**CPP Lab 3**

Roll no. 248121

PRN: 240841220016

Name: Akshay Chavan

1. Create a class Date with data members as dd, mm, yy.

\* Write a method acceptDate(inti,int j,int k) fro assigning day month year;

\*Also add the display function to print day,month,year

\*Create the

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

**code:**

// 1.

// \* Create a class Date with data members as dd, mm, yy.

// \* Write a method   acceptDate(inti,int j,int k) fro assigning day month year;

// \*Also add the display function to print day,month,year

// \*Create the

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

#include <iostream>

using namespace std;

class Date

{

private:

    int date, month, year;

public:

    void acceptDate(int i, int j, int k)

    {

        date = i;

        month = j;

        year = k;

    }

public:

    void display()

    {

        cout << "Entered date is : " << date << "-" << month << "-" << year << endl;

    }

};

int main()

{

    Date date;

    int d, m, y;

    cout << "Enter Date as DD, MM, YYYY: ";

    cin >> d >> m >> y;

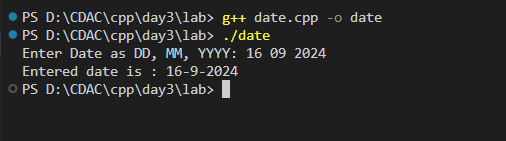
    date.acceptDate(d, m, y);

    date.display();

    return 0;

}

**output:**



1. Create a class Box with data members

\*create function to assign values and retrive value

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

**code:**

**box.cpp**

// 2:Create a class Box with data members

//  \*create function to assign values and retrive value

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

#include <iostream>

using namespace std;

class Box

{

private:

    int length, breadth, height;

    string color;

public:

    void

    acceptData(int i, int j, int k, string c)

    {

        length = i;

        breadth = j;

        height = k;

        color = c;

    }

    void display()

    {

        cout << "Box \nLength:" << length << ", Breadth:" << breadth << ", Height:" << height << endl

             << "Color: " << color << endl;

    }

};

int main()

{

    cout << "----Box----" << endl;

    Box box;

    int l, b, h;

    string col;

    cout << "Enter length, breadth, height" << endl;

    cin >> l >> b >> h;

    cout << "Enter color: " << endl;

    cin >> col;

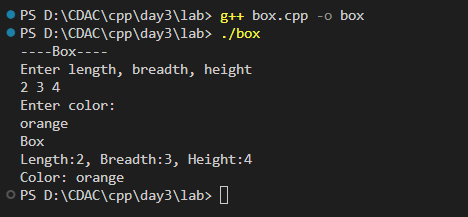
    box.acceptData(l, b, h, col);

    box.display();

    return 0;

}

**output:**

****

1. Create a class ComplexNumber with data members real, imaginary. \*create a methods for assigning values and displaying values

\*write a method for retriving realNumber

ex. int getReal(){ return real;}

write same method for getImaginary

\*create an objct in main method and invoke all methods

**code:**

**complexNum.cpp**

// 3:Create a class ComplexNumber with data members real, imaginary.  \*create a methods for assigning values and displaying values

// \*write a method for retriving realNumber

// ex. int getReal(){ return real;}

//  write same method for getImaginary

// \*create an objct in main method and invoke all methods

#include <iostream>

using namespace std;

class ComplexNum

{

private:

    int real, imaginary;

public:

    void assignVal(int r, int i)

    {

        real = r;

        imaginary = i;

    }

    void display()

    {

        cout << "Complex Num entered is " << real << " + " << imaginary << "i." << endl;

    }

    int getReal()

    {

        return this->real;

    }

    int getimaginary()

    {

        return this->imaginary;

    }

};

int main()

{

    cout << "---Complex number----" << endl;

    ComplexNum num;

    int r, i;

    cout << "Enter real and imaginary part: ";

    cin >> r >> i;

    num.assignVal(r, i);

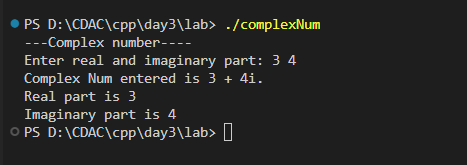
    num.display();

    cout << "Real part is " << num.getReal() << endl;

    cout << "Imaginary part is " << num.getimaginary() << endl;

}

**output:**

****

1. Write all above classes with default and parameter constructor

and test it.

**code:**

**constrs.cpp**

// 4:Write all above classes with default and parameter constructor

// and test it.

// Todo:

#include <iostream>

using namespace std;

// -------------------------------Date Class----------------------------------------------

class Date

{

private:

    int date, month, year;

public:

    Date()

    {

        cout << "-----------Default constr of Date called-----------" << endl;

        date = 1;

        month = 6;

        year = 2001;

    }

    Date(int d, int m, int y)

    {

        cout << "-----------Parameterized constr of Date called-----------" << endl;

        date = d;

        month = m;

        year = y;

    }

    void display()

    {

        cout << "Date: " << date << "-" << month << "-" << year << endl;

    }

};

// -------------------------------Box Class----------------------------------------------

class Box

{

private:

    int length, breadth, height;

    string color;

    // You must have default constructor (written yourself) if you want to have parametrized constr

public:

    Box()

    {

        cout << "-----------Default constr of Box called-----------" << endl;

        length = 2;

        breadth = 3;

        height = 4;

        color = "Yellow";

    }

    Box(int i, int j, int k, string c)

    {

        cout << "-----------Parameterized constr of Box called-----------" << endl;

        length = i;

        breadth = j;

        height = k;

        color = c;

    }

    void display()

    {

        cout << "--Box Information--" << endl;

        cout << "Length:" << length << ", Breadth:" << breadth << ", Height:" << height << endl

             << "Color: " << color << endl;

    }

};

// -------------------------------ComplexNum Class----------------------------------------------

class ComplexNum

{

private:

    int real, imaginary;

public:

    ComplexNum()

    {

        cout << "-----------Default constr of ComplexNum called-----------" << endl;

        real = 0;

        imaginary = 0;

    }

    ComplexNum(int r, int i)

    {

        cout << "-----------Parameterized constr called-----------" << endl;

        real = r;

        imaginary = i;

    }

    void display()

    {

        cout << "Your complex num: " << real << " + " << imaginary << "i." << endl;

    }

};

int main()

{

    int ch;

    cout << "---Default and parameterized constructors" << endl;

    cout << "1.Date  2.Box  3.ComplexNum" << endl;

    do

    {

        cout << "Enter choice: ";

        cin >> ch;

        switch (ch)

        {

        case 1:

        {

            Date date1;

            int d, m, y;

            cout << "Default date is: ";

            date1.display();

            cout << "Enter date as dd mm yyyy: ";

            cin >> d >> m >> y;

            Date date2(d, m, y);

            date2.display();

            break;

        }

        case 2:

        {

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

            Box box;

            int l, b, h;

            string col;

            cout << "Enter Length, breadth, height: ";

            cin >> l >> b >> h;

            cout << "Enter color: ";

            cin >> col;

            Box box2(l, b, h, col);

            box2.display();

            break;

        }

        case 3:

        {

            ComplexNum num1;

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

            num1.display();

            int r, i;

            cout << "Enter real and imaginary parts: ";

            cin >> r >> i;

            ComplexNum num2(r, i);

            num2.display();

            break;

        }

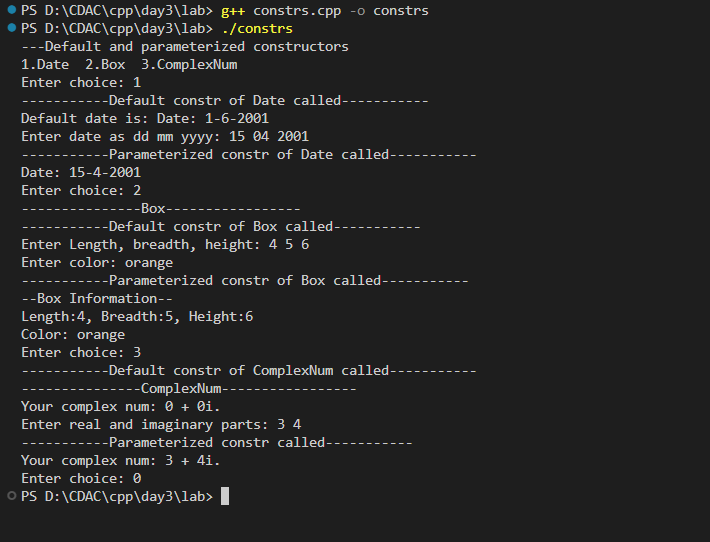
        }

    } while (ch != 0);

    return 0;

}

**output:**

****