|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| /\*Write a program to implement a class ‘Box’ with data members (Length, Breadth,  Height). Include different constructors to initialize data members and also include  members functions to compute surface area and volume of each box objects\*/  //22co14 Falgun Kole 13/10/2023  #include<iostream>  using namespace std;  class box  {      int length,breadth,height;      public:      box(int x,int y,int z)      {          length=x;          breadth=y;          height=z;      }  float surface\_area()      {          cout<<"surface area of box="<<2\*length\*breadth+2\*length\*height+2\*breadth\*height<<endl;      }      float volume()      {          cout<<"volume of box="<<length\*breadth\*height<<endl;      }  };  int main()  {      box b(10,10,10);      b.volume();      b.surface\_area();      return 0;  }  Output:   |  | | --- | | volume of box=1000  surface area of box=600 |   /\*Write a program to implement class ‘Date’ with data members(date,month,year).  Include different constructors to initialize data members and also include members  functions to display each date object in ‘dd-mm-yyyy’ format.\*/  //22co14 Falgun Kole 13/10/2023  #include<iostream>  using namespace std;  class Date  {      public:      int day,month,year;      Date(int d,int m,int y)      {          day=d;          month=m;          year=y;      }      void format()      {          cout<<"Date:"<<day<<"-"<<month<<"-"<<year<<endl;      }  };  int main()  {      Date d(20,10,2004),d1(30,12,2022),d2(29,02,2023);      d.format();      d1.format();      d2.format();      return 0;  }  Output:   |  | | --- | | Date:20-10-2004  Date:30-12-2022  Date:29-2-2023 |   /\*Write a program to implement a class ‘Solid’ with data members (radius,height).  Include different constructors to initialize data members and also include members  functions to compute volume of cylinder and cone. Include default argument for the  data member ‘radius’.\*/  //22co14 Falgun Kole 13/10/2023  #include<iostream>  using namespace std;  class solid  {      int radius,height;      solid()          {radius=0;}      public:      solid(int r,int h)      {          radius=r;          height=h;      }      float volume\_cylinder()      {          cout<<"\nvolume of cylinder="<<3.1415\*radius\*radius\*height<<endl;      }      float volume\_cone()      {          cout<<"volume of cone= "<<0.333\*3.1415\*radius\*radius\*height<<endl;      }  };  int main()  {      solid s1(5,10),s2(10,10),s3(20,50);      s1.volume\_cylinder();      s1.volume\_cone();      s2.volume\_cylinder();      s2.volume\_cone();      s3.volume\_cylinder();      s3.volume\_cone();      return 0;  }  Output:   |  | | --- | | volume of cylinder=785.375  volume of cone= 261.53  volume of cylinder=3141.5  volume of cone= 1046.12  volume of cylinder=62830  volume of cone= 20922.4 |   /\*Write a program to print the details of students by creating a Student class. If no data is  passed while creating an object of the Student class, then default values should be  assigned. Include different constructors to initialize data members and also include  members functions to display student details\*/  //falgun kole 22co14 13/10/2023  #include<iostream>  #include<iomanip>  using namespace std;  class student  {      int rollno,fee;      string branch,name;      public:      void setRollno(int r){rollno=r;}      void setFee(int f){fee=f;}      void setName(string n){name=n;}      void setBranch(string b){branch=b;}  //default constructor      student()      {          rollno=0;          fee=00000;          name="BLANK";          branch="BLANK";      }  //parameterised constructor      student(int r,int f,string n,string b)      {          fee=f;          rollno=r;          name=n;          branch=b;      }  // copy constructor      student(student &s)      {          fee=s.fee;          rollno=s.rollno;          name=s.name;          branch=s.branch;      }  //display function      void display()      {          cout<<"NAME:"<<name<<endl<<"ROLL NO:"<<rollno<<endl<<"BRANCH:"<<branch<<"\nFEE:"<<fee<<endl<<endl;      }  };  int main()  {      int roll,fee;      string name,branch;      student s1(10,12000,"Rahul","COMP"),s2,s3;      s1.display();      s2.display();      cout<<"enter student name, roll no,department and fee :"<<endl;      cin>>name>>roll>>branch>>fee;      s3.setName(name);      s3.setRollno(roll);      s3.setBranch(branch);      s3.setFee(fee);      s3.display();      student s4(s1);      s4.display();      return 0;  }  Output:   |  | | --- | | NAME:Rahul  ROLL NO:10  BRANCH:COMP  FEE:12000  NAME:BLANK  ROLL NO:0  BRANCH:BLANK  FEE:0  enter student name, roll no,department and fee :  falgun  1029  comp  21344  NAME:falgun  ROLL NO:1029  BRANCH:comp  FEE:21344  NAME:Rahul  ROLL NO:10  BRANCH:COMP  FEE:12000 |   /\*Write a program to implement a class ‘Complex Nos’ with data members(real,imag).  Include parameterised and copy constructors to initialize data members and also  include members functions to compute and display the sum and difference of complex  nos.\*/  //22co14 Falgun Kole 13/10/2023  #include <iostream>  using namespace std;  class ComplexNos  {      int real;      int imag;  public:      // Parameterized constructor      ComplexNos(int r, int i)      {      real=r;      imag=i;      }      // Copy constructor      ComplexNos(const ComplexNos& s)      {          real=s.real;          imag=s.imag;      }      // Member function to compute the sum of two complex numbers      ComplexNos add(const ComplexNos& s)      {          int sumReal = real + s.real;          int sumImag = imag + s.imag;          return ComplexNos(sumReal, sumImag);      }      // Member function to compute the difference of two complex numbers      ComplexNos subtract(const ComplexNos& s)      {          int diffReal = real - s.real;          int diffImag = imag - s.imag;          return ComplexNos(diffReal, diffImag);      }      // Member function to display the complex number      void display()      {          cout << "" << real << " + " << imag << "i" <<endl;      }  };  int main()  {      // Create two complex numbers      ComplexNos num1(3.0, 4.0);      ComplexNos num2(1.0, 2.0);      // Display the original complex numbers      cout << "Complex Number 1: ";      num1.display();      cout << "Complex Number 2: ";      num2.display();      //display the sum and diff of the complex numbers      ComplexNos sum = num1.add(num2);      ComplexNos diff = num1.subtract(num2);      cout << "Sum: ",sum.display();      cout << "Difference: ",diff.display();      return 0;  }  Output:   |  | | --- | | Complex Number 1: 3 + 4i  Complex Number 2: 1 + 2i  Sum: 4 + 6i  Difference: 2 + 2i |   /\*Write a program demonstrating use of destructrors.\*/  //22co14 Falgun Kole 13/10/2023  #include<iostream>  using namespace std;  class Point  {      int x,y;      public:          Point(int a,int b)          {              x=a;              y=b;          }          ~Point()          {              cout<<"Object is destroyed"<<endl;          }          void display()          {              cout<<"x= "<<x<<" "<<"y= "<<y<<endl;          }  };  int main()  {      int x,y;      cout<<"Enter value of x and y: "<<endl;      cin>>x>>y;      Point p1(x,y);      {          int x,y;          cout<<"Enter value of x and y: "<<endl;          cin>>x>>y;          Point p2(x,y);          p2.display();      }      p1.display();  }  Output:   |  | | --- | | Enter value of x and y:  90  80  Enter value of x and y:  34  44  x= 34 y= 44  Object is destroyed  x= 90 y= 80  Object is destroyed | |