        rec.display_area();
        break;

    }
    return 0;
}
```