**Chapter 12: Classes and Data Abstraction**

**Laboratory Exercises (6)**

**Intro to Object Oriented Programming**

**KEYWORDS: class, object**

**Program 1:** Define a class named time having hour, minute and second as data members. The class should have two member functions named gettime and displaytime. The gettime function is used for setting values of hour, minute and second. The displaytime function is used for displaying hour. minute and time in appropriate format. Use the class in your main program and set the time as 12:35:55.

#include <iostream>

using namespace std;

class Time

{

private:

int hour;

int minute;

int second;

public:

void gettime(int h, int m, int s)

{

hour = h;

minute = m;

second = s;

}

void displaytime()

{

cout << hour << ":" << minute << ":" << second <<endl;

}

};

int main()

{

Time t;

t.gettime(12, 35, 55);

t.displaytime();

return 0;

}

===================== another way ======================

#include <iostream>

using namespace std;

class Time

{

private:

int hour;

int minute;

int second;

public:

void gettime(int h, int m, int s);

void displaytime();

};

void Time::gettime(int h, int m, int s)

{

hour = h;

minute = m;

second = s;

}

void Time::displaytime()

{

cout << hour << ":" << minute << ":" << second <<endl;

}

int main()

{

Time t;

t.gettime(12, 35, 55);

t.displaytime();

return 0;

}

**======================= How to make the displaytime function const. =============**

#include <iostream>

using namespace std;

class Time

{

private:

int hour;

int minute;

int second;

public:

void gettime(int h, int m, int s);

void displaytime() const;

};

void Time::gettime(int h, int m, int s)

{

hour = h;

minute = m;

second = s;

}

void Time::displaytime() const

{

hour =3;

cout << hour << ":" << minute << ":" << second <<endl;

}

int main()

{

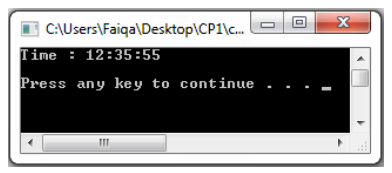
Time t;

t.gettime(12, 35, 55);

t.displaytime();

return 0;

}

****

**Sample Output**

**Program 2:** Define a class named rectangle having height and width as data members. The class has three member functions named getdata, displaydata and area. The getdata function is used for setting the values of width and height. The displaydata function is used for displaying width and height. Finally, the area function is used for calculating the area of rectangle and returning the calculated value. Use the class in your main program and calculate the area for (2, 4) and (6, 9).

#include <iostream>

using namespace std;

class Rectangle

{

private:

int height;

int width;

public:

void getDate(int h,int w)

{

height = h;

width = w;

}

void displayData()

{

cout <<"Height = " << height << endl;

cout <<"Width = " << width <<endl;

}

int area()

{

return height \* width;

}

};

int main()

{

Rectangle r1;

Rectangle r2;

r1.getDate(2, 4);

r1.displayData();

cout << "Area = " <<r1.area() << endl << endl;

r2.getDate(6, 9);

r2.displayData();

cout << "Area = " <<r2.area() << endl;

return 0;

}

============================= another way =============================

#include <iostream>

using namespace std;

class Rectangle

{

private:

int height;

int width;

public:

void getDate(int, int);

void displayData();

int area();

};

void Rectangle::getDate(int h,int w)

{

height = h;

width = w;

}

void Rectangle::displayData()

{

cout <<"Height = " << height << endl;

cout <<"Width = " << width <<endl;

}

int Rectangle::area()

{

return height \* width;

}

int main()

{

Rectangle r1;

Rectangle r2;

r1.getDate(2, 4);

r1.displayData();

cout << "Area = " <<r1.area() << endl << endl;

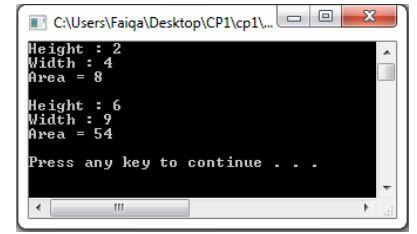
r2.getDate(6, 9);

r2.displayData();

cout << "Area = " <<r2.area() << endl;

return 0;

}



**Sample Output**

**C++ Constructors**

**C++ program to calculate the area of a Rectangle.**

**// C++ program to understand Default and Parameterized Constructor**

#include <iostream>

using namespace std;

// declare a class

class Rect

{

private:

double length;

double height;

public:

// create a default constructor

Rect()

{

// initialize private variables

length = 2;

height = 2;

}

// create parameterized constructor

Rect(double L, double H)

{

// initialize private variables

length = L;

height = H;

}

double calculateArea() {

return length \* height;

}

};

int main() {

// create object and initialize data members

Rect rec1;

Rect rec2(6.0, 8.4);

cout << "The Area of Rectangle 1: " << rec1.calculateArea() << endl;

cout << "The Area of Rectangle 2: " << rec2.calculateArea() << endl;

return 0;

}

===============================================

**C++ program to calculate the area of a Rectangle.**

**// C++ program to understand Constructors with Default Parameters**

#include <iostream>

using namespace std;

// declare a class

class Rect

{

private:

double length;

double height;

public:

// create parameterized constructor

Rect(double L = 2, double H = 2)

{

// initialize private variables

length = L;

height = H;

}

double calculateArea() {

return length \* height;

}

};

int main() {

// create object and initialize data members

Rect rec1;

Rect rec2(10.0);

Rect rec3(6.0, 8.0);

cout << "The Area of Rectangle 1: " << rec1.calculateArea() << endl;

cout << "The Area of Rectangle 2: " << rec2.calculateArea() << endl;

cout << "The Area of Rectangle 2: " << rec3.calculateArea() << endl;

return 0;

}