**1:Write a program to create student class with data members rollno, marks1,mark2,mark3.**

**Accept data (acceptInfo()) and display using display member function.**

**Also display total,percentage and grade.**

#include<iostream>

using namespace std;

class student{

private:

int rollNo;

double marks1, marks2, marks3, sum, per;

public:

void getRollno(int rollNo)

{

this->rollNo=rollNo;

}

void getMarks(double marks1, double marks2, double marks3){

this->marks1=marks1;

this->marks2=marks2;

this->marks3=marks3;

}

void display()

{

cout<<"RollNo. : "<<rollNo<<endl;

cout<<"Marks : "<<marks1<<" "<<marks2<<" "<<marks3<<endl;

//cin>>marks1>>marks2>>marks3

}

void calTotalMarks(){

sum = marks1 + marks2 + marks3;

cout<<"Total marks is "<<sum<<endl;

}

void percent()

{

per = (sum/300)\*100;

cout<<"Total percentage is: "<<per<<endl;

}

void showGrade(){

if(per>=85 && per<=100)

{

cout<<"Grade is A";

}

else if(per>=65 && per<=85)

{

cout<<"Grade is B";

}

else if(per>=45 && per<=65)

{

cout<<"Grade is C";

}

else{

cout<<"Grade is F";

}

}

};

int main()

{

student obj;

int rollNo;

double m1, m2, m3;

cout<<"Enter the roll no. :"<<endl;

cin>>rollNo;

obj.getRollno(rollNo);

cout<<"Enter the marks: "<<endl;

cin>>m1>>m2>>m3;

obj.getMarks(m1,m2,m3);

obj.display();

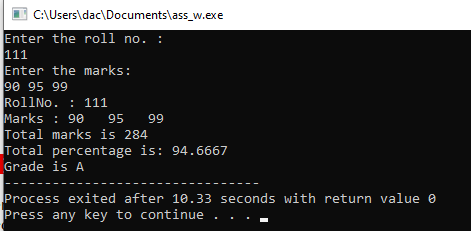
obj.calTotalMarks();

obj.percent();

obj.showGrade();

return 0;

}

****

**2. Create a class Person with data members as name, age, city. Write getters and setters for all the data**

**members. Also add the display function. Create Default and Parameterized constructors. Create the**

**object of this class in main method and invoke all the methods in that class.**

**Using parameterized consturctor**

#include<iostream>

using namespace std;

class person{

private:

int age1;

string name, city;

public:

person()

{

cout<<"Default constr called"<<endl;

age1= 18;

name= "default";

city = "abc";

}

person(int age, string name, string city){

cout<<"parameterize constr called"<<endl;

this->age1=age;

this->name=name;

this->city=city;

}

void setName(string name)

{

this->name= name;

}

void setAge(int age){

this->age1= age;

}

void setCity(string city){

this->city=city;

}

int getAge()

{

return age1;

}

string getName()

{

return name;

}

string getCity()

{

return city;

}

void display()

{

cout<<name<<" "<<age1<<" "<<city<<endl;

}

};

int main()

{

int age;

string name,city;

person p1;

p1.display();

cout<<"Enter your age, name, city"<<endl;

cin>>age>>name>>city;

// p1.setAge(age);

// p1.setName(name);

// p1.setCity(city);

// person p2(33,"abc","xyz");

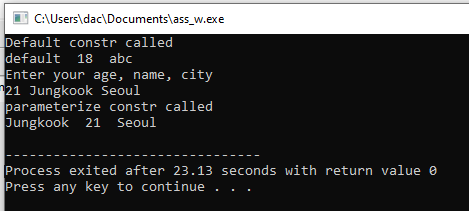
person p2(age,name,city);

p2.display();

// cout<<"Age is:"<<p1.getAge()<<endl;

return 0;

}

****

**Using getters and setters**

#include<iostream>

using namespace std;

class person{

private:

int age1;

string name, city;

public:

person()

{

cout<<"Default constr called"<<endl;

age1= 18;

name= "default";

city = "abc";

}

person(int age, string name, string city){

cout<<"parameterize constr called"<<endl;

this->age1=age;

this->name=name;

this->city=city;

}

void setName(string name)

{

this->name= name;

}

void setAge(int age){

this->age1= age;

}

void setCity(string city){

this->city=city;

}

int getAge()

{

return age1;

}

string getName()

{

return name;

}

string getCity()

{

return city;

}

void display()

{

cout<<name<<" "<<age1<<" "<<city<<endl;

}

};

int main()

{

int age;

string name,city;

person p1;

p1.display();

cout<<"Enter your age, name, city"<<endl;

cin>>age>>name>>city;

p1.setAge(age);

p1.setName(name);

p1.setCity(city);

// person p2(33,"abc","xyz");

// person p2(age,name,city);

// p2.display();

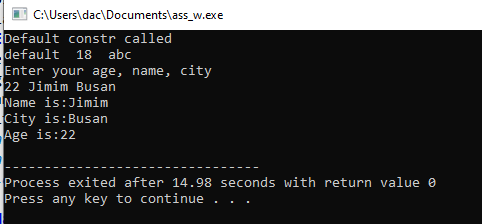
cout<<"Name is:"<<p1.getName()<<endl;

cout<<"City is:"<<p1.getCity()<<endl;

cout<<"Age is:"<<p1.getAge()<<endl;

return 0;

}

****

**3. Create a class Date with data members as dd, mm, yy. Write getters and setters for all the data members. Also add the display function. Create Default and Parameterized constructors. Create the object of this class in main method and invoke all the methods in that class.**

**Using parameterized consturctor**

#include<iostream>

using namespace std;

class date{

private:

int day, month, year;

public:

void setDay(int day){

this->day=day;

}

void setMonth(int month){

this->month= month;

}

void setYear(int year){

this->year= year;

}

int getDay(){

return day;

}

int getMonth(){

return month;

}

int getYear(){

return year;

}

void display(){

cout<<"Date : "<<day<<" - "<<month<<" - "<<year;

}

};

int main()

{

int day, month, year;

date dt;

cout<<"Enter date (Day-Month-Year) : ";

cin>>day>>month>>year;

dt.setDay(day);

dt.setMonth(month);

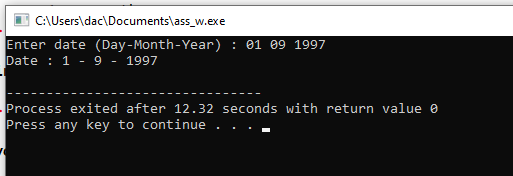
dt.setYear(year);

// dt.display(); //print by dispaly method

cout<<"Date : "<<dt.getDay()<<" - "<<dt.getMonth()<<" - "<<dt.getYear()<<endl; //print by getter methods

return 0;

}

****

**Using getters and setters**

#include<iostream>

using namespace std;

class date{

private:

int day, month, year;

public:

void setDay(int day){

this->day=day;

}

void setMonth(int month){

this->month= month;

}

void setYear(int year){

this->year= year;

}

int getDay(){

return day;

}

int getMonth(){

return month;

}

int getYear(){

return year;

}

void display(){

cout<<"Date : "<<day<<" - "<<month<<" - "<<year;

}

};

int main()

{

int day, month, year;

date dt;

cout<<"Enter date (Day-Month-Year) : ";

cin>>day>>month>>year;

dt.setDay(day);

dt.setMonth(month);

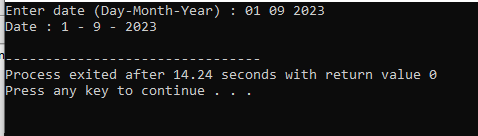
dt.setYear(year);

// dt.display(); //print by dispaly method

cout<<"Date : "<<dt.getDay()<<" - "<<dt.getMonth()<<" - "<<dt.getYear()<<endl; //print by getter methods

return 0;

}

****

**4. Create a class Book with data members as bname,id,author,price. Write getters and setters for all the data members. Also add the display function. Create Default and Parameterized constructors. Create the object of this class in main method and invoke all the methods in that class.**

**Using getters and setters**

#include<iostream>

using namespace std;

class Book{

private:

int id,price;

string bname,author;

public:

Book(){

//cout<<"Default constr called"<<endl;

id=101;

price= 600;

bname = "abc";

}

Book(int id, string bname, string author, int price){

cout<<"parameterize const. is called"<<endl;

this->id=id;

this->bname = bname;

this->author= author;

this->price = price;

}

void setBookId(int id){

this->id=id;

}

void setBookName(string bname){

this->bname = bname;

}

void setAuthor(string author){

this->author = author;

}

void setPrice(int price){

this->price = price;

}

int getBookId(){

return id;

}

int getPrice(){

return price;

}

string getBookName(){

return bname;

}

string getAuthorName(){

return author;

}

void display(){

cout<<id<<" "<<bname<<" "<<author<<" "<<price<<endl;

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

}

};

int main()

{

// Book bk1;// default constructor is called

// bk1.display();

int id, price;

string bname, author;

cout<<"Enter Bookid BookName Author Price: ";

cin>>id>>bname>>author>>price;

// Book bk2(id, bname, author, price); //parameterize const. is called

// bk2.display();

Book bk3;

bk3.setBookId(id);

bk3.setBookName(bname);

bk3.setAuthor(author);

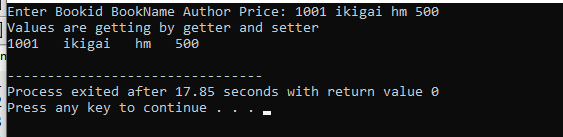
bk3.setPrice(price);

cout<<"Values are getting by getter and setter"<<endl;

cout<<bk3.getBookId()<<" "<<bk3.getBookName()<<" "<<bk3.getAuthorName()<<" "<<bk3.getPrice()<<endl;

return 0;

}

****

**Using parameterized constructor**

#include<iostream>

using namespace std;

class Book{

private:

int id,price;

string bname,author;

public:

Book(){

//cout<<"Default constr called"<<endl;

id=101;

price= 600;

bname = "abc";

}

Book(int id, string bname, string author, int price){

cout<<"parameterize const. is called"<<endl;

this->id=id;

this->bname = bname;

this->author= author;

this->price = price;

}

void setBookId(int id){

this->id=id;

}

void setBookName(string bname){

this->bname = bname;

}

void setAuthor(string author){

this->author = author;

}

void setPrice(int price){

this->price = price;

}

int getBookId(){

return id;

}

int getPrice(){

return price;

}

string getBookName(){

return bname;

}

string getAuthorName(){

return author;

}

void display(){

cout<<id<<" "<<bname<<" "<<author<<" "<<price<<endl;

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

}

};

int main()

{

// Book bk1;// default constructor is called

// bk1.display();

int id, price;

string bname, author;

cout<<"Enter Bookid BookName Author Price: ";

cin>>id>>bname>>author>>price;

Book bk2(id, bname, author, price); //parameterize const. is called

bk2.display();

Book bk3;

bk3.setBookId(id);

bk3.setBookName(bname);

bk3.setAuthor(author);

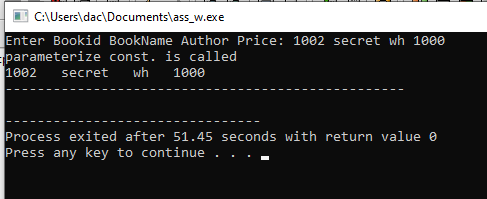
bk3.setPrice(price);

// cout<<"Values are getting by getter and setter"<<endl;

// cout<<bk3.getBookId()<<" "<<bk3.getBookName()<<" "<<bk3.getAuthorName()<<" "<<bk3.getPrice()<<endl;

return 0;

}

****

**5. Create a class Point with data members as x,y. Create Default and Parameterized constructors. Write**

**getters and setters for all the data members. Also add the display function. Create the object of this**

**class in main method and invoke all the methods in that class.**

**Using getters and setters**

#include<iostream>

using namespace std;

class Point{

private:

int x, y;

public:

Point(){

cout<<"----------------Default consturctor is called------------\n";

cout<<"Default values \n";

x = 0;

y = 0;

}

Point(int x, int y){

cout<<"----------------Parameterized consturctor is called------------\n";

this->x = x;

this->y = y;

}

void setCoordinate1(int x){

this->x = x;

}

void setCoordinate2(int y){

this->y = y;

}

int getCoordinate1(){

return x;

}

int getCoordinate2(){

return y;

}

void displayCoordinates(){

cout<<"\nCoordinates (x,y) : "<<"("<<x<<","<<y<<")"<<endl;

}

};

int main5(){

Point pt;

pt.displayCoordinates();

int x, y;

cout<<"Enter x and y coordinates : ";

cin>>x>>y;

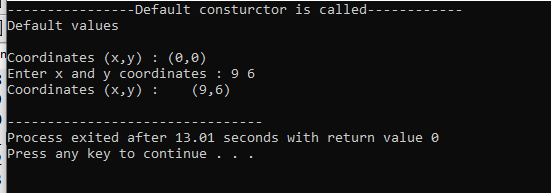
pt.setCoordinate1(x);

pt.setCoordinate2(y);

cout<<"Coordinates (x,y) : "<<"("<<pt.getCoordinate1()<<","<<pt.getCoordinate2()<<")"<<endl;

// printing values by getters and setters

}



**Using parameterized constructor**

#include<iostream>

using namespace std;

class Point{

private:

int x, y;

public:

Point(){

cout<<"----------------Default consturctor is called------------\n";

cout<<"Default values \n";

x = 0;

y = 0;

}

Point(int x, int y){

cout<<"----------------Parameterized consturctor is called------------\n";

this->x = x;

this->y = y;

}

void setCoordinate1(int x){

this->x = x;

}

void setCoordinate2(int y){

this->y = y;

}

int getCoordinate1(){

return x;

}

int getCoordinate2(){

return y;

}

void displayCoordinates(){

cout<<"\nCoordinates (x,y) : "<<"("<<x<<","<<y<<")"<<endl;

}

};

int main(){

Point pt;

pt.displayCoordinates();

int x, y;

cout<<"Enter x and y coordinates : ";

cin>>x>>y;

// pt.setCoordinate1(x);

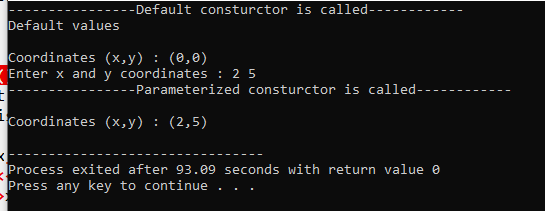
// pt.setCoordinate2(y);

// cout<<"Coordinates (x,y) : "<<"("<<pt.getCoordinate1()<<","<<pt.getCoordinate2()<<")"<<endl; // printing values by getters and setters

Point pt1(x,y);

pt1.displayCoordinates();

}

****

**6. Create a class ComplexNumber with data members real, imaginary. Create Default and Parameterized constructors. Write getters and setters for all the data members. Also add the display function. Create the object of this class in main method and invoke all the methods in that class.**

**Using getters and setters**

#include<iostream>

using namespace std;

class complexNumber{

private:

int realNum, imaginaryNum;

public:

complexNumber(){

cout<<"----------------Default consturctor is called------------\n";

realNum = 0;

imaginaryNum = 0;

}

complexNumber(int realNum,int imaginaryNum){

cout<<"\n----------------Parameterized consturctor is called------------\n";

this->realNum = realNum;

this->imaginaryNum = imaginaryNum;

}

void setRealNumber(int realNum){

this->realNum = realNum;

}

void setImgNumber(int imaginaryNum){

this->imaginaryNum = imaginaryNum;

}

int getRealNumber(){

return realNum;

}

int getImgNumber(){

return imaginaryNum;

}

void display(){

cout<<"\nComplex Number is : "<<realNum<<" + "<<imaginaryNum<<"i"<<endl;

cout<<"------------------------------------------------------\n";

}

};

int main(){

complexNumber clx;

clx.display();

int realNum, imaginaryNum;

cout<<"Enter real number : ";

cin>>realNum;

cout<<"\nEnter Imaginary Value : ";

cin>>imaginaryNum;

// complexNumber clx1(realNum, imaginaryNum);

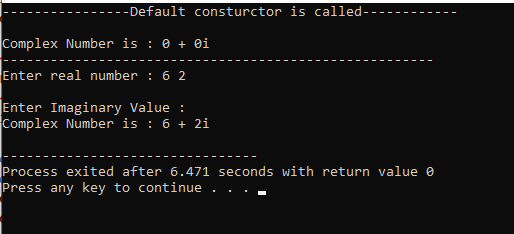
// clx1.display();

clx.setRealNumber(realNum);

clx.setImgNumber(imaginaryNum);

cout<<"\nComplex Number is : "<<clx.getRealNumber()<<" + "<<clx.getImgNumber()<<"i"<<endl; // printing values by getters and setters

return 0;

}****

**Using parameterized constructor**

#include<iostream>

using namespace std;

class complexNumber{

private:

int realNum, imaginaryNum;

public:

complexNumber(){

cout<<"----------------Default consturctor is called------------\n";

realNum = 0;

imaginaryNum = 0;

}

complexNumber(int realNum,int imaginaryNum){

cout<<"\n----------------Parameterized consturctor is called------------\n";

this->realNum = realNum;

this->imaginaryNum = imaginaryNum;

}

void setRealNumber(int realNum){

this->realNum = realNum;

}

void setImgNumber(int imaginaryNum){

this->imaginaryNum = imaginaryNum;

}

int getRealNumber(){

return realNum;

}

int getImgNumber(){

return imaginaryNum;

}

void display(){

cout<<"\nComplex Number is : "<<realNum<<" + "<<imaginaryNum<<"i"<<endl;

cout<<"------------------------------------------------------\n";

}

};

int main(){

complexNumber clx;

clx.display();

int realNum, imaginaryNum;

cout<<"Enter real number : ";

cin>>realNum;

cout<<"\nEnter Imaginary Value : ";

cin>>imaginaryNum;

complexNumber clx1(realNum, imaginaryNum);

clx1.display();

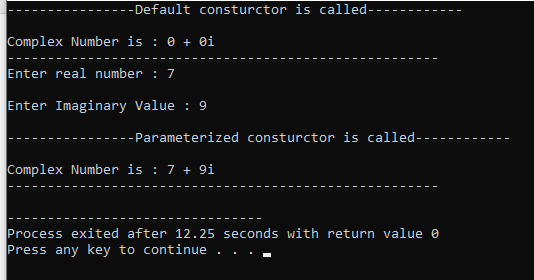
// clx.setRealNumber(realNum);

// clx.setImgNumber(imaginaryNum);

// cout<<"\nComplex Number is : "<<clx.getRealNumber()<<" + "<<clx.getImgNumber()<<"i"<<endl; // printing values by getters and setters

return 0;

}

****