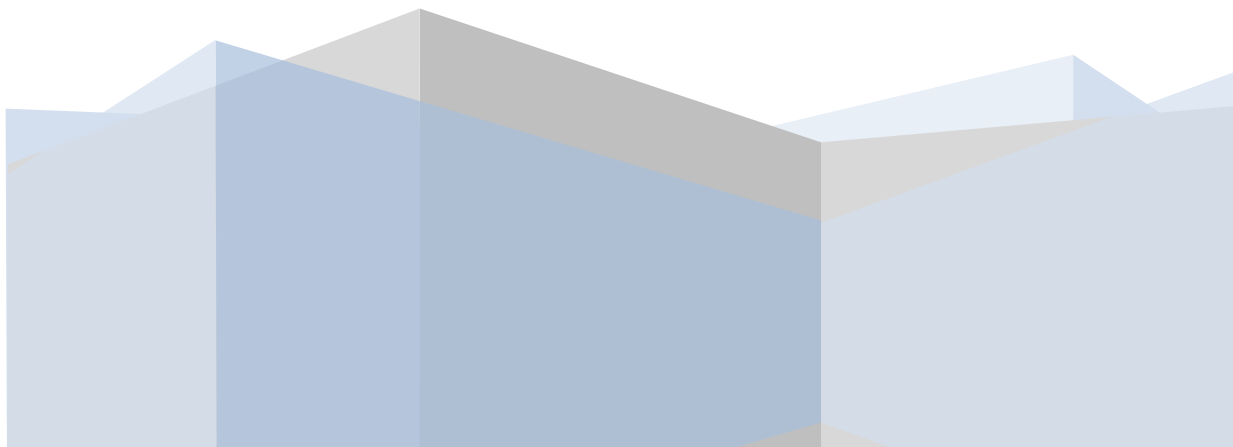


Mehran University of Engineering and Technology Jamshoro



ASSIGNMENT	:LAB 010
SUBJECT	:PROGRAMMING FUNDAMENTALS
ROLL NO	:24BSAI29
SUBMITTED BY	: SYED MUHAMMAD QASIM
SUBMITTED TO	:MA'AM FAHAMA BARKZAI



Department of Software Engineering
Mehran University of Engineering and Technology, Jamshoro

Course: AI-112 – Programming Fundamentals

Instructor	Engr. Fahama Barakzai	Practical/Lab No.	10 – 12
Date		CLOs	3
Signature		Assessment Score	03 Marks

Topic **Object Oriented Programming in C++**

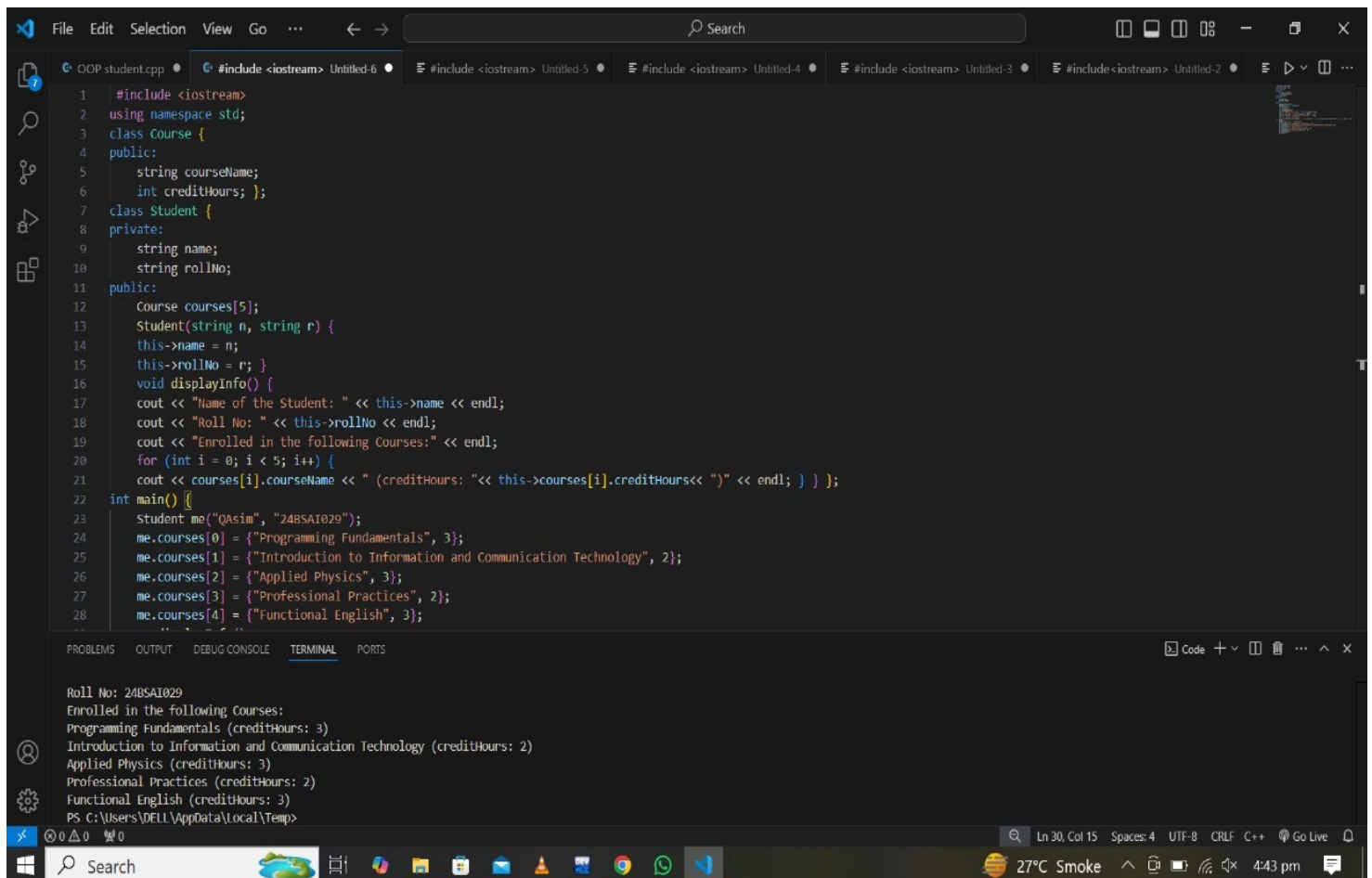
Objectives	<ul style="list-style-type: none">- To become familiar with OOP in C++.- To work with concepts like Classes, Constructors and Destructors.- To become familiar with Inheritance and Polymorphism in C++.
-------------------	--

Lab Discussion: Theoretical concepts and Procedural steps

TOOLS: TURBOO C++/ DEV C++/ VS-CODE/ CODE BLOCKS

Lab Tasks

1. To develop the Student Registration System by object-oriented scenarios. It should consist of two classes, namely Student and Course.



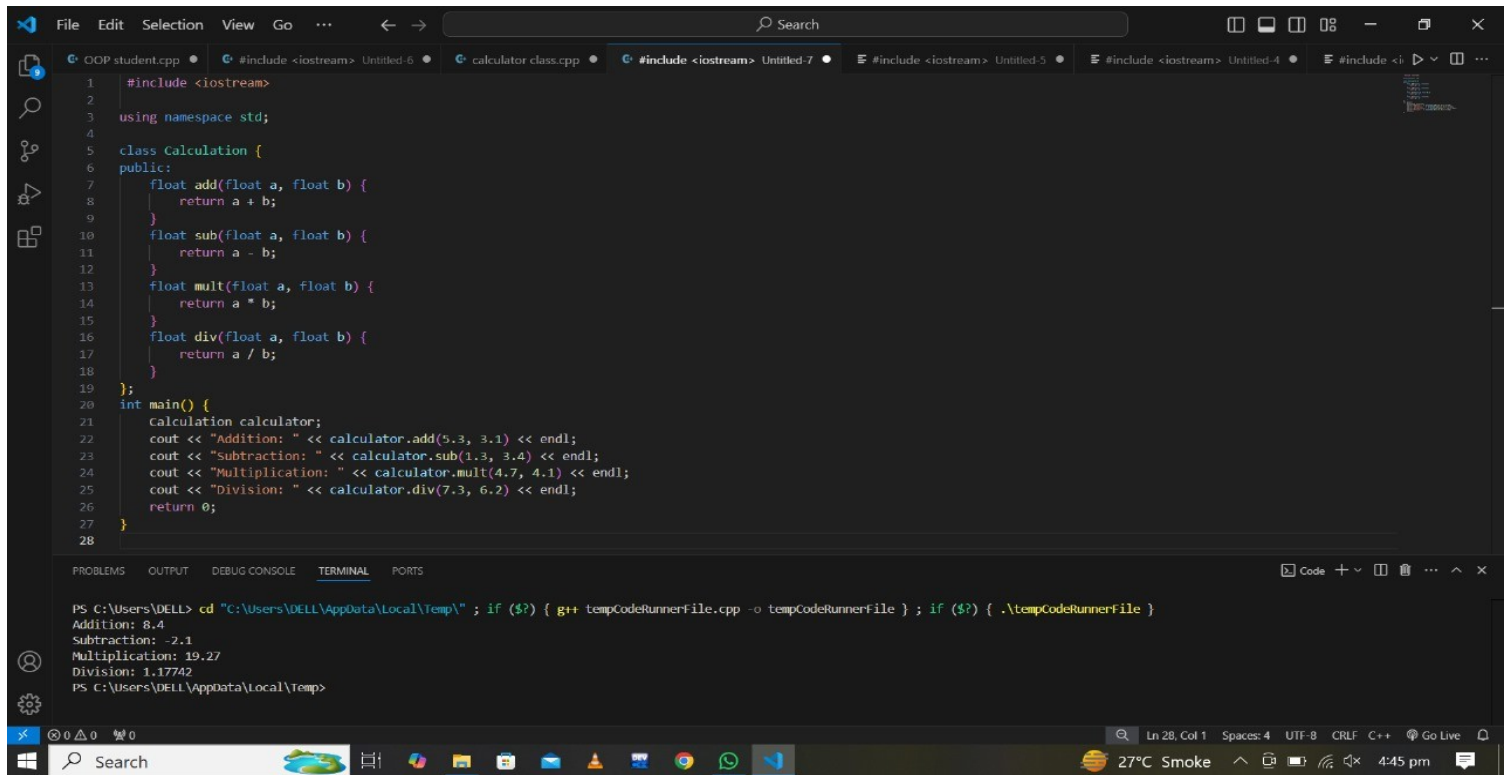
The screenshot shows a C++ IDE with a dark theme. The main editor window displays a C++ program for a Student Registration System. The code defines two classes: `Course` and `Student`. The `Course` class has a `courseName` (string) and `creditHours` (int). The `Student` class has a `name` (string), `rollNo` (string), and an array of `Course` objects named `courses`. The `Student` class includes a constructor, a `displayInfo` method, and a `main` function. The `main` function creates a `Student` object named `me` and initializes the `courses` array with five courses: "Programming Fundamentals" (3), "Introduction to Information and Communication Technology" (2), "Applied Physics" (3), "Professional Practices" (2), and "Functional English" (3). The `displayInfo` method is called, which prints the student's name, roll number, and a list of enrolled courses with their credit hours.

```
1 #include <iostream>
2 using namespace std;
3 class Course {
4 public:
5     string courseName;
6     int creditHours; };
7 class Student {
8 private:
9     string name;
10    string rollNo;
11 public:
12    Course courses[5];
13    Student(string n, string r) {
14        this->name = n;
15        this->rollNo = r; }
16    void displayInfo() {
17        cout << "Name of the Student: " << this->name << endl;
18        cout << "Roll No: " << this->rollNo << endl;
19        cout << "Enrolled in the following Courses:" << endl;
20        for (int i = 0; i < 5; i++) {
21            cout << courses[i].courseName << " (creditHours: "<< this->courses[i].creditHours<< ") " << endl; } } };
22 int main() {
23     Student me("Qasim", "24BSAI029");
24     me.courses[0] = {"Programming Fundamentals", 3};
25     me.courses[1] = {"Introduction to Information and Communication Technology", 2};
26     me.courses[2] = {"Applied Physics", 3};
27     me.courses[3] = {"Professional Practices", 2};
28     me.courses[4] = {"Functional English", 3};
29 }
```

The terminal window at the bottom shows the output of the program:

```
Roll No: 24BSAI029
Enrolled in the following Courses:
Programming Fundamentals (creditHours: 3)
Introduction to Information and Communication Technology (creditHours: 2)
Applied Physics (creditHours: 3)
Professional Practices (creditHours: 2)
Functional English (creditHours: 3)
PS C:\Users\DELL\AppData\Local\Temp>
```

2). Try to create a class having four functions, add, sub, mult and div. Each function accepts 2 parameters and returns the sum, difference, multiplication and division of these numbers. Create a main class having a main function that uses the above class.

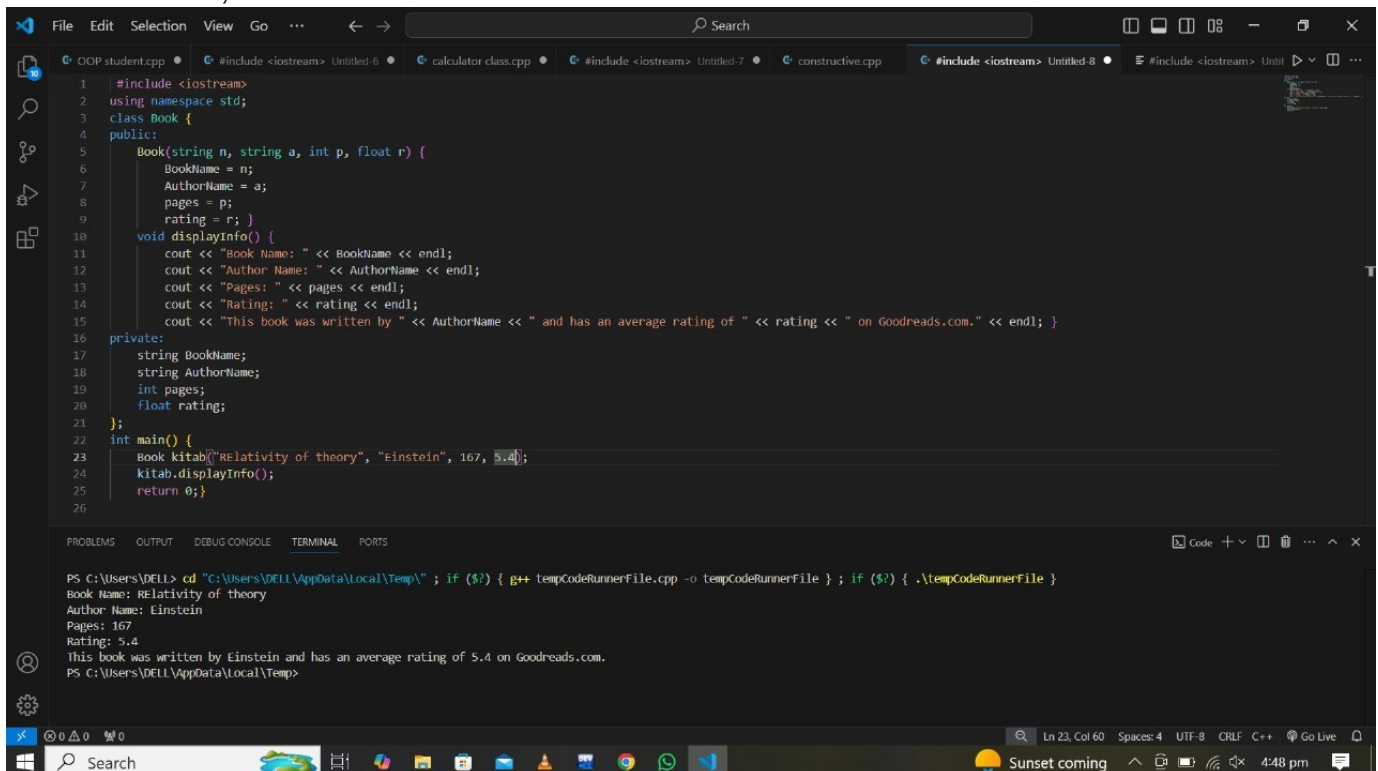


```
1 #include <iostream>
2
3 using namespace std;
4
5 class Calculation {
6 public:
7     float add(float a, float b) {
8         return a + b;
9     }
10    float sub(float a, float b) {
11        return a - b;
12    }
13    float mult(float a, float b) {
14        return a * b;
15    }
16    float div(float a, float b) {
17        return a / b;
18    }
19 };
20
21 int main() {
22     Calculation calculator;
23     cout << "Addition: " << calculator.add(5.3, 3.1) << endl;
24     cout << "Subtraction: " << calculator.sub(1.3, 3.4) << endl;
25     cout << "Multiplication: " << calculator.mult(4.7, 4.1) << endl;
26     cout << "Division: " << calculator.div(7.3, 6.2) << endl;
27     return 0;
28 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\DELL> cd "C:\Users\DELL\AppData\Local\Temp\"; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunnerFile }; if ($?) { .\tempCodeRunnerFile }
Addition: 8.4
Subtraction: -2.1
Multiplication: 19.27
Division: 1.17742
PS C:\Users\DELL\AppData\Local\Temp>
```

3) Demonstrate the use of the following:
Constructor, Public and Private class members.

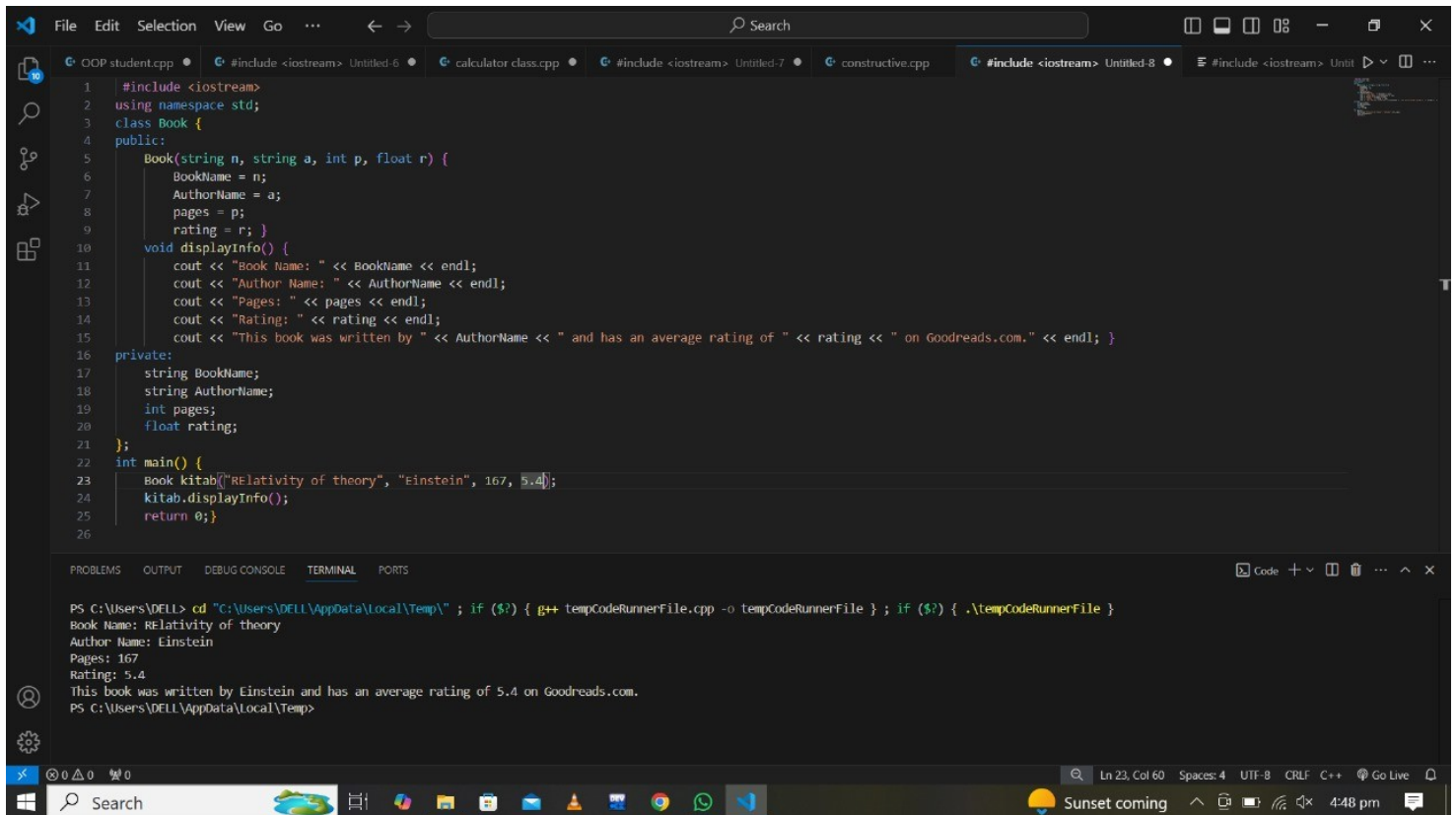


```
1 #include <iostream>
2 using namespace std;
3 class Book {
4 public:
5     Book(string n, string a, int p, float r) {
6         BookName = n;
7         AuthorName = a;
8         pages = p;
9         rating = r;
10    }
11    void displayInfo() {
12        cout << "Book Name: " << BookName << endl;
13        cout << "Author Name: " << AuthorName << endl;
14        cout << "Pages: " << pages << endl;
15        cout << "Rating: " << rating << endl;
16        cout << "This book was written by " << AuthorName << " and has an average rating of " << rating << " on Goodreads.com." << endl;
17    }
18 private:
19     string BookName;
20     string AuthorName;
21     int pages;
22     float rating;
23 };
24
25 int main() {
26     Book kitab("Relativity of theory", "Einstein", 167, 5.4);
27     kitab.displayInfo();
28     return 0;
29 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\DELL> cd "C:\Users\DELL\AppData\Local\Temp\"; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunnerFile }; if ($?) { .\tempCodeRunnerFile }
Book Name: Relativity of theory
Author Name: Einstein
Pages: 167
Rating: 5.4
This book was written by Einstein and has an average rating of 5.4 on Goodreads.com.
PS C:\Users\DELL\AppData\Local\Temp>
```

- 4) Create a class Circle with a single member variable Radius and 2 member functions: a default constructor, which sets the radius to 0, and a parameterized constructor, which sets the radius to a given value. Define a member function Area which calculates and returns the area of the circle.



The screenshot shows a C++ IDE with a file named 'OOP student.cpp'. The code defines a class 'Book' with a constructor 'Book(string n, string a, int p, float r)' that initializes 'BookName', 'AuthorName', 'pages', and 'rating'. It also has a 'displayInfo()' method that prints these details. The 'main()' function creates a 'Book' object 'kitab' with the values 'Relativity of theory', 'Einstein', 167, and 5.4, and calls 'displayInfo()'. The terminal output shows the program's execution, printing the book's details.

```
1 #include <iostream>
2 using namespace std;
3 class Book {
4 public:
5     Book(string n, string a, int p, float r) {
6         BookName = n;
7         AuthorName = a;
8         pages = p;
9         rating = r; }
10    void displayInfo() {
11        cout << "Book Name: " << BookName << endl;
12        cout << "Author Name: " << AuthorName << endl;
13        cout << "Pages: " << pages << endl;
14        cout << "Rating: " << rating << endl;
15        cout << "This book was written by " << AuthorName << " and has an average rating of " << rating << " on Goodreads.com." << endl; }
16 private:
17     string BookName;
18     string AuthorName;
19     int pages;
20     float rating;
21 };
22 int main() {
23     Book kitab("Relativity of theory", "Einstein", 167, 5.4);
24     kitab.displayInfo();
25     return 0; }
26
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\DELL> cd "C:\Users\DELL\AppData\Local\Temp\" ; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Book Name: Relativity of theory
Author Name: Einstein
Pages: 167
Rating: 5.4
This book was written by Einstein and has an average rating of 5.4 on Goodreads.com.
PS C:\Users\DELL\AppData\Local\Temp>
```

Ln 23, Col 60 Spaces: 4 UTF-8 CRLF C++ Go Live 4:48 pm

- 5) Create a class Rectangle that has two private data members length and width, and one public member function area() that calculates the area of rectangle.

```
1 #include <iostream>
2 using namespace std;
3
4 class Rectangle {
5 public:
6     Rectangle(float l, float w) {
7         length = l;
8         width = w;
9     }
10    float area() {
11        return length * width;
12    }
13 private:
14    float length;
15    float width;
16 };
17
18 int main() {
19     Rectangle rect = Rectangle(27.5, 21.8);
20     cout << "Area of our Rectangle is: " << rect.area() << endl;
21     return 0;
22 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\DELL> cd "C:\Users\DELL\AppData\Local\Temp\" ; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Area of our Rectangle is: 599.5
PS C:\Users\DELL\AppData\Local\Temp>
```

Ln 22, Col 1 Spaces: 4 UTF-8 CRLF C++ Go Live

27°C Smoke 4:51 pm