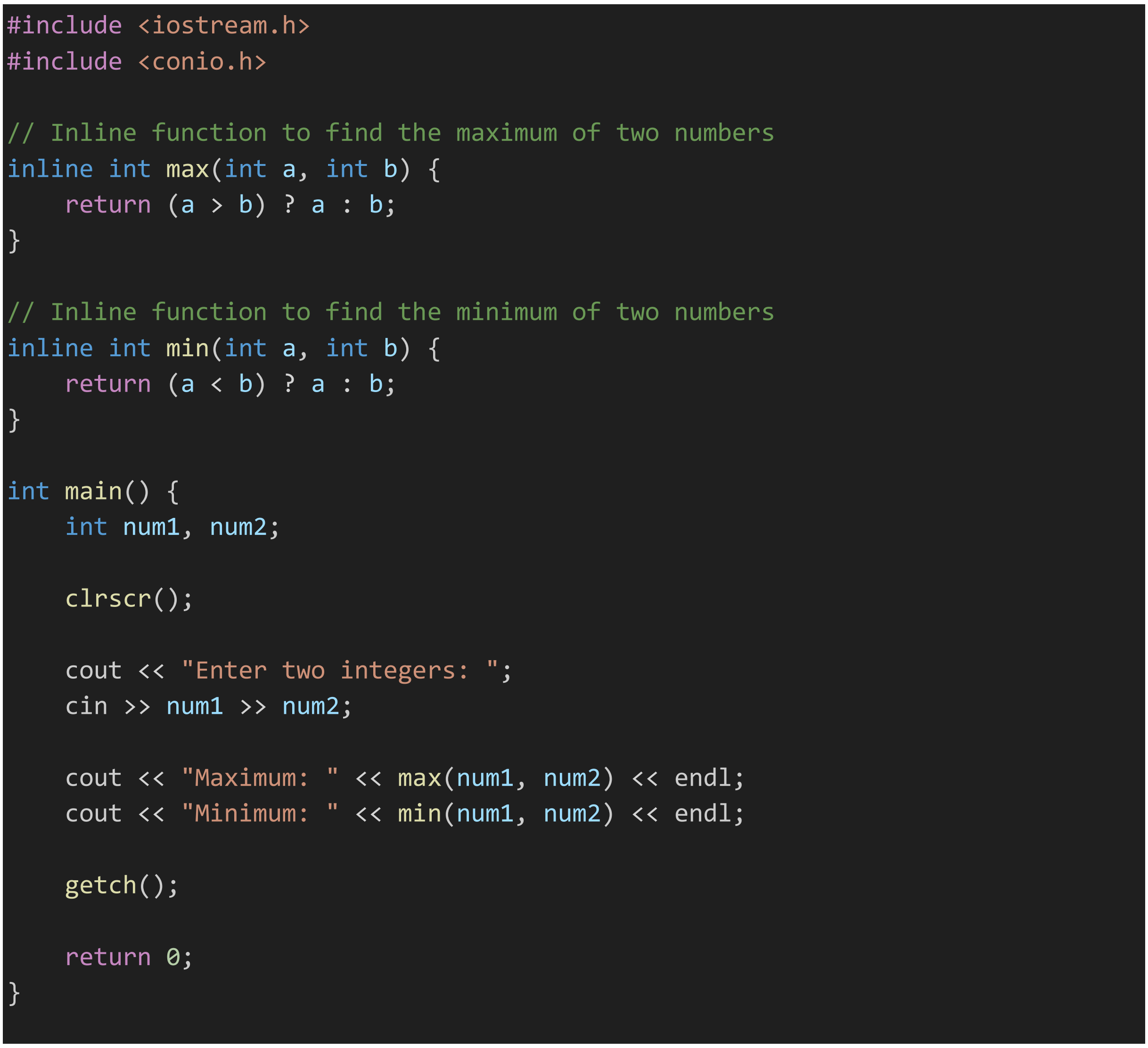
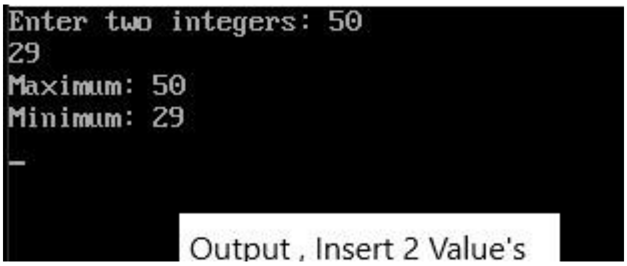
Slip -1

Q. 1)

a) Write a C++ program to check maximum and minimum of two integer numbers. (Use Inline Function and Conditional Operator) [Marks 15]



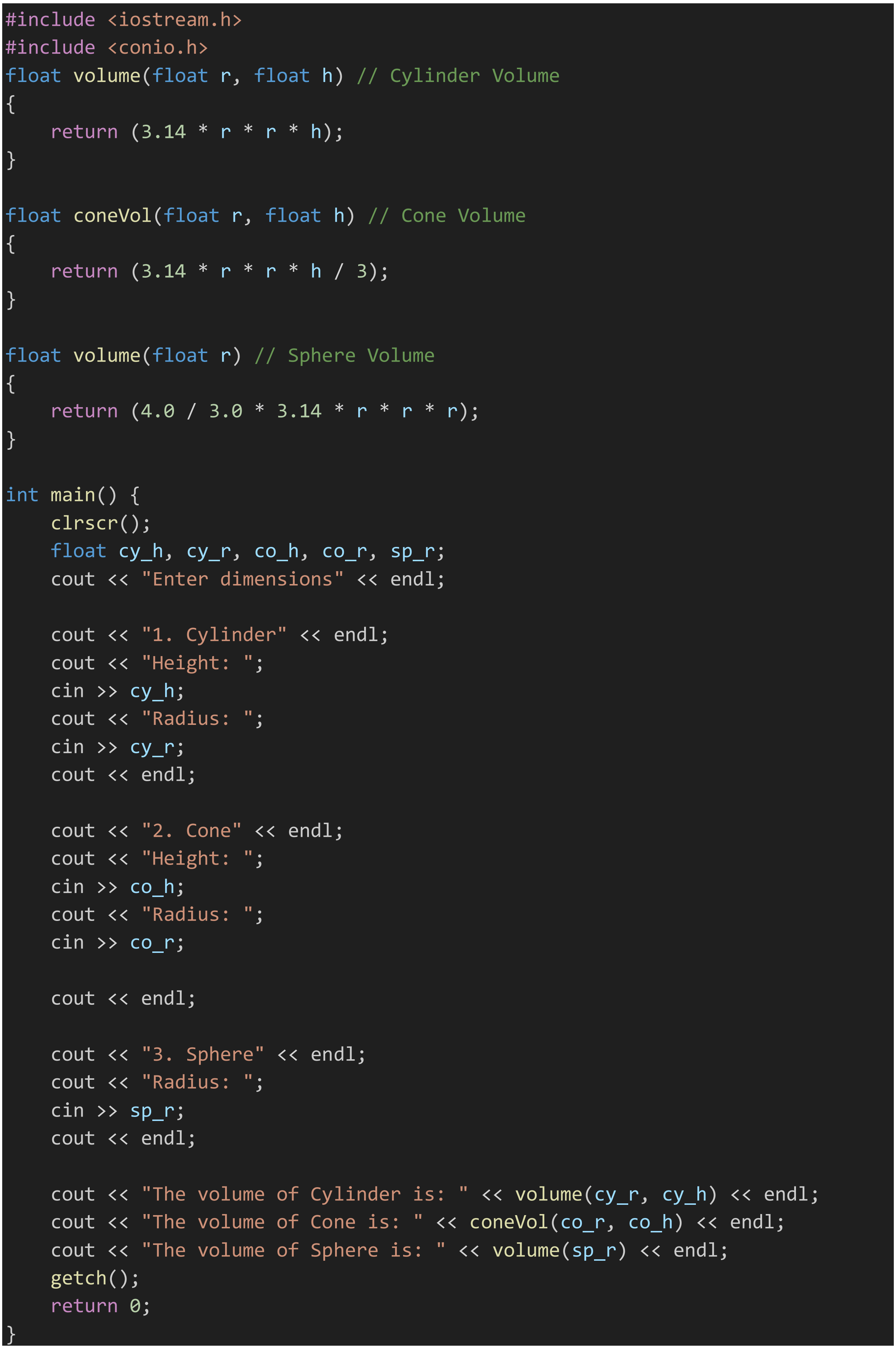
```
#include <iostream.h>
#include <conio.h>

// Inline function to find the maximum of two numbers
inline int max(int a, int b) {
    return (a > b) ? a : b;
}

// Inline function to find the minimum of two numbers
inline int min(int a, int b) {
    return (a < b) ? a : b;
}

int main() {
    int num1, num2;

    clrscr();

    cout << "Enter two integers: ";
    cin >> num1 >> num2;

    cout << "Maximum: " << max(num1, num2) << endl;
    cout << "Minimum: " << min(num1, num2) << endl;

    getch();

    return 0;
}
```



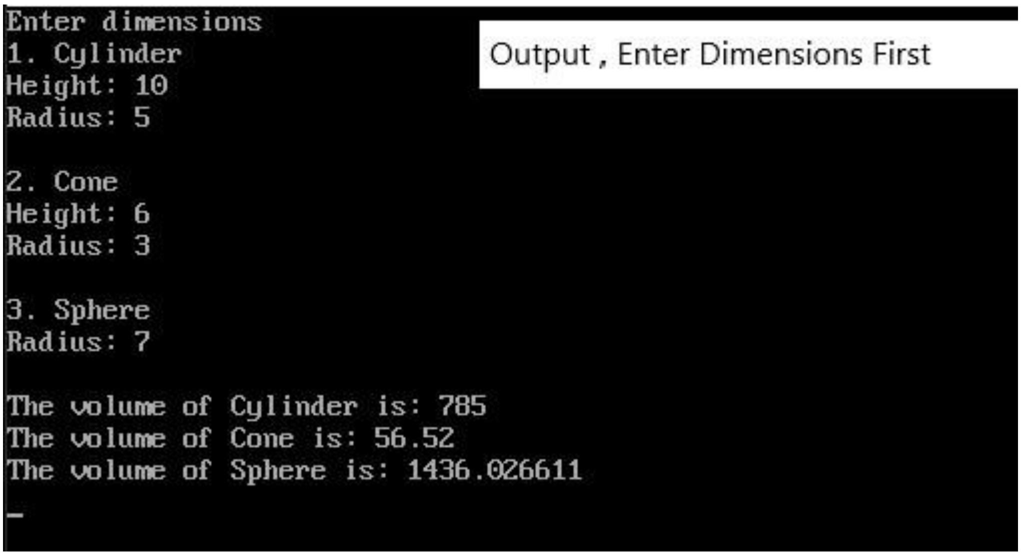
```
Enter two integers: 50
29
Maximum: 50
Minimum: 29
_
```

Output , Insert 2 Value's

b) Write a C++ Program to find volume of cylinder, cone and sphere. (Use function overloading) [Marks 25]



```cpp
#include <iostream.h>
#include <conio.h>
float volume(float r, float h) // Cylinder Volume
{
    return (3.14 * r * r * h);
}

float coneVol(float r, float h) // Cone Volume
{
    return (3.14 * r * r * h / 3);
}

float volume(float r) // Sphere Volume
{
    return (4.0 / 3.0 * 3.14 * r * r * r);
}

int main() {
    clrscr();
    float cy_h, cy_r, co_h, co_r, sp_r;
    cout << "Enter dimensions" << endl;

    cout << "1. Cylinder" << endl;
    cout << "Height: ";
    cin >> cy_h;
    cout << "Radius: ";
    cin >> cy_r;
    cout << endl;

    cout << "2. Cone" << endl;
    cout << "Height: ";
    cin >> co_h;
    cout << "Radius: ";
    cin >> co_r;

    cout << endl;

    cout << "3. Sphere" << endl;
    cout << "Radius: ";
    cin >> sp_r;
    cout << endl;

    cout << "The volume of Cylinder is: " << volume(cy_r, cy_h) << endl;
    cout << "The volume of Cone is: " << coneVol(co_r, co_h) << endl;
    cout << "The volume of Sphere is: " << volume(sp_r) << endl;
    getch();
    return 0;
}
```



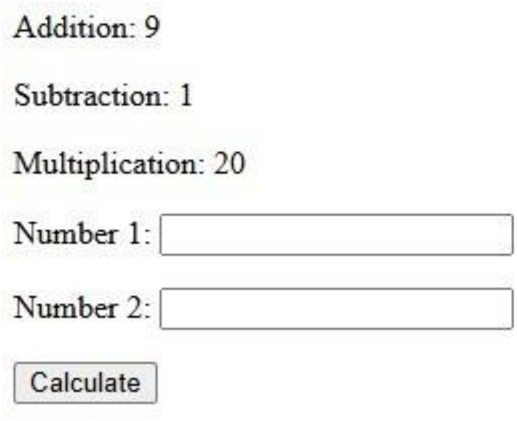
Output , Enter Dimensions First

```
Enter dimensions
1. Cylinder
Height: 10
Radius: 5

2. Cone
Height: 6
Radius: 3

3. Sphere
Radius: 7

The volume of Cylinder is: 785
The volume of Cone is: 56.52
The volume of Sphere is: 1436.026611
_
```

Q. 2)

a) Write a PHP script to create a simple calculator that can accept two numbers and perform operations like add, sub, multiply (Use the concept of Class) [Marks 15]

```php
<?php
class Calculator {
    function add($a, $b) { return $a + $b; }
    function subtract($a, $b) { return $a - $b; }
    function multiply($a, $b) { return $a * $b; }
}

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $num1 = $_POST['num1'];
    $num2 = $_POST['num2'];
    $calc = new Calculator();
    echo "<p>Addition: {$calc->add($num1, $num2)}</p>";
    echo "<p>Subtraction: {$calc->subtract($num1, $num2)}</p>";
    echo "<p>Multiplication: {$calc->multiply($num1, $num2)}</p>";
}
?>

<form method="post">
    <p>Number 1: <input type="number" name="num1" required></p>
    <p>Number 2: <input type="number" name="num2" required></p>
    <p><input type="submit" value="Calculate"></p>
</form>
```



Addition: 9

Subtraction: 1

Multiplication: 20

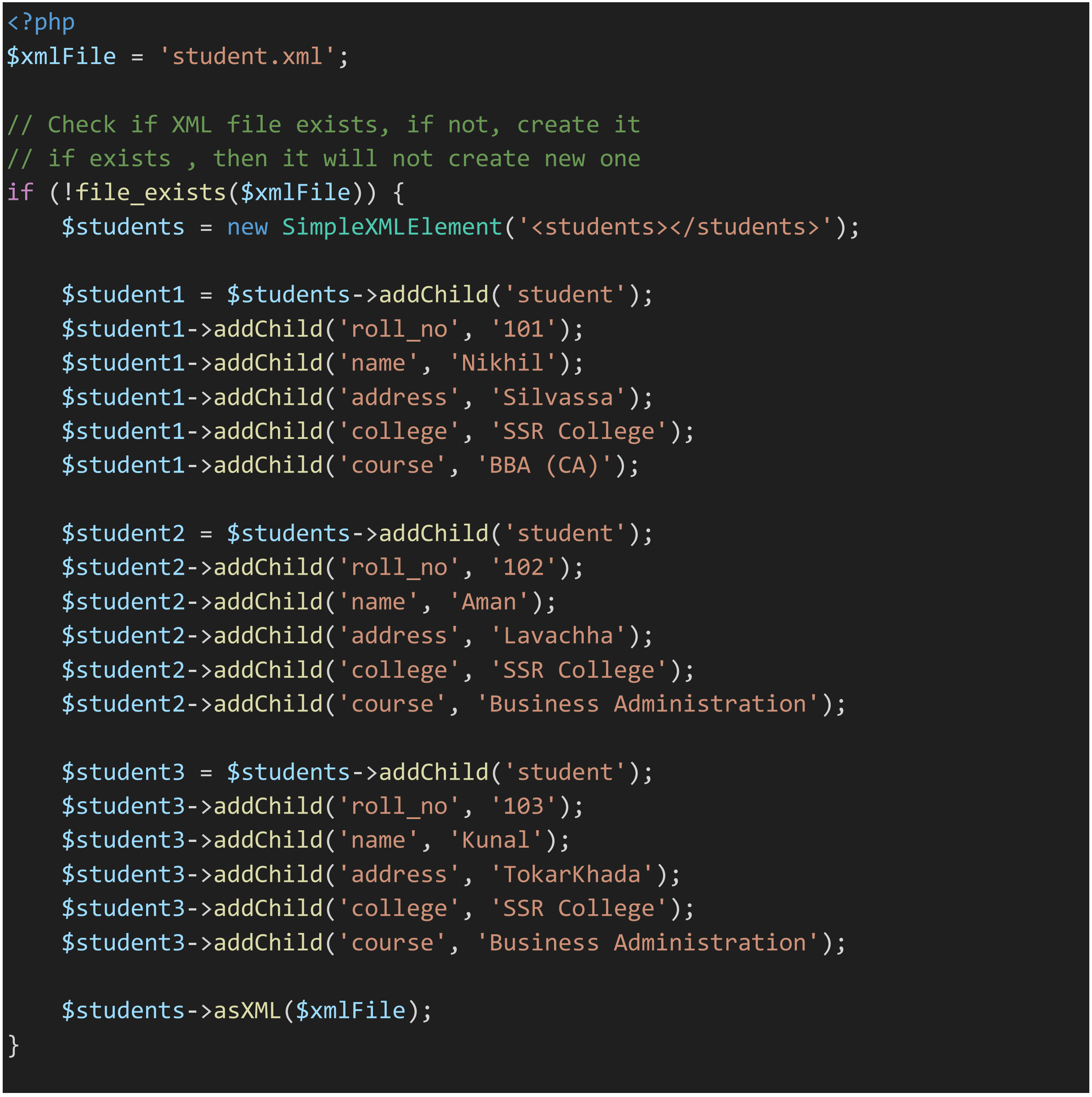
Number 1:

Number 2:

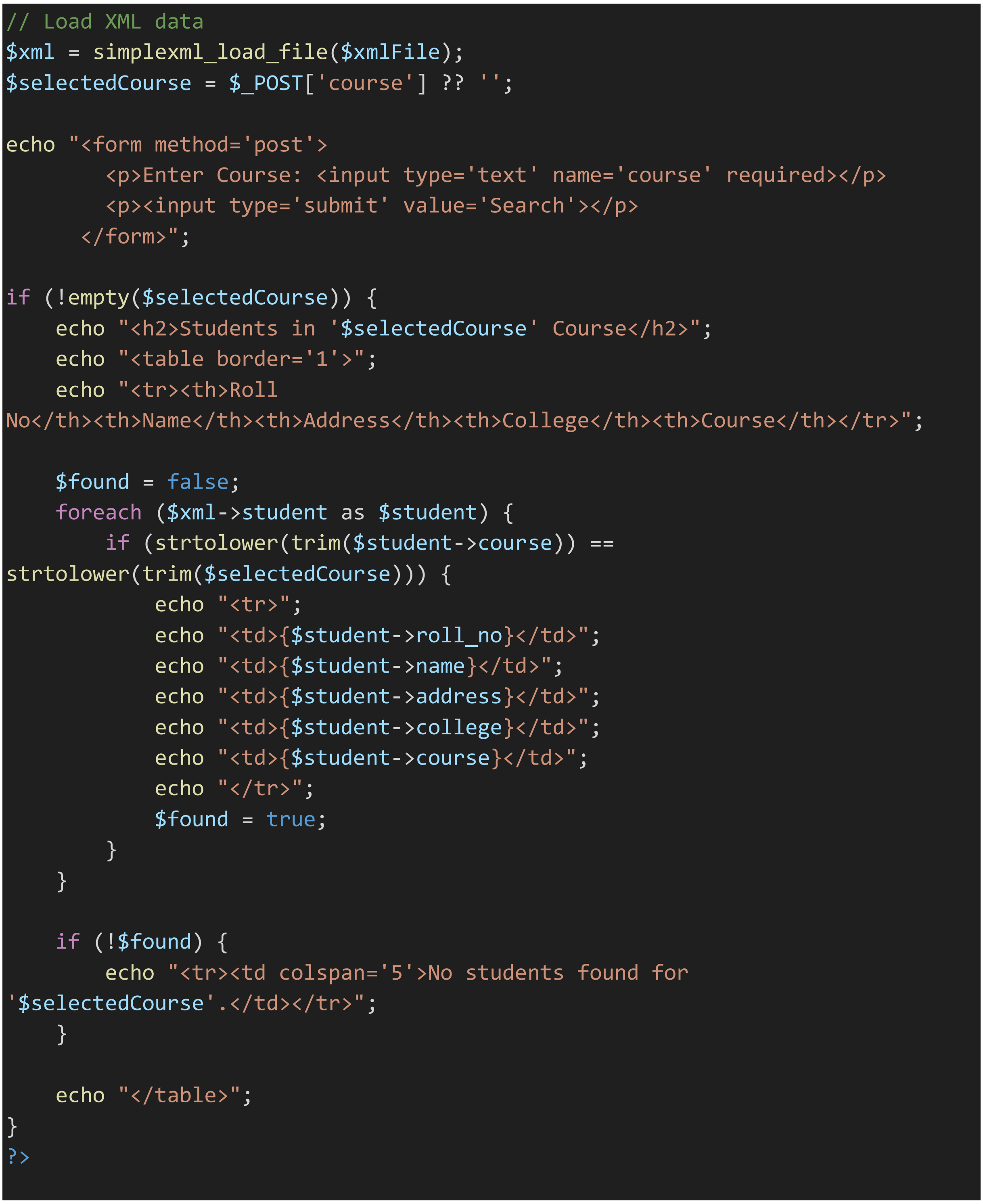
Output ,
Insert Number 1 & Number 2

b) Write a PHP Script to create student.xml file which contains student roll no, name, address, college course. Print students details of specific course in tabular format after accepting course as input.

[Marks 25]

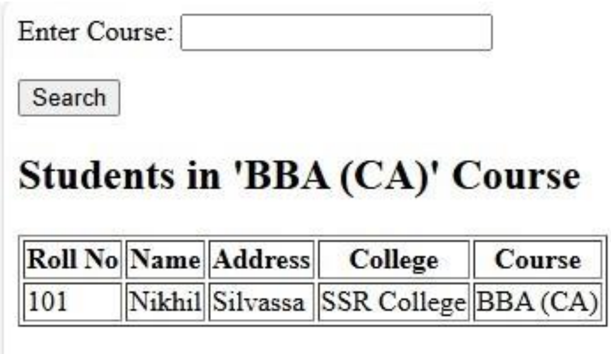


```php
<?php
$xmlFile = 'student.xml';

// Check if XML file exists, if not, create it
// if exists , then it will not create new one
if (!file_exists($xmlFile)) {
    $students = new SimpleXMLElement('<students></students>');

    $student1 = $students->addChild('student');
    $student1->addChild('roll_no', '101');
    $student1->addChild('name', 'Nikhil');
    $student1->addChild('address', 'Silvassa');
    $student1->addChild('college', 'SSR College');
    $student1->addChild('course', 'BBA (CA)');

    $student2 = $students->addChild('student');
    $student2->addChild('roll_no', '102');
    $student2->addChild('name', 'Aman');
    $student2->addChild('address', 'Lavachha');
    $student2->addChild('college', 'SSR College');
    $student2->addChild('course', 'Business Administration');

    $student3 = $students->addChild('student');
    $student3->addChild('roll_no', '103');
    $student3->addChild('name', 'Kunal');
    $student3->addChild('address', 'TokarKhada');
    $student3->addChild('college', 'SSR College');
    $student3->addChild('course', 'Business Administration');

    $students->asXML($xmlFile);
}
```



```php
// Load XML data
$xml = simplexml_load_file($xmlFile);
$selectedCourse = $_POST['course'] ?? '';

echo "<form method='post'>
        <p>Enter Course: <input type='text' name='course' required></p>
        <p><input type='submit' value='Search'></p>
      </form>";

if (!empty($selectedCourse)) {
    echo "<h2>Students in '$selectedCourse' Course</h2>";
    echo "<table border='1'>";
    echo "<tr><th>Roll
No</th><th>Name</th><th>Address</th><th>College</th><th>Course</th></tr>";

    $found = false;
    foreach ($xml->student as $student) {
        if (strtolower(trim($student->course)) ==
strtolower(trim($selectedCourse))) {
            echo "<tr>";
            echo "<td>{$student->roll_no}</td>";
            echo "<td>{$student->name}</td>";
            echo "<td>{$student->address}</td>";
            echo "<td>{$student->college}</td>";
            echo "<td>{$student->course}</td>";
            echo "</tr>";
            $found = true;
        }
    }

    if (!$found) {
        echo "<tr><td colspan='5'>No students found for
'$selectedCourse'.</td></tr>";
    }

    echo "</table>";
}
?>
```

<< As Output The Student.xml File will be created and following screenshots are output in Browser … For Changing student details , make changes in PHP -> XML section >>



Enter Course:

Students in 'BBA (CA)' Course

Roll No	Name	Address	College	Course
101	Nikhil	Silvassa	SSR College	BBA (CA)

Output , Refer PHP code For Course Name

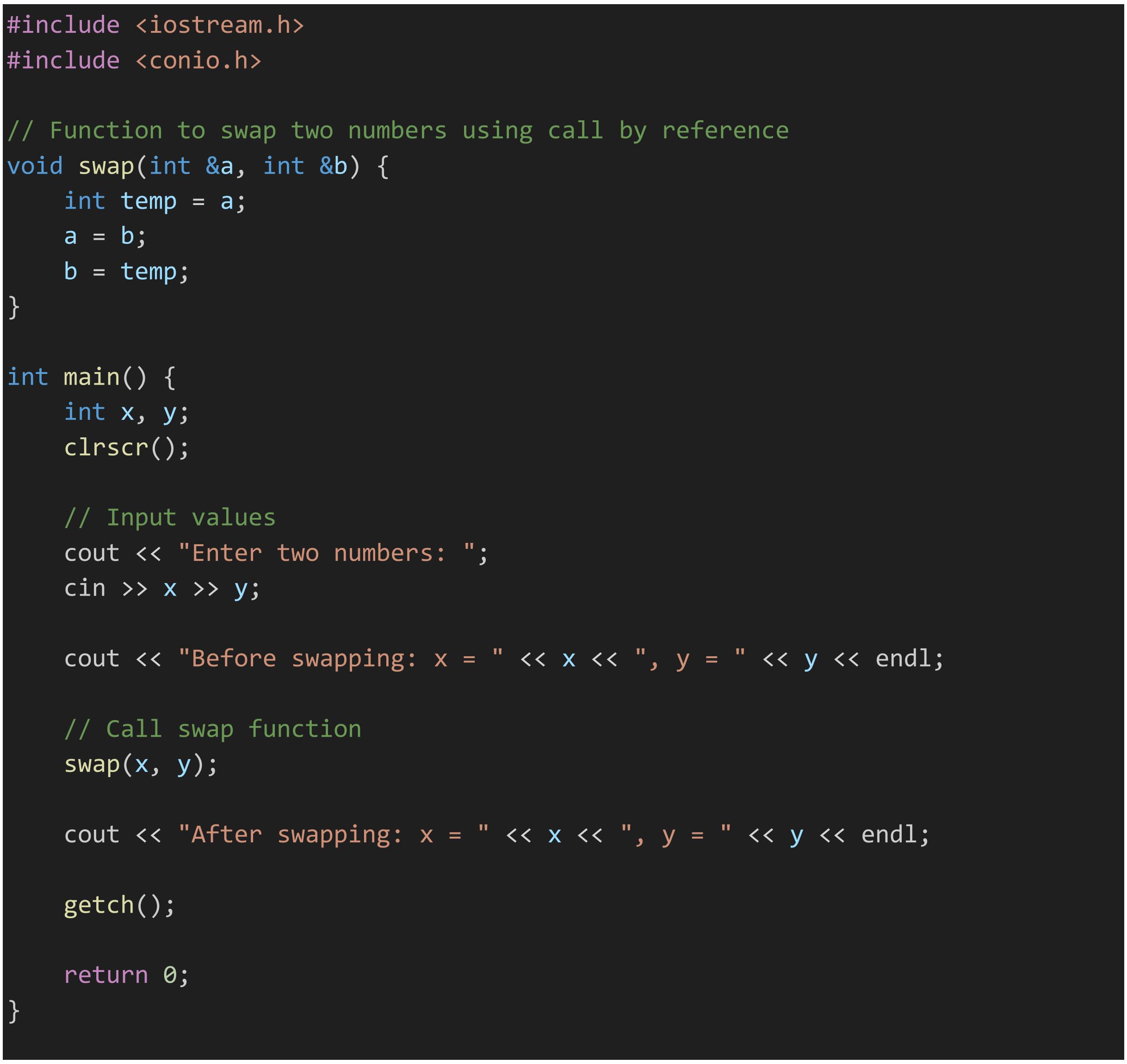
Q. 3) Lab Book **[Marks 10]**

Q. 4) Viva **[Marks 10]**

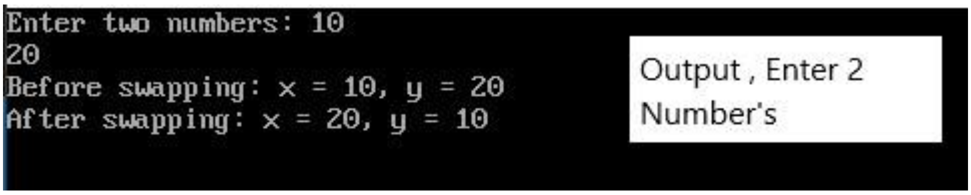
Slip -2

Q. 1)

a) Write a C++ Program to interchange values of two integer numbers. (Use call by reference)
[Marks 15]



```
#include <iostream.h>
#include <conio.h>

// Function to swap two numbers using call by reference
void swap(int &a, int &b) {
    int temp = a;
    a = b;
    b = temp;
}

int main() {
    int x, y;
    clrscr();

    // Input values
    cout << "Enter two numbers: ";
    cin >> x >> y;

    cout << "Before swapping: x = " << x << ", y = " << y << endl;

    // Call swap function
    swap(x, y);

    cout << "After swapping: x = " << x << ", y = " << y << endl;

    getch();

    return 0;
}
```



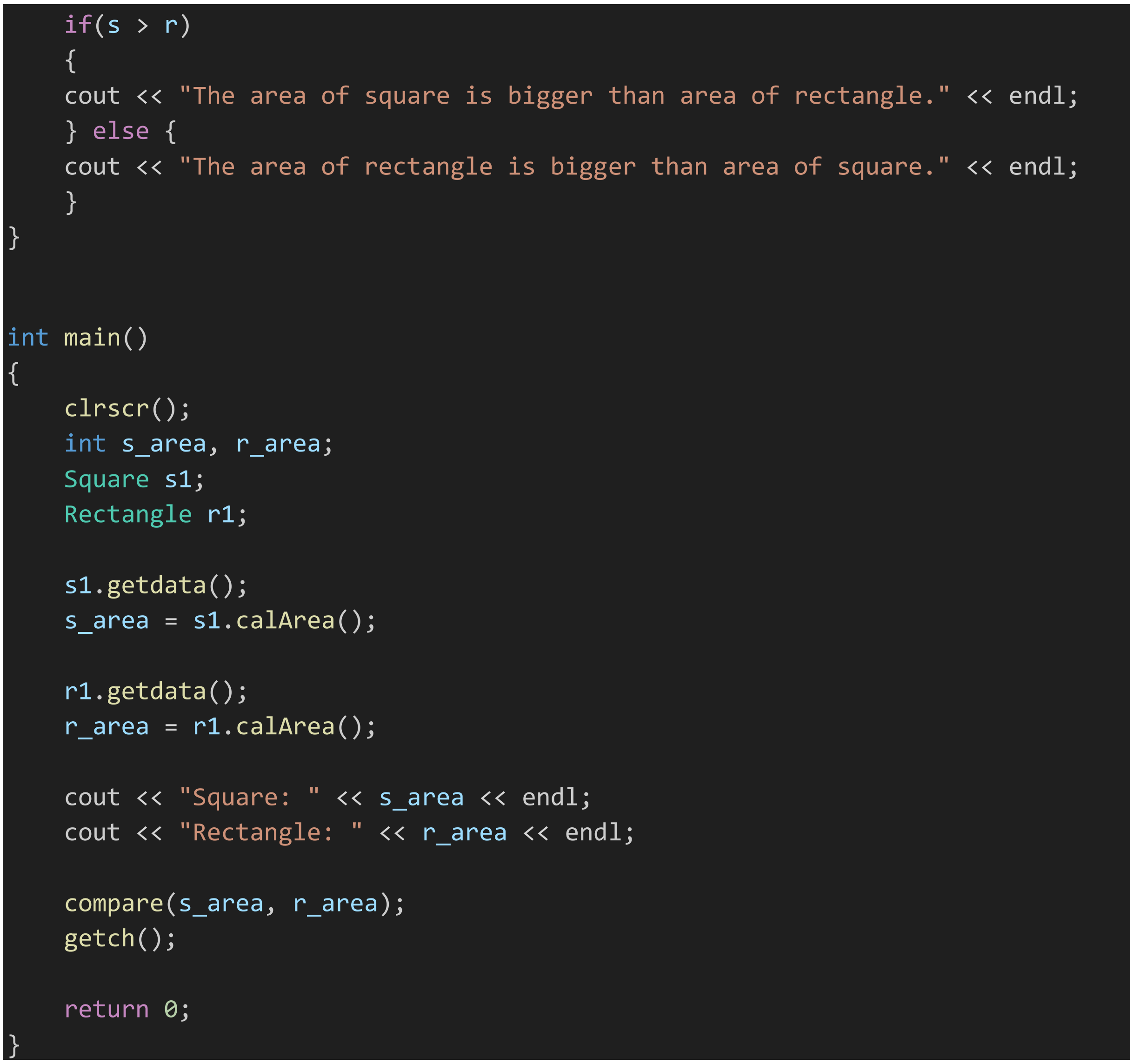
```
Enter two numbers: 10
20
Before swapping: x = 10, y = 20
After swapping: x = 20, y = 10
```

Output , Enter 2 Number's

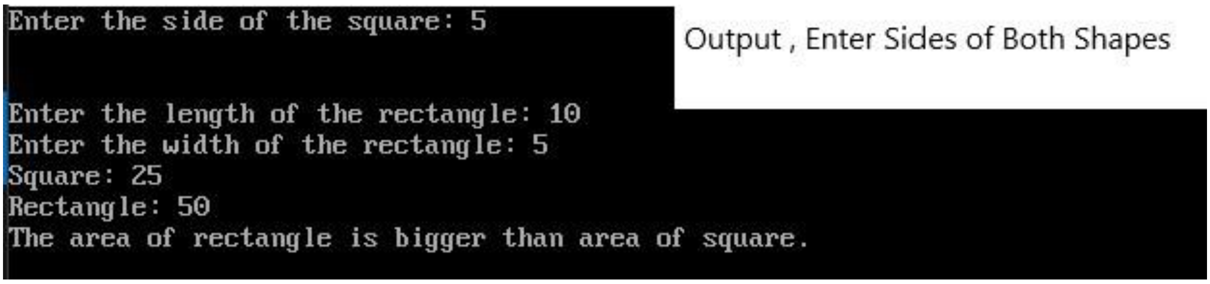
b) Write a C++ Program to create two classes square and rectangle. Compare area of both the shapes using Friend Function. Accept appropriate data members for both the classes.

[Marks 25]

```
#include <iostream.h>
#include <conio.h>
class Square
{
    public:
    int s;

    void getdata()
    {
            cout << "Enter the side of the square: ";
            cin >> s;
    }

    int calArea()
    {
        return (s*s);
    }

    friend void compare(int s, int r);
};

class Rectangle
{
    public:
    int l, w;

    void getdata()
    {
        cout << "\n\nEnter the length of the rectangle: ";
        cin >> l;
        cout << "Enter the width of the rectangle: ";
        cin >> w;
    }

    int calArea()
    {
        return (l*w);
    }

    friend void compare(int s, int r);
};

void compare(int s, int r)
{
```



```
    if(s > r)
    {
    cout << "The area of square is bigger than area of rectangle." << endl;
    } else {
    cout << "The area of rectangle is bigger than area of square." << endl;
    }
}


int main()
{
    clrscr();
    int s_area, r_area;
    Square s1;
    Rectangle r1;

    s1.getdata();
    s_area = s1.calArea();

    r1.getdata();
    r_area = r1.calArea();

    cout << "Square: " << s_area << endl;
    cout << "Rectangle: " << r_area << endl;

    compare(s_area, r_area);
    getch();

    return 0;
}
```

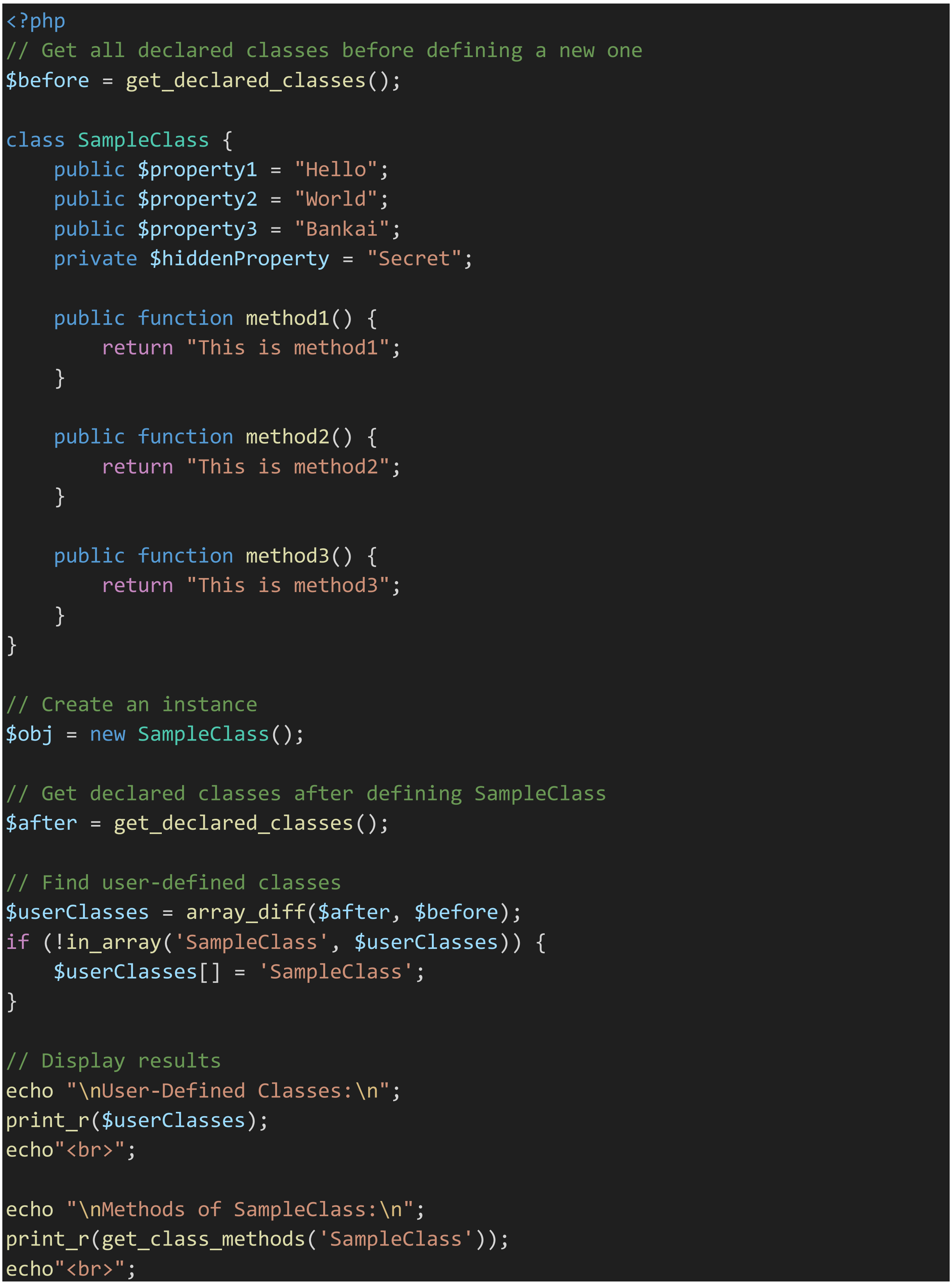


```
Enter the side of the square: 5


Enter the length of the rectangle: 10
Enter the width of the rectangle: 5
Square: 25
Rectangle: 50
The area of rectangle is bigger than area of square.
```

Output , Enter Sides of Both Shapes

Q. 2)

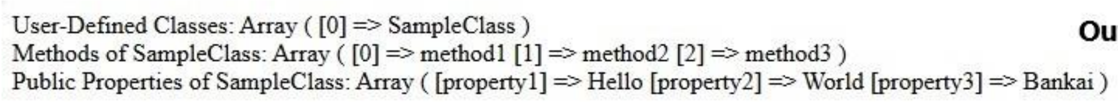
a) Write a PHP script to demonstrate the introspection for examining classes and objects. (use function get_declared_classes() ,get_class_inethods() and get_class_vars()).
[Marks 15]



```php
<?php
// Get all declared classes before defining a new one
$before = get_declared_classes();

class SampleClass {
    public $property1 = "Hello";
    public $property2 = "World";
    public $property3 = "Bankai";
    private $hiddenProperty = "Secret";

    public function method1() {
        return "This is method1";
    }

    public function method2() {
        return "This is method2";
    }

    public function method3() {
        return "This is method3";
    }
}

// Create an instance
$obj = new SampleClass();

// Get declared classes after defining SampleClass
$after = get_declared_classes();

// Find user-defined classes
$userClasses = array_diff($after, $before);
if (!in_array('SampleClass', $userClasses)) {
    $userClasses[] = 'SampleClass';
}

// Display results
echo "\nUser-Defined Classes:\n";
print_r($userClasses);
echo"<br>";

echo "\nMethods of SampleClass:\n";
print_r(get_class_methods('SampleClass'));
echo"<br>";
```

```
echo "\nPublic Properties of SampleClass:\n";
print_r(get_class_vars('SampleClass'));
echo"<br>";
?>
```

Output



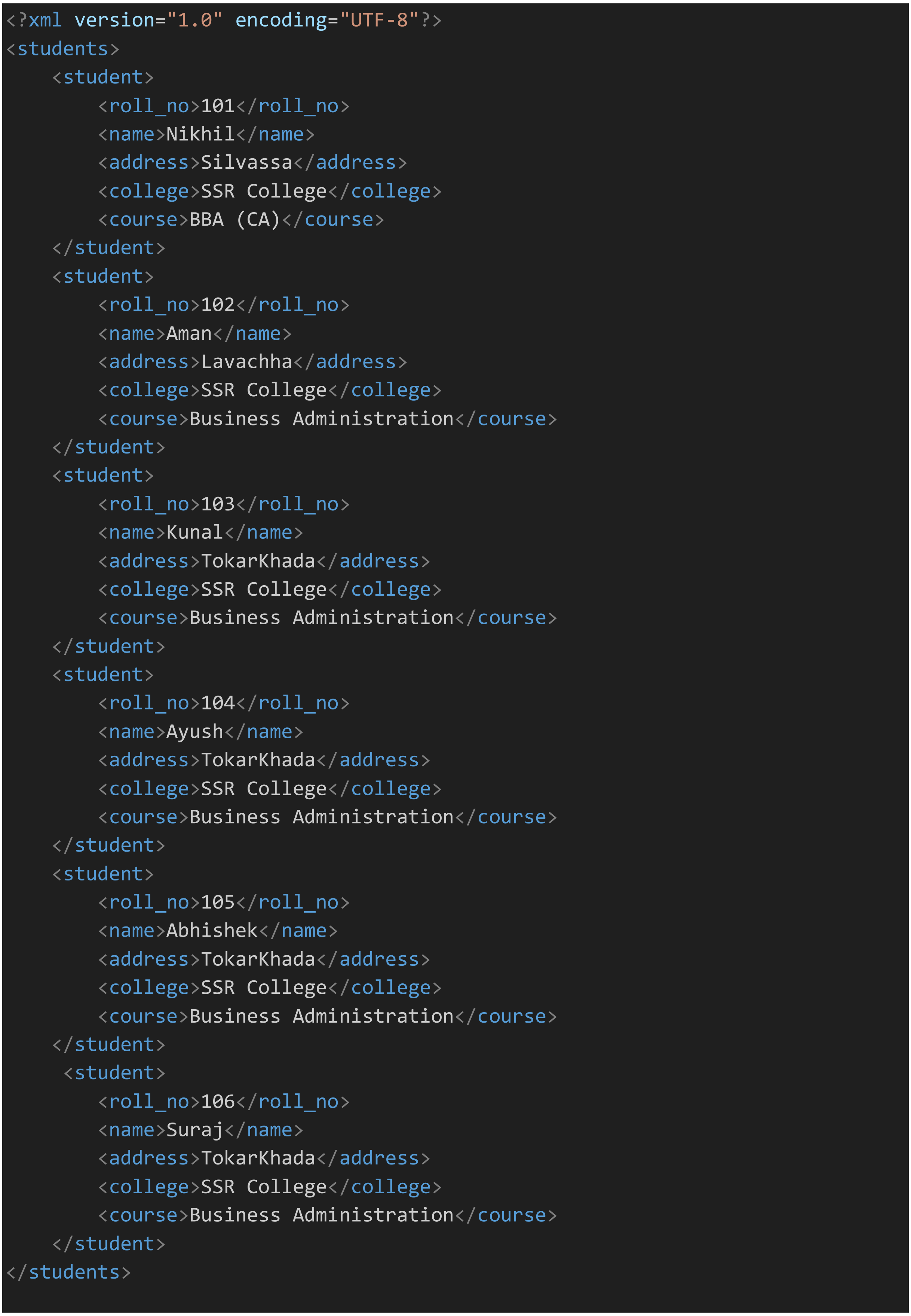
```
User-Defined Classes: Array ( [0] => SampleClass )
Methods of SampleClass: Array ( [0] => method1 [1] => method2 [2] => method3 )
Public Properties of SampleClass: Array ( [property1] => Hello [property2] => World [property3] => Bankai )
```

b) Write a script to solve follow ing questions (Use “Student.xml” file)
[Marks 25]
i) Create a D OM Document object and load this XML file.
ii) Get the output of this Docurrlent to the browser.
iii) Save this [. XML] document in another format i.e. in [.doc]
Write a XML Script to print the names of the student present in “Student.xml” file.

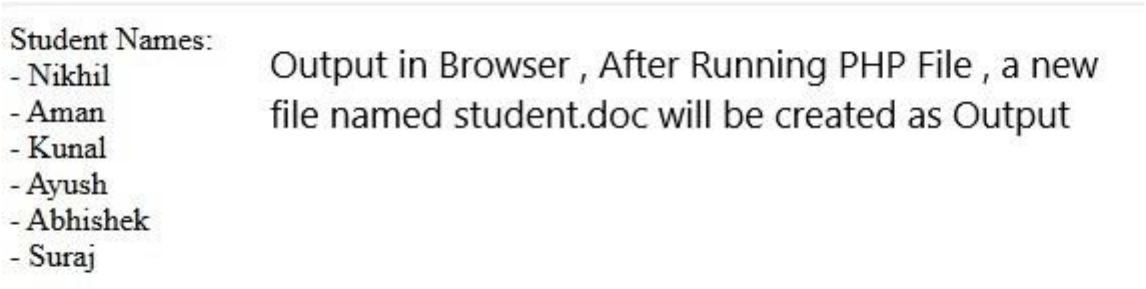
PHP File :-

```
<?php
// Load the XML file
$xml = new DOMDocument();
$xml->load("student.xml");

// Extract and print student names
$students = $xml->getElementsByTagName("student");

echo "Student Names:<br>";
foreach ($students as $student) {
    echo "- " . $student->getElementsByTagName("name")->item(0)->nodeValue .
"<br>";
}
// Save the XML content to a .doc file
file_put_contents("student.doc", $xml->saveXML());
?>
```

Student.xml :-



```
<?xml version="1.0" encoding="UTF-8"?>
<students>
    <student>
        <roll_no>101</roll_no>
        <name>Nikhil</name>
        <address>Silvassa</address>
        <college>SSR College</college>
        <course>BBA (CA)</course>
    </student>
    <student>
        <roll_no>102</roll_no>
        <name>Aman</name>
        <address>Lavachha</address>
        <college>SSR College</college>
        <course>Business Administration</course>
    </student>
    <student>
        <roll_no>103</roll_no>
        <name>Kunal</name>
        <address>TokarKhada</address>
        <college>SSR College</college>
        <course>Business Administration</course>
    </student>
    <student>
        <roll_no>104</roll_no>
        <name>Ayush</name>
        <address>TokarKhada</address>
        <college>SSR College</college>
        <course>Business Administration</course>
    </student>
    <student>
        <roll_no>105</roll_no>
        <name>Abhishek</name>
        <address>TokarKhada</address>
        <college>SSR College</college>
        <course>Business Administration</course>
    </student>
     <student>
        <roll_no>106</roll_no>
        <name>Suraj</name>
        <address>TokarKhada</address>
        <college>SSR College</college>
        <course>Business Administration</course>
    </student>
</students>
```



```
Student Names:
- Nikhil
- Aman
- Kunal
- Ayush
- Abhishek
- Suraj
```

Output in Browser , After Running PHP File , a new file named student.doc will be created as Output

Q. 3) Lab Book **[Marks 10]**

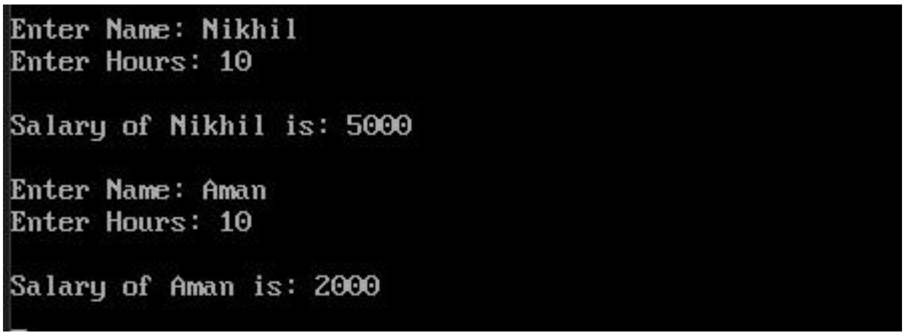
Q. 4) Viva **[Marks 10]**

Slip -3

Q. 1)

a) Write a C++ program to accept Worker information worker name, no of hours worked, pay rate and salary. Write necessary functions to calculate and display the salary of worker. (Use default value for pay_rate) [Marks 15]

```
#include <iostream.h>
#include <conio.h>

class Worker {
    string name;
    int hours;

public:
    void accept() {
        cout << "\nEnter Name: ";
        cin >> name;
        cout << "Enter Hours: ";
        cin >> hours;
    }

    void calculateSalary(int rate = 20) {  // Default parameter in declaration
        cout << "\nSalary of " << name << " is: " << (hours * 10) * rate <<
endl;
    }
};

int main() {
    clrscr();
    Worker w1, w2;

    w1.accept();
    w1.calculateSalary(50);  // Custom rate

    w2.accept();
    w2.calculateSalary();  // Default rate

    getch();
    return 0;
}
```



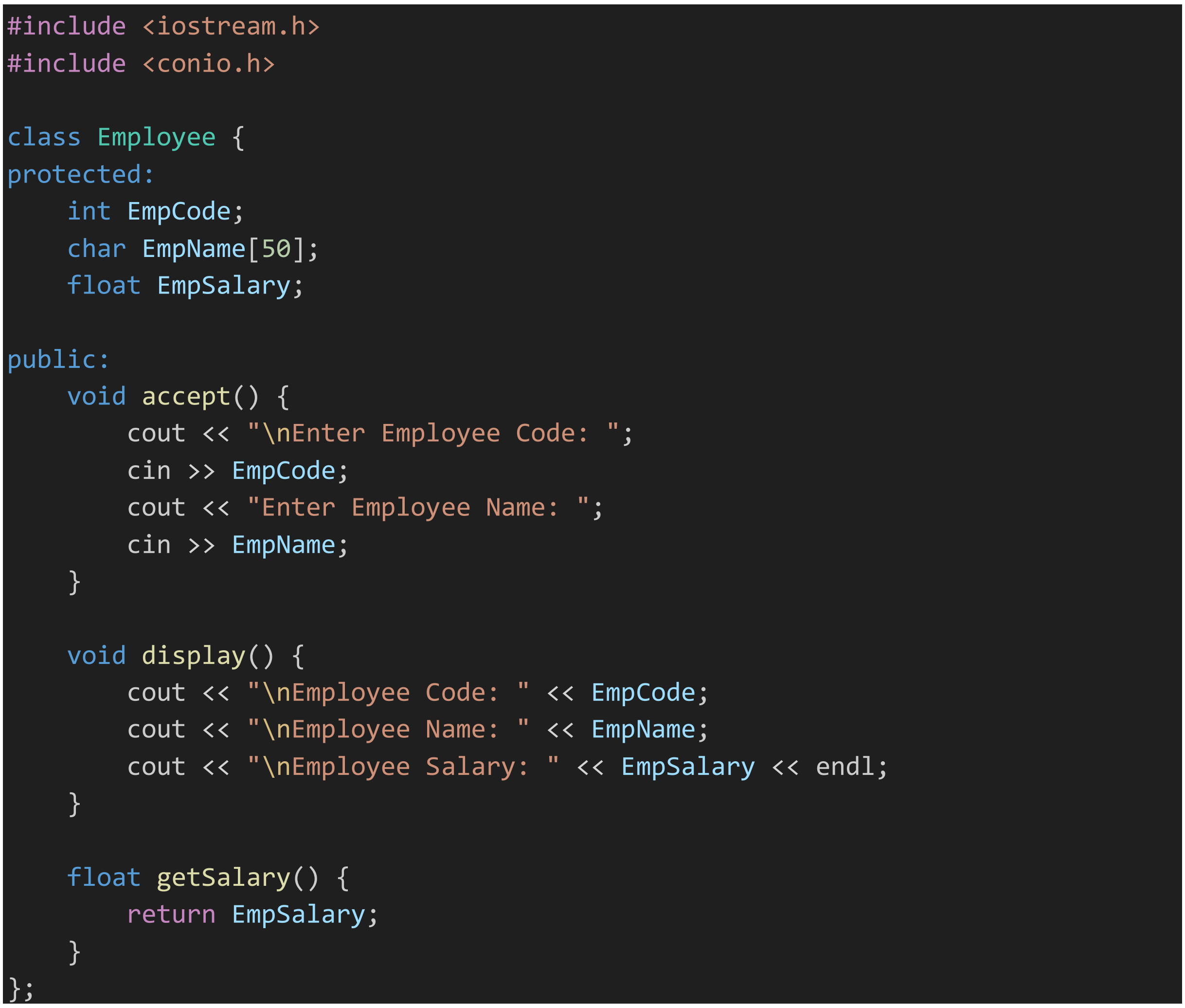
```
Enter Name: Nikhil
Enter Hours: 10

Salary of Nikhil is: 5000

Enter Name: Aman
Enter Hours: 10

Salary of Aman is: 2000
```

b) Write a C++ Program to create a base class Employee(EmpCode, Empname, EmpSalary). Derive two classes as fulltime(DailyWages, No_Of_Days) and Parttime (No_of_Hrs, Hr_Wages).

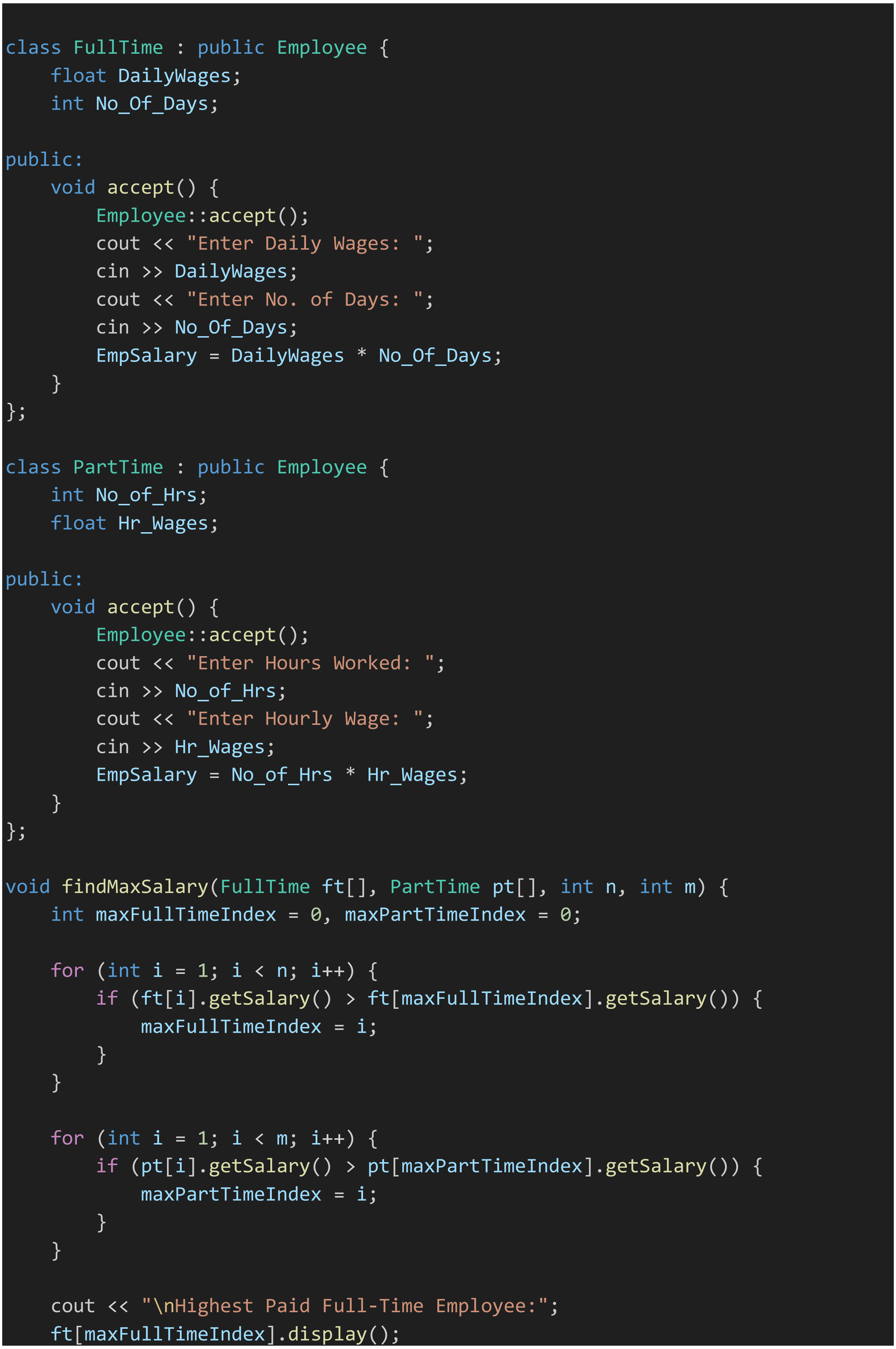
Write a menu driven program to perform following functions.

1. **Accept the details of ‘n’ employees and calculate salary.**
2. **Display the details of ‘n’ employees.**
3. **Display the details of employee having maximum salary for both types of employees**

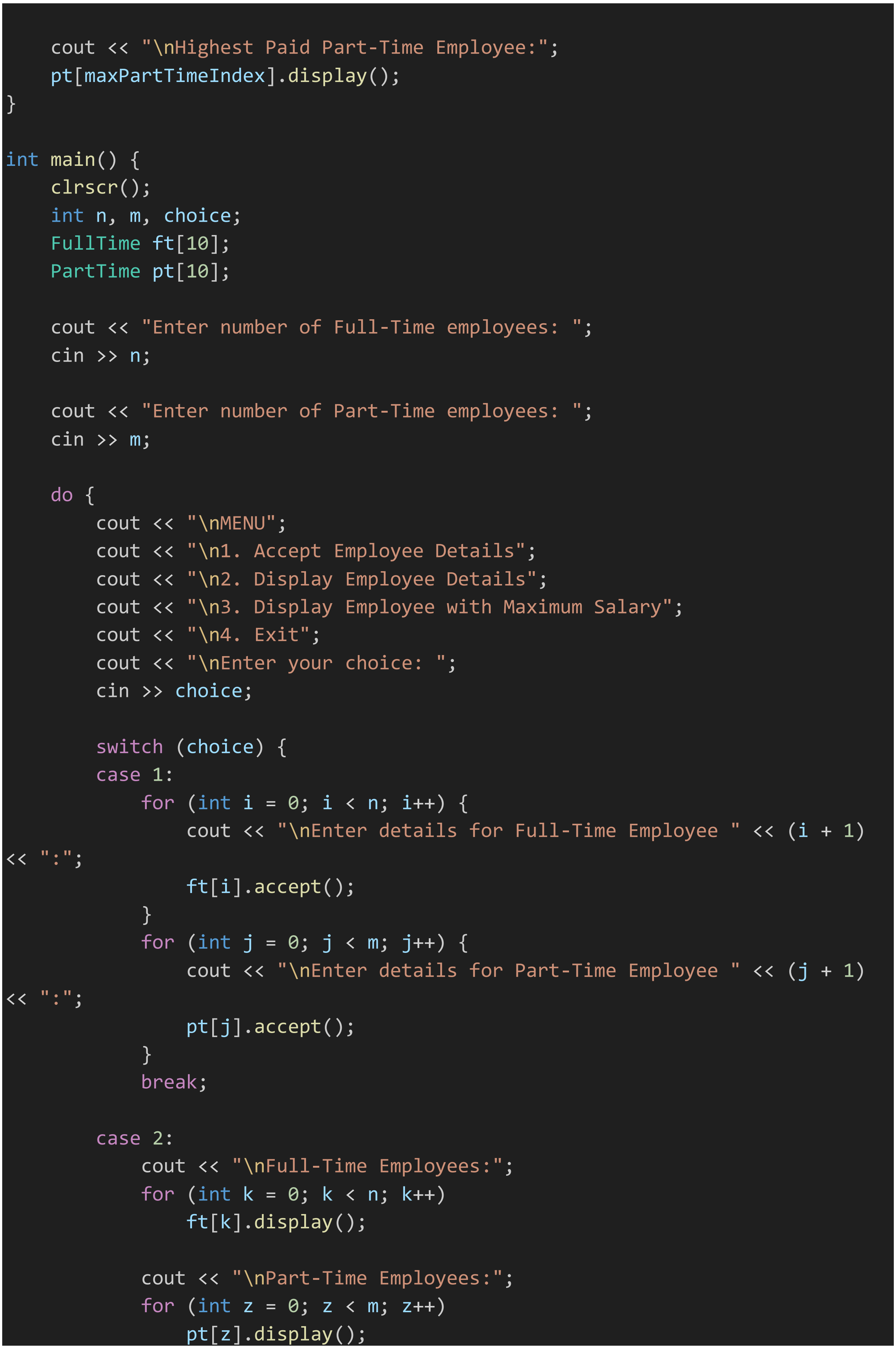
[Marks 25]



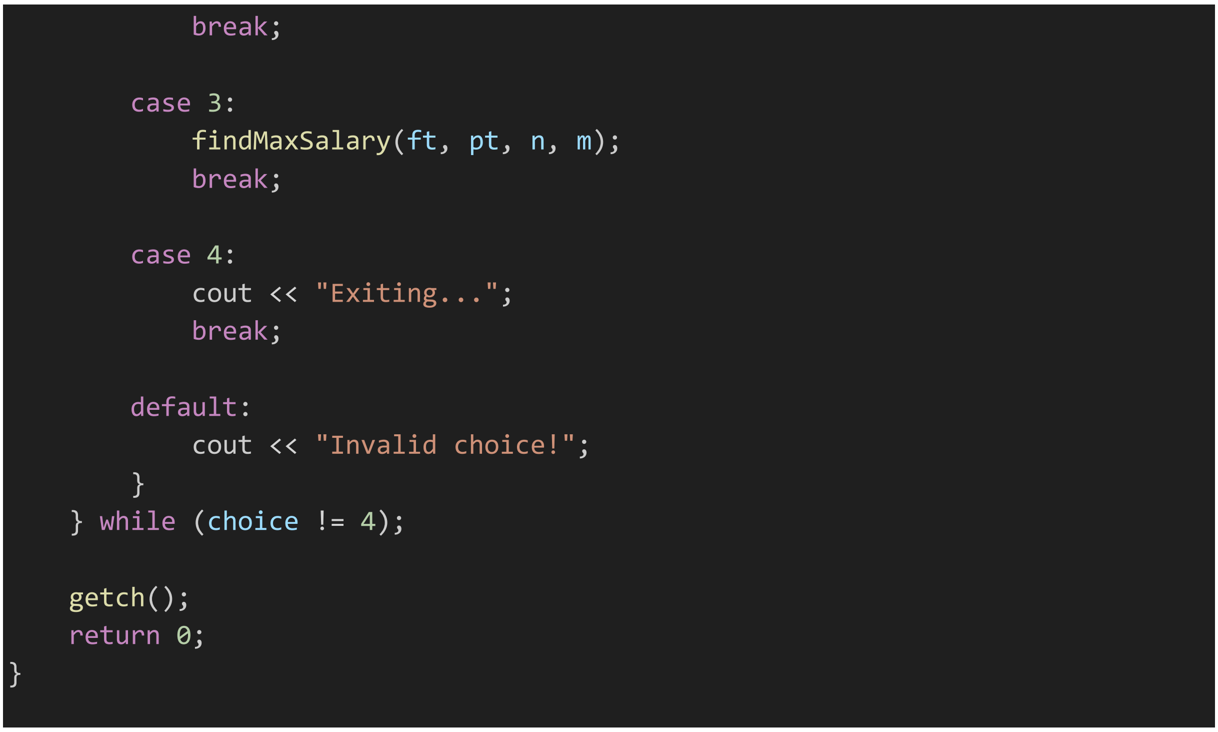
```
#include <iostream.h>
#include <conio.h>

class Employee {
protected:
    int EmpCode;
    char EmpName[50];
    float EmpSalary;

public:
    void accept() {
        cout << "\nEnter Employee Code: ";
        cin >> EmpCode;
        cout << "Enter Employee Name: ";
        cin >> EmpName;
    }

    void display() {
        cout << "\nEmployee Code: " << EmpCode;
        cout << "\nEmployee Name: " << EmpName;
        cout << "\nEmployee Salary: " << EmpSalary << endl;
    }

    float getSalary() {
        return EmpSalary;
    }
};
```



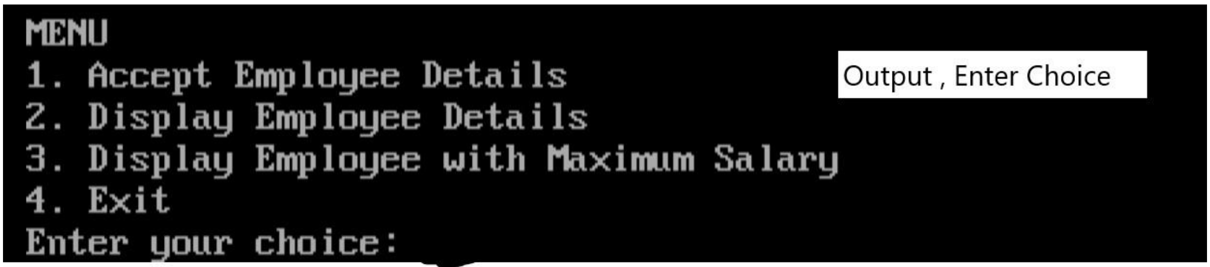
```
class FullTime : public Employee {
    float DailyWages;
    int No_Of_Days;

public:
    void accept() {
        Employee::accept();
        cout << "Enter Daily Wages: ";
        cin >> DailyWages;
        cout << "Enter No. of Days: ";
        cin >> No_Of_Days;
        EmpSalary = DailyWages * No_Of_Days;
    }
};

class PartTime : public Employee {
    int No_of_Hrs;
    float Hr_Wages;

public:
    void accept() {
        Employee::accept();
        cout << "Enter Hours Worked: ";
        cin >> No_of_Hrs;
        cout << "Enter Hourly Wage: ";
        cin >> Hr_Wages;
        EmpSalary = No_of_Hrs * Hr_Wages;
    }
};

void findMaxSalary(FullTime ft[], PartTime pt[], int n, int m) {
    int maxFullTimeIndex = 0, maxPartTimeIndex = 0;

    for (int i = 1; i < n; i++) {
        if (ft[i].getSalary() > ft[maxFullTimeIndex].getSalary()) {
            maxFullTimeIndex = i;
        }
    }

    for (int i = 1; i < m; i++) {
        if (pt[i].getSalary() > pt[maxPartTimeIndex].getSalary()) {
            maxPartTimeIndex = i;
        }
    }

    cout << "\nHighest Paid Full-Time Employee:";
    ft[maxFullTimeIndex].display();
```



```
    cout << "\nHighest Paid Part-Time Employee:";
    pt[maxPartTimeIndex].display();
}

int main() {
    clrscr();
    int n, m, choice;
    FullTime ft[10];
    PartTime pt[10];

    cout << "Enter number of Full-Time employees: ";
    cin >> n;

    cout << "Enter number of Part-Time employees: ";
    cin >> m;

    do {
        cout << "\nMENU";
        cout << "\n1. Accept Employee Details";
        cout << "\n2. Display Employee Details";
        cout << "\n3. Display Employee with Maximum Salary";
        cout << "\n4. Exit";
        cout << "\nEnter your choice: ";
        cin >> choice;

        switch (choice) {
        case 1:
            for (int i = 0; i < n; i++) {
                cout << "\nEnter details for Full-Time Employee " << (i + 1)
<< ":";
                ft[i].accept();
            }
            for (int j = 0; j < m; j++) {
                cout << "\nEnter details for Part-Time Employee " << (j + 1)
<< ":";
                pt[j].accept();
            }
            break;

        case 2:
            cout << "\nFull-Time Employees:";
            for (int k = 0; k < n; k++)
                ft[k].display();

            cout << "\nPart-Time Employees:";
            for (int z = 0; z < m; z++)
                pt[z].display();
```

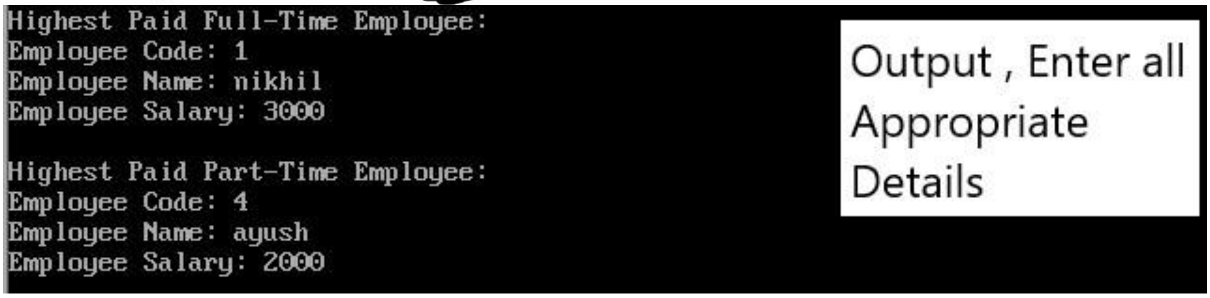


```
            break;

        case 3:
            findMaxSalary(ft, pt, n, m);
            break;

        case 4:
            cout << "Exiting...";
            break;

        default:
            cout << "Invalid choice!";
        }
    } while (choice != 4);

    getch();
    return 0;
}
```



```
MENU
1. Accept Employee Details
2. Display Employee Details
3. Display Employee with Maximum Salary
4. Exit
Enter your choice:
```

Output , Enter Choice



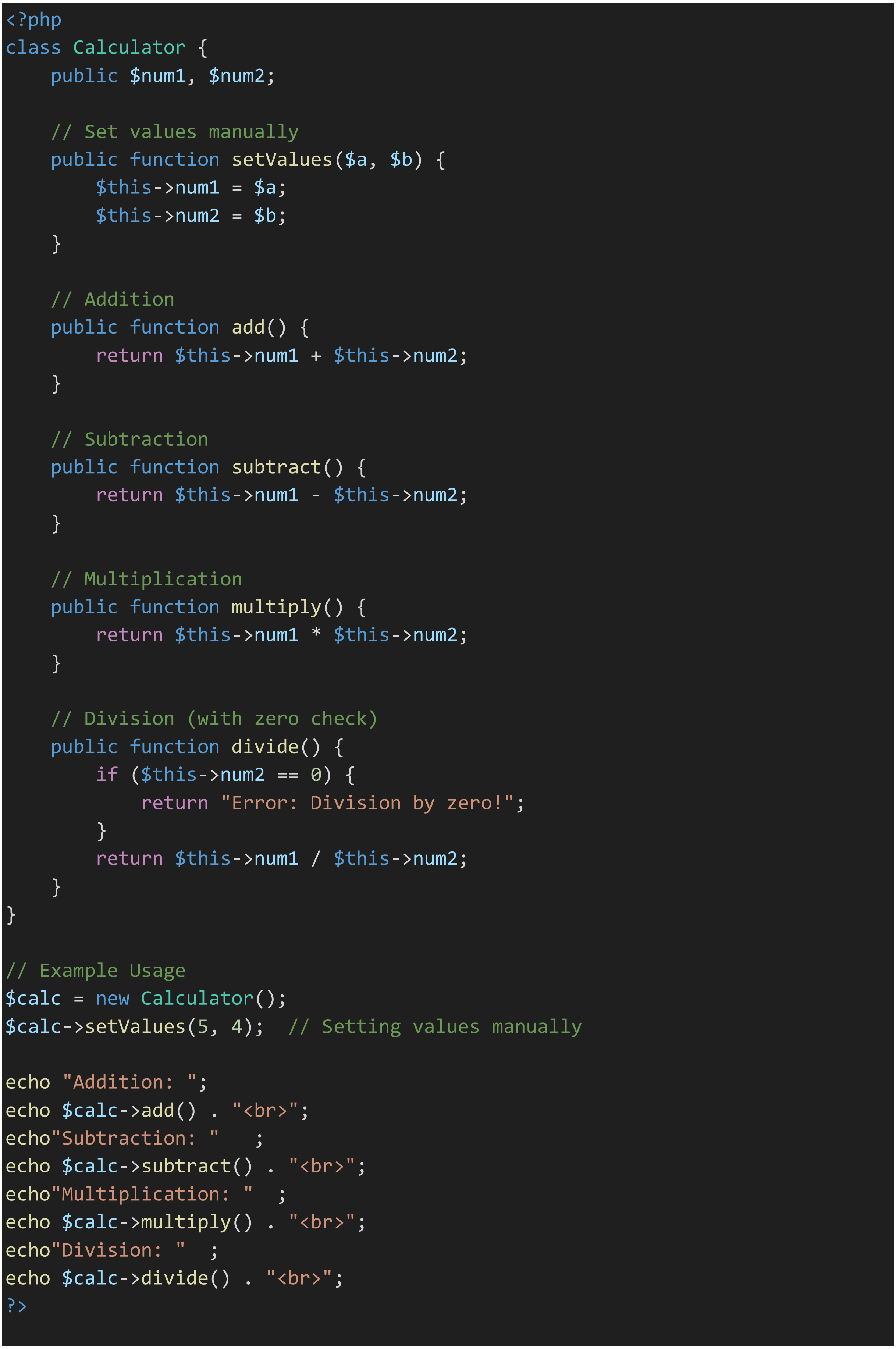
```
Highest Paid Full-Time Employee:
Employee Code: 1
Employee Name: nikhil
Employee Salary: 3000

Highest Paid Part-Time Employee:
Employee Code: 4
Employee Name: ayush
Employee Salary: 2000
```

Output , Enter all Appropriate Details

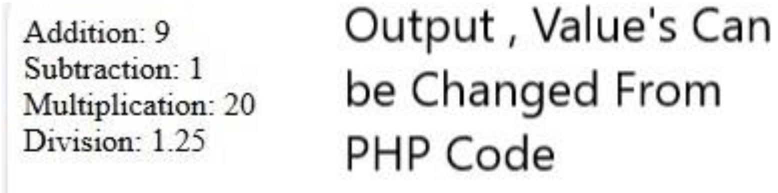
Q. 2)

a) Write a Calculator class that can accept two values, then add, subtract, multiply them or divide them on request.
For example:

$calc = new C alculator(3, 4); echo $ calc- > add(); // Displays "7"
echo $ calc- > multiply(); // Displays "12" **[Marks l5]**



```php
<?php
class Calculator {
    public $num1, $num2;

    // Set values manually
    public function setValues($a, $b) {
        $this->num1 = $a;
        $this->num2 = $b;
    }

    // Addition
    public function add() {
        return $this->num1 + $this->num2;
    }

    // Subtraction
    public function subtract() {
        return $this->num1 - $this->num2;
    }

    // Multiplication
    public function multiply() {
        return $this->num1 * $this->num2;
    }

    // Division (with zero check)
    public function divide() {
        if ($this->num2 == 0) {
            return "Error: Division by zero!";
        }
        return $this->num1 / $this->num2;
    }
}

// Example Usage
$calc = new Calculator();
$calc->setValues(5, 4);  // Setting values manually

echo "Addition: ";
echo $calc->add() . "<br>";
echo"Subtraction: "   ;
echo $calc->subtract() . "<br>";
echo"Multiplication: "  ;
echo $calc->multiply() . "<br>";
echo"Division: "  ;
echo $calc->divide() . "<br>";
?>
```



```
Addition: 9
Subtraction: 1
Multiplication: 20
Division: 1.25
```

Output , Value's Can be Changed From PHP Code

b) Write a script to create “cricket.xml” file with multiple elements as shown below:

< CricketTeam>
< Team country=”Australia”>
< players>   </players>
<runs>   </runs>
<wicket>   </wicket>
</Team>
</CricketTeam>

Write a script to add multiple elements in “cricket.xml” file of category, country=”India”.[Marks 25]

```php
<?php
$root = new SimpleXMLElement("<root/>");

$e1 = $root->addChild("CricketTeam");


$e2 = $e1->addChild("team");
$e2->addAttribute("country", "India");
$e2->addChild("player", "Virat Kohli");
$e2->addChild("run", "10000");
$e2->addChild("wicket", "5");

$e2 = $e1->addChild("team");
$e2->addAttribute("country", "Pakistan");
$e2->addChild("player", "Babar Azam");
$e2->addChild("run", "1000");
$e2->addChild("wicket", "2");

$e2 = $e1->addChild("team");
$e2->addAttribute("country", "India");
$e2->addChild("player", "Rohit Sharma");
$e2->addChild("run", "5000");
$e2->addChild("wicket", "4");


$root->asXML("cricket.xml");

?>
```

**** Output :- Result in Creating new file in same Directory , named << cricket.xml >> ****

Q. 3) Lab Book **[Marks 10]**

Q. 4) Viva **[Marks 10]**

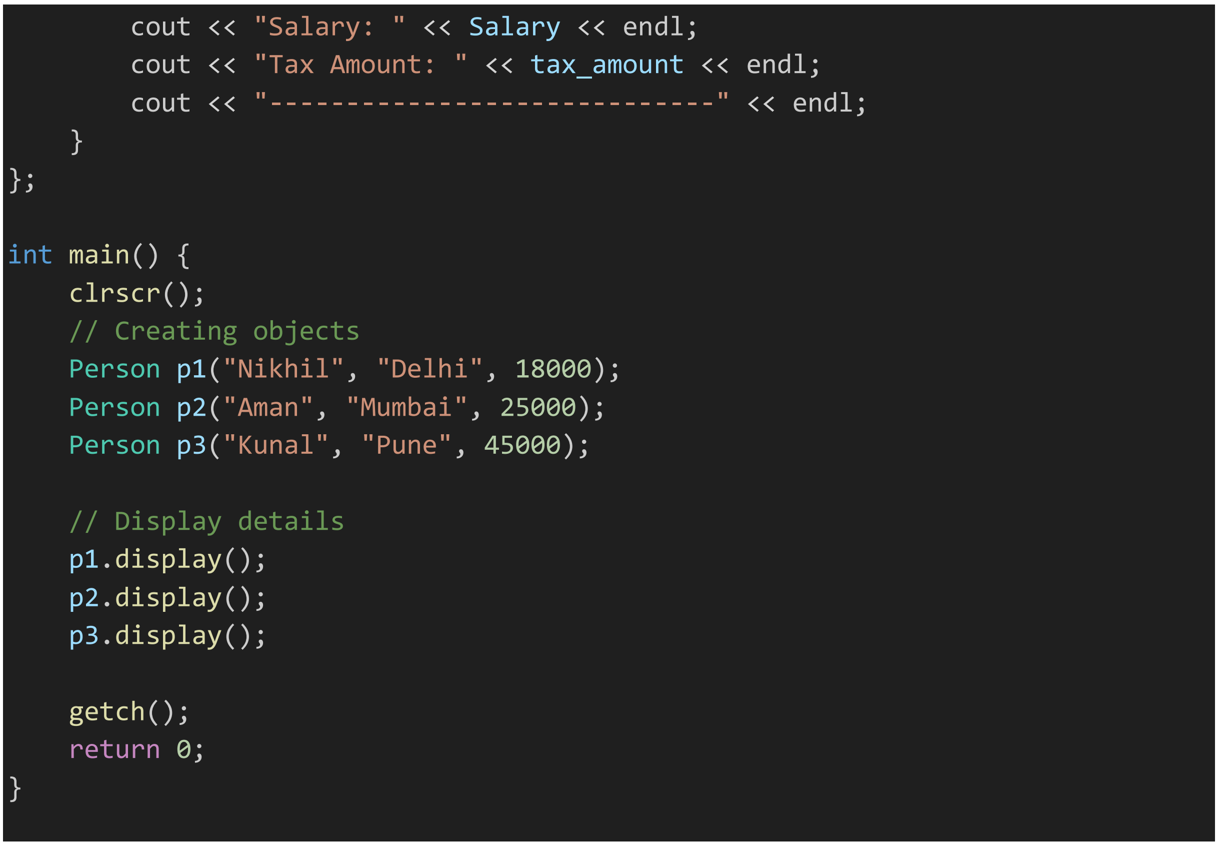
Slip -4

Q. 1)

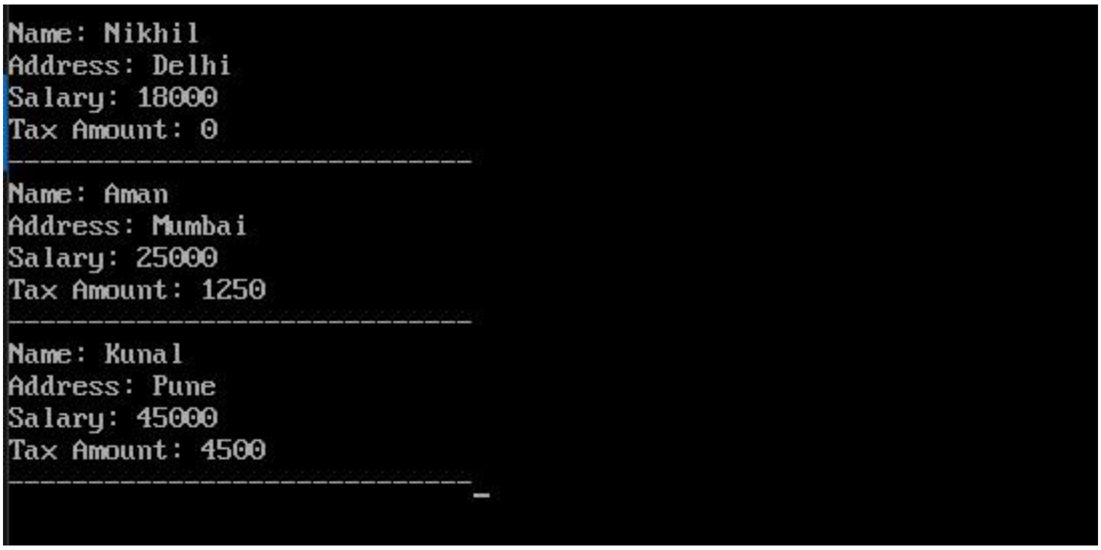
a) Consider the following C++ Class

Class Person
{
Char Name[20];
Char Add[30];
Float Salary;
Float Tax_Amt;
Public :
};
Calculate tax amount by checking salary of a person
-> for salary <=20000 Tax Rate = 0
-> for salary >20000 || <=40000 Tax Rate = 5% of salary
-> for salary >40000 Tax Rate = 10% of salary [Marks 15]

```
#include <iostream.h>
#include <string.h>
#include <conio.h>


class Person {
    char Name[50];
    char Addr[100];
    float Salary;
    float tax_amount;

public:
    // Constructor to initialize values
    Person(const char name[], const char addr[], float salary) {
       strcpy(Name , name); // copy string
       strcpy(Addr, addr);
        Salary = salary;
        tax_amount = calculateTax();
    }

    // Function to calculate tax
    float calculateTax() {
        if (Salary <= 20000) return 0;
        else if (Salary <= 40000) return Salary * 0.05;
        else return Salary * 0.10;
    }

    // Function to display details
    void display() {
        cout << "Name: " << Name << endl;
        cout << "Address: " << Addr << endl;
```



```
        cout << "Salary: " << Salary << endl;
        cout << "Tax Amount: " << tax_amount << endl;
        cout << "-----------------------------" << endl;
    }
};

int main() {
    clrscr();
    // Creating objects
    Person p1("Nikhil", "Delhi", 18000);
    Person p2("Aman", "Mumbai", 25000);
    Person p3("Kunal", "Pune", 45000);

    // Display details
    p1.display();
    p2.display();
    p3.display();

    getch();
    return 0;
}
```



```
Name: Nikhil
Address: Delhi
Salary: 18000
Tax Amount: 0
-----------------------------
Name: Aman
Address: Mumbai
Salary: 25000
Tax Amount: 1250
-----------------------------
Name: Kunal
Address: Pune
Salary: 45000
Tax Amount: 4500
-----------------------------_
```

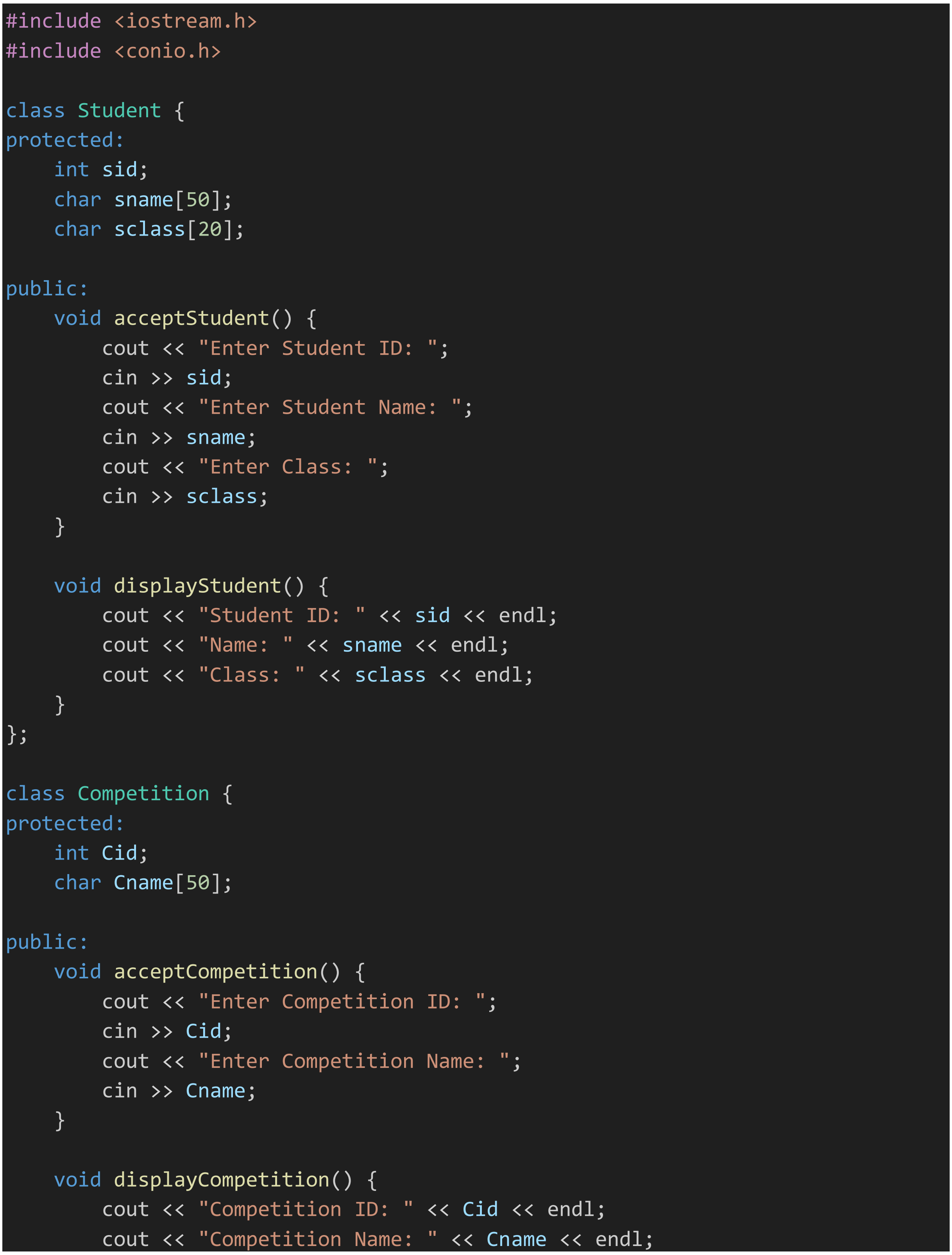
b) Design two base classes student(sid,sname,class) and Competition(Cid, Cname). Derive a class Stud_Comp(Rank) from it. Write a menu driven program to perform following functions.

i. Accept Information

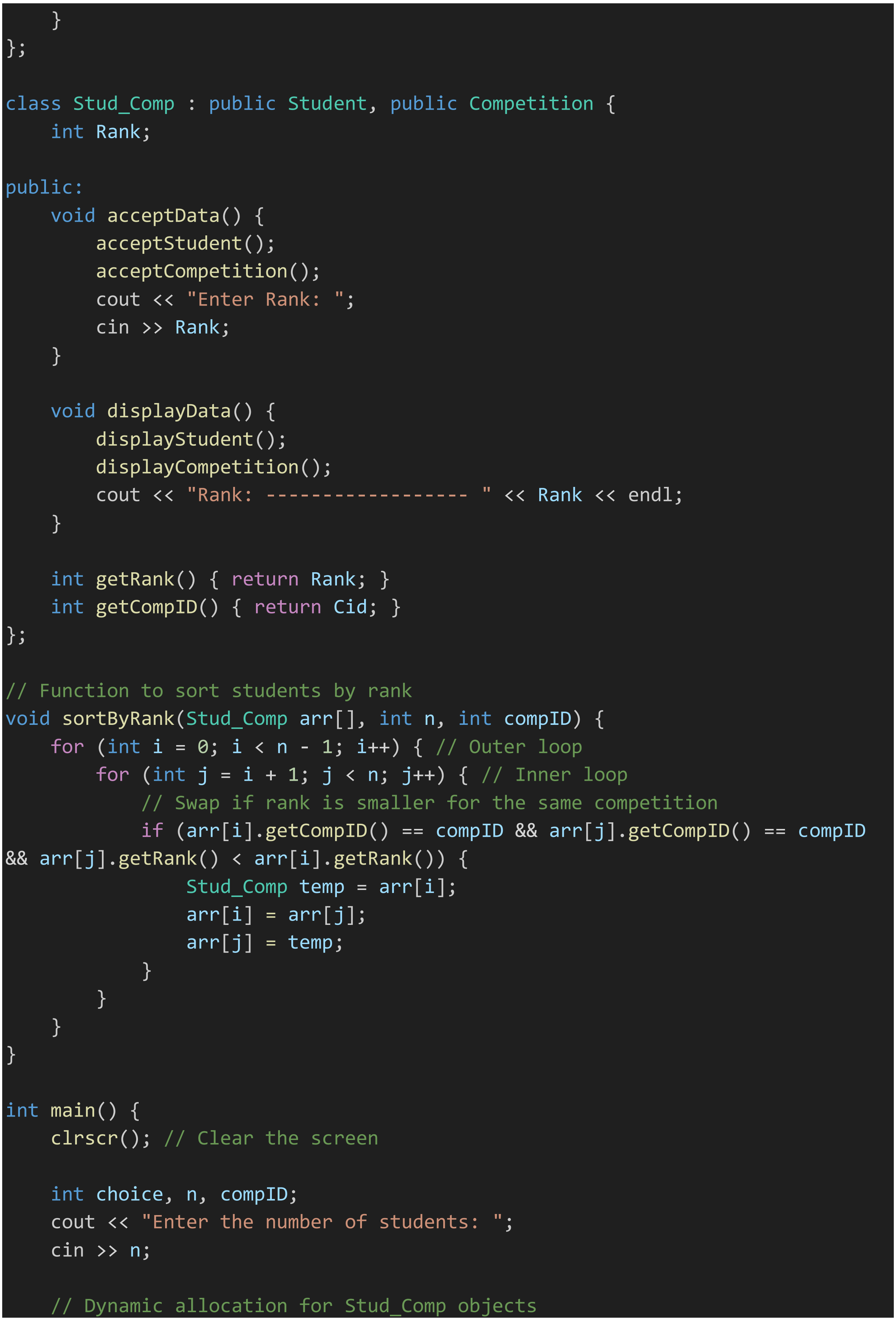
ii. Display Information

iii. Display student details in the ascending order of rank of a specified competition.

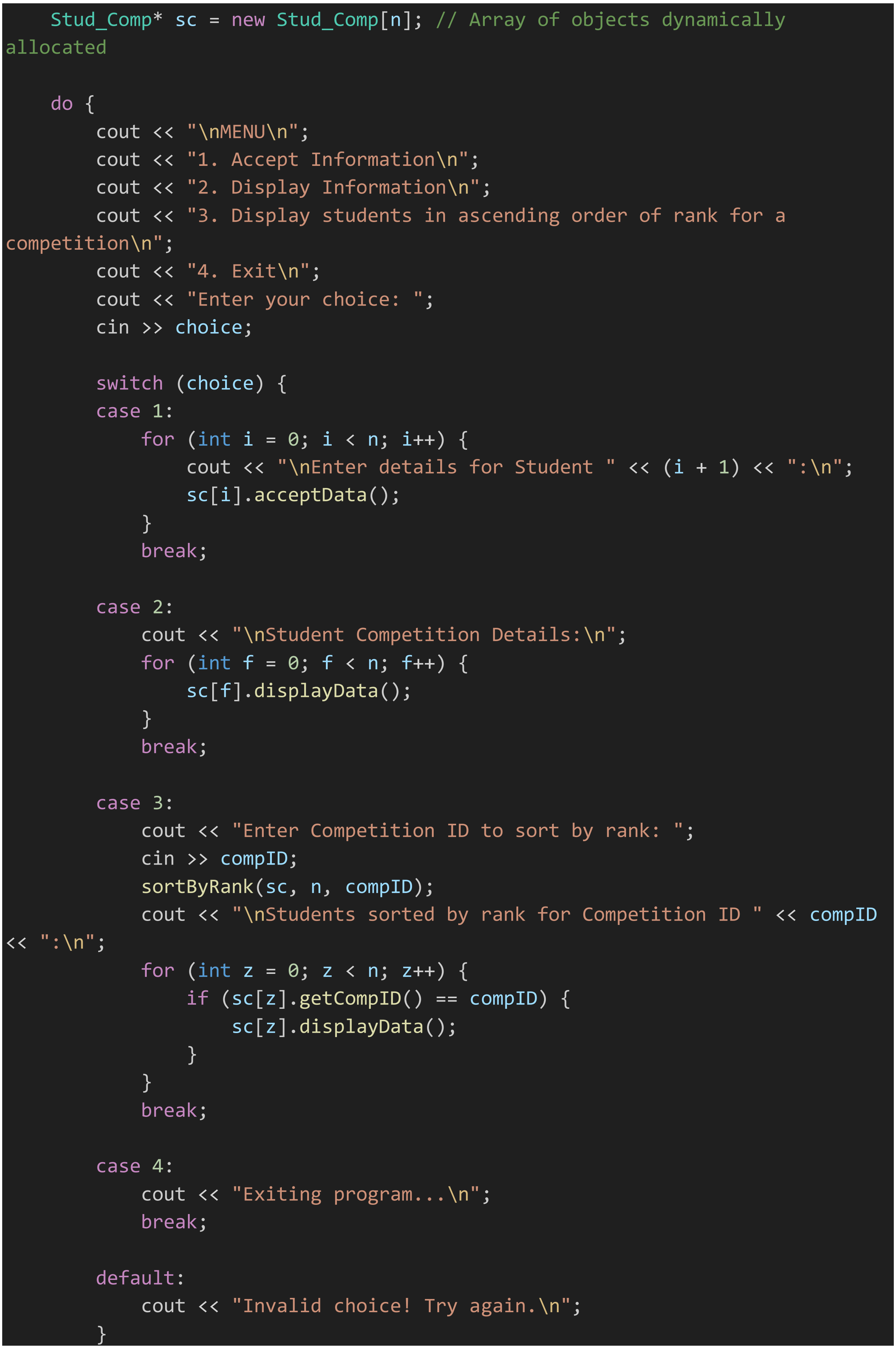
(Use array of objects)
[Marks 25]



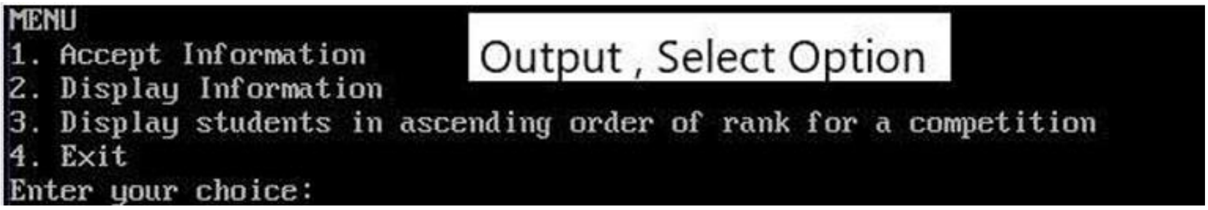
```
#include <iostream.h>
#include <conio.h>

class Student {
protected:
    int sid;
    char sname[50];
    char sclass[20];

public:
    void acceptStudent() {
        cout << "Enter Student ID: ";
        cin >> sid;
        cout << "Enter Student Name: ";
        cin >> sname;
        cout << "Enter Class: ";
        cin >> sclass;
    }

    void displayStudent() {
        cout << "Student ID: " << sid << endl;
        cout << "Name: " << sname << endl;
        cout << "Class: " << sclass << endl;
    }
};

class Competition {
protected:
    int Cid;
    char Cname[50];

public:
    void acceptCompetition() {
        cout << "Enter Competition ID: ";
        cin >> Cid;
        cout << "Enter Competition Name: ";
        cin >> Cname;
    }

    void displayCompetition() {
        cout << "Competition ID: " << Cid << endl;
        cout << "Competition Name: " << Cname << endl;
```



```
    }
};

class Stud_Comp : public Student, public Competition {
    int Rank;

public:
    void acceptData() {
        acceptStudent();
        acceptCompetition();
        cout << "Enter Rank: ";
        cin >> Rank;
    }

    void displayData() {
        displayStudent();
        displayCompetition();
        cout << "Rank: ------------------ " << Rank << endl;
    }

    int getRank() { return Rank; }
    int getCompID() { return Cid; }
};

// Function to sort students by rank
void sortByRank(Stud_Comp arr[], int n, int compID) {
    for (int i = 0; i < n - 1; i++) { // Outer loop
        for (int j = i + 1; j < n; j++) { // Inner loop
            // Swap if rank is smaller for the same competition
            if (arr[i].getCompID() == compID && arr[j].getCompID() == compID
&& arr[j].getRank() < arr[i].getRank()) {
                Stud_Comp temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }
    }
}

int main() {
    clrscr(); // Clear the screen

    int choice, n, compID;
    cout << "Enter the number of students: ";
    cin >> n;

    // Dynamic allocation for Stud_Comp objects
```

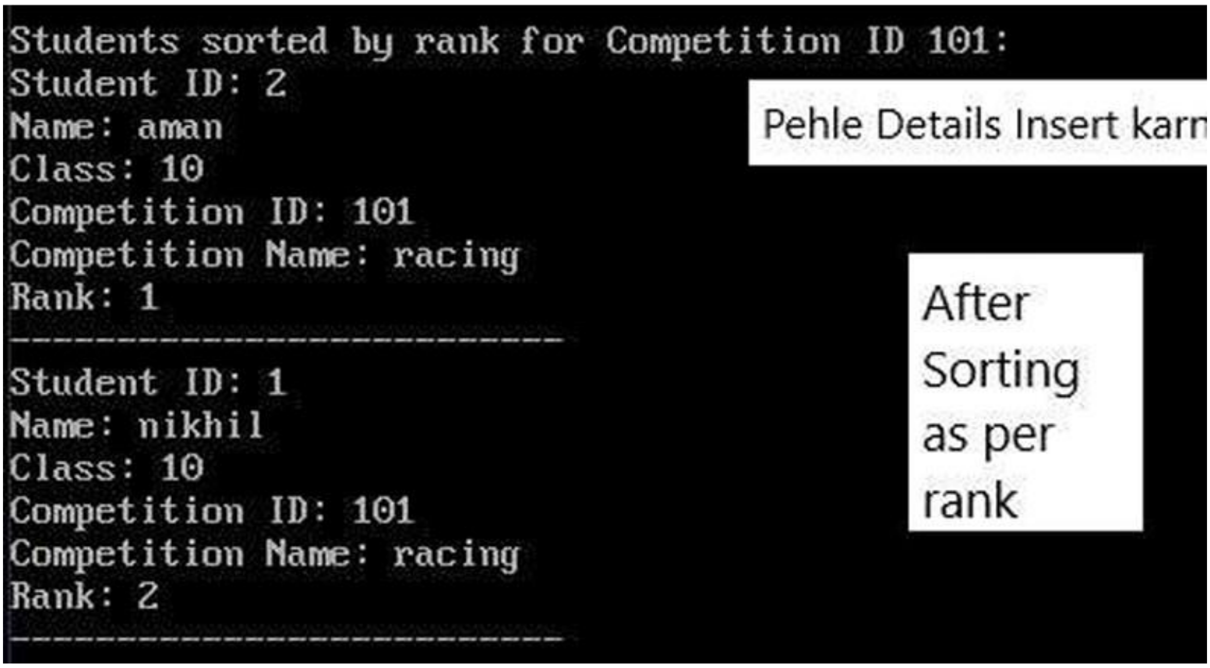


```cpp
    Stud_Comp* sc = new Stud_Comp[n]; // Array of objects dynamically allocated

    do {
        cout << "\nMENU\n";
        cout << "1. Accept Information\n";
        cout << "2. Display Information\n";
        cout << "3. Display students in ascending order of rank for a competition\n";
        cout << "4. Exit\n";
        cout << "Enter your choice: ";
        cin >> choice;

        switch (choice) {
        case 1:
            for (int i = 0; i < n; i++) {
                cout << "\nEnter details for Student " << (i + 1) << ":\n";
                sc[i].acceptData();
            }
            break;

        case 2:
            cout << "\nStudent Competition Details:\n";
            for (int f = 0; f < n; f++) {
                sc[f].displayData();
            }
            break;

        case 3:
            cout << "Enter Competition ID to sort by rank: ";
            cin >> compID;
            sortByRank(sc, n, compID);
            cout << "\nStudents sorted by rank for Competition ID " << compID << ":\n";
            for (int z = 0; z < n; z++) {
                if (sc[z].getCompID() == compID) {
                    sc[z].displayData();
                }
            }
            break;

        case 4:
            cout << "Exiting program...\n";
            break;

        default:
            cout << "Invalid choice! Try again.\n";
        }
    } while (choice != 4);

    delete[] sc; // Free dynamically allocated memory
    getch(); // Wait for key press before closing
    return 0;
}
```



```
MENU
1. Accept Information
2. Display Information
3. Display students in ascending order of rank for a competition
4. Exit
Enter your choice:
```

Output , Select Option



```
Students sorted by rank for Competition ID 101:
Student ID: 2
Name: aman
Class: 10
Competition ID: 101
Competition Name: racing
Rank: 1
------------------------------
Student ID: 1
Name: nikhil
Class: 10
Competition ID: 101
Competition Name: racing
Rank: 2
------------------------------
```

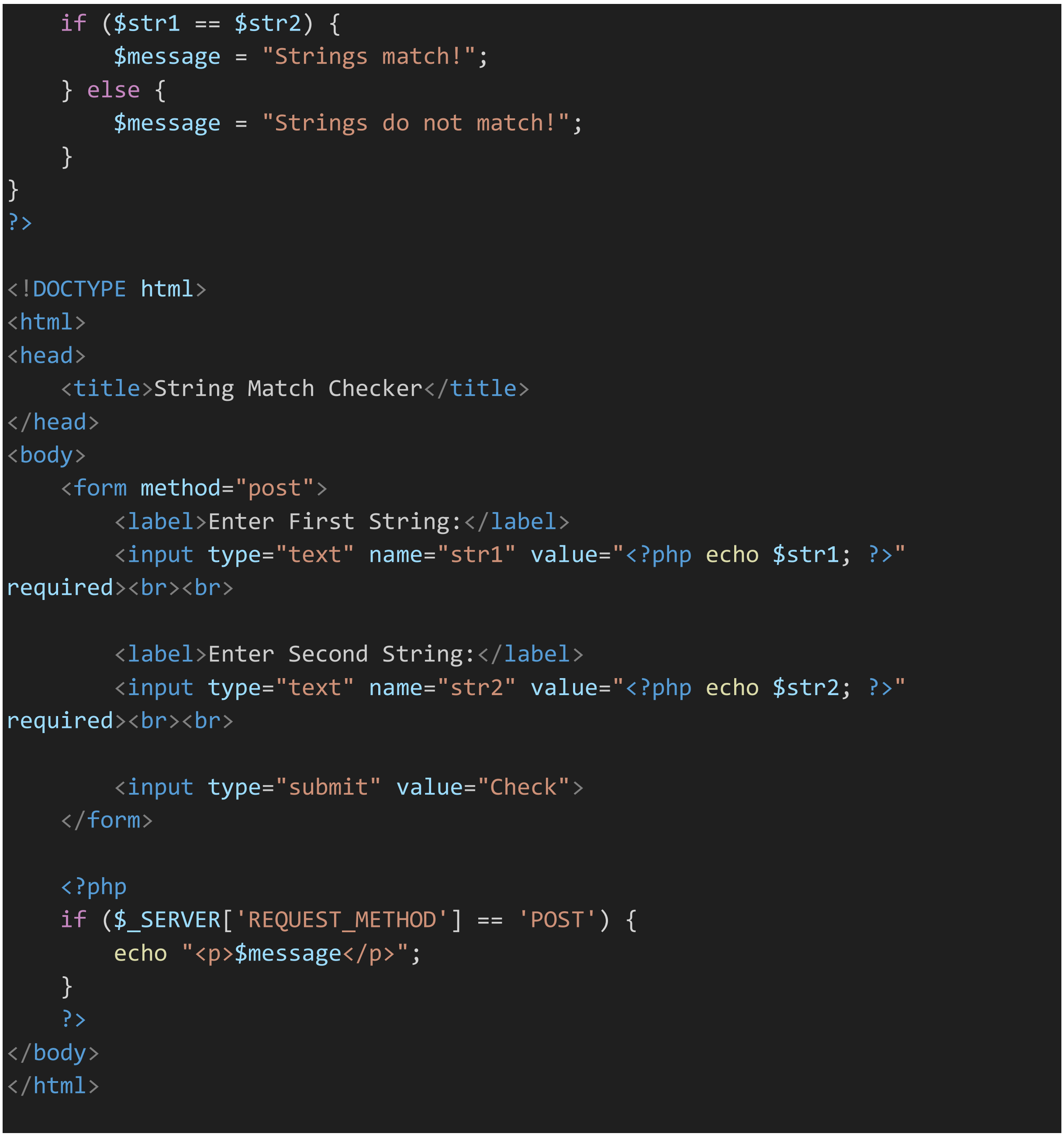
Pehle Details Insert karn

After Sorting as per rank

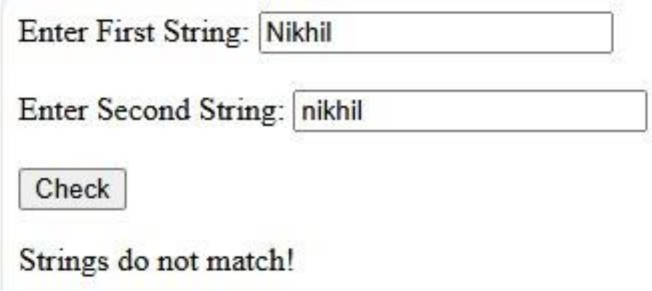
Q. 2)

a) Write a PHP program to accept two string from user and check whether entered strings are matching or not. (Use sticky form concept) [Marks 15]

```
<?php
$str1 = $_POST['str1'] ?? '';
$str2 = $_POST['str2'] ?? '';
$message = '';

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
```



```
    if ($str1 == $str2) {
        $message = "Strings match!";
    } else {
        $message = "Strings do not match!";
    }
}
?>

<!DOCTYPE html>
<html>
<head>
    <title>String Match Checker</title>
</head>
<body>
    <form method="post">
        <label>Enter First String:</label>
        <input type="text" name="str1" value="<?php echo $str1; ?>"
required><br><br>

        <label>Enter Second String:</label>
        <input type="text" name="str2" value="<?php echo $str2; ?>"
required><br><br>

        <input type="submit" value="Check">
    </form>

    <?php
    if ($_SERVER['REQUEST_METHOD'] == 'POST') {
        echo "<p>$message</p>";
    }
    ?>
</body>
</html>
```



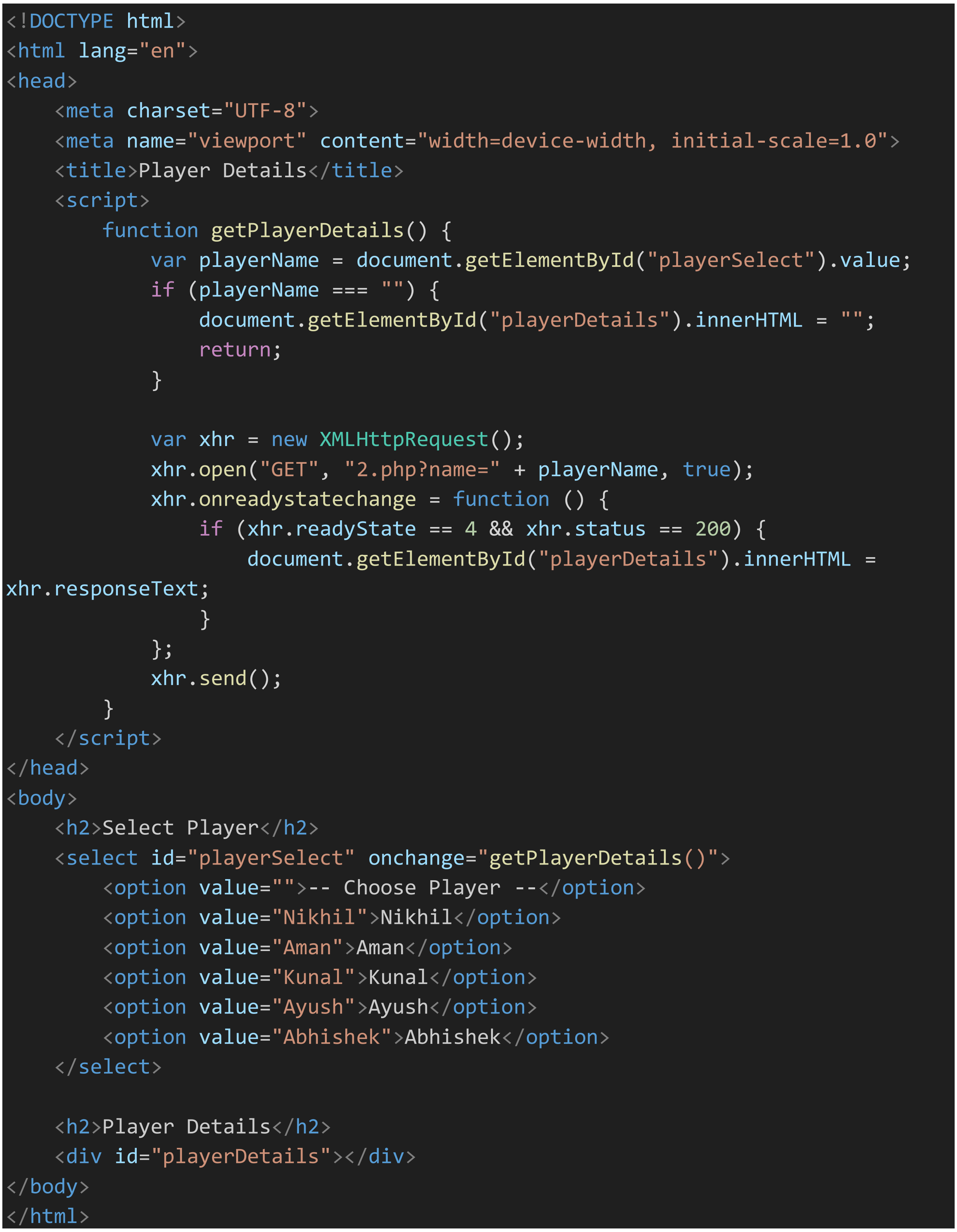
Enter First String: Nikhil

Enter Second String: nikhil

Strings do not match!

b) Write an Ajax script to get player details from XML file when user select player name. Create XML file to store details of player (name, country, wickets and runs). [Marks 25]

HTML / AJAX File :-



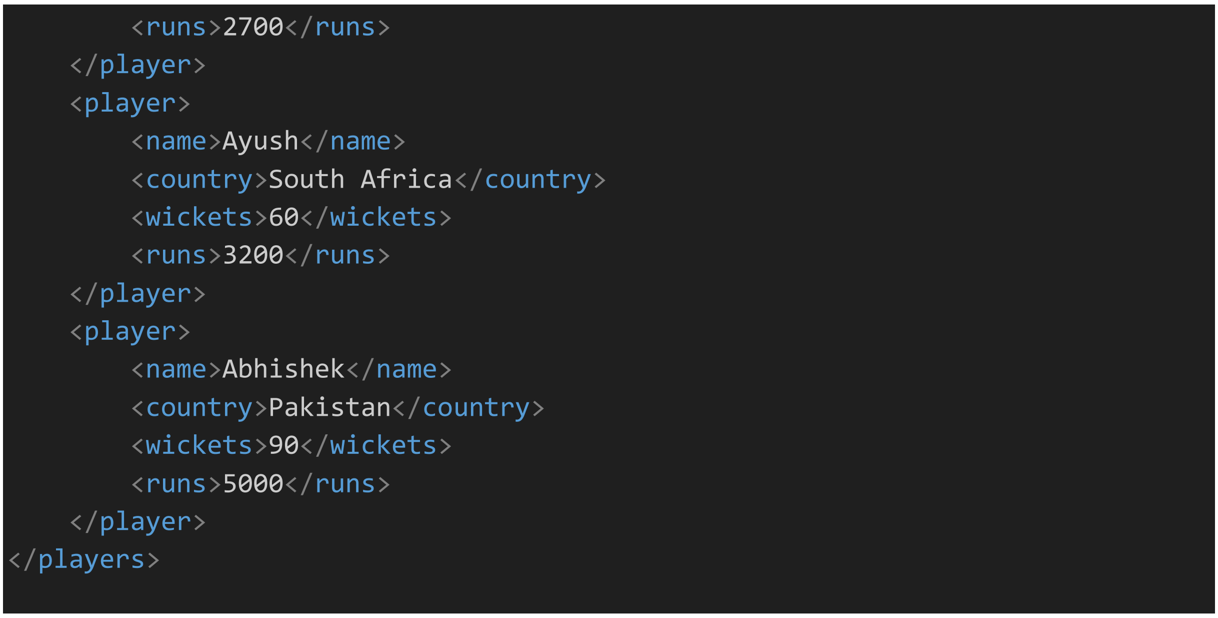
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Player Details</title>
    <script>
        function getPlayerDetails() {
            var playerName = document.getElementById("playerSelect").value;
            if (playerName === "") {
                document.getElementById("playerDetails").innerHTML = "";
                return;
            }

            var xhr = new XMLHttpRequest();
            xhr.open("GET", "2.php?name=" + playerName, true);
            xhr.onreadystatechange = function () {
                if (xhr.readyState == 4 && xhr.status == 200) {
                    document.getElementById("playerDetails").innerHTML =
xhr.responseText;
                }
            };
            xhr.send();
        }
    </script>
</head>
<body>
    <h2>Select Player</h2>
    <select id="playerSelect" onchange="getPlayerDetails()">
        <option value="">-- Choose Player --</option>
        <option value="Nikhil">Nikhil</option>
        <option value="Aman">Aman</option>
        <option value="Kunal">Kunal</option>
        <option value="Ayush">Ayush</option>
        <option value="Abhishek">Abhishek</option>
    </select>

    <h2>Player Details</h2>
    <div id="playerDetails"></div>
</body>
</html>
```

PHP File :-

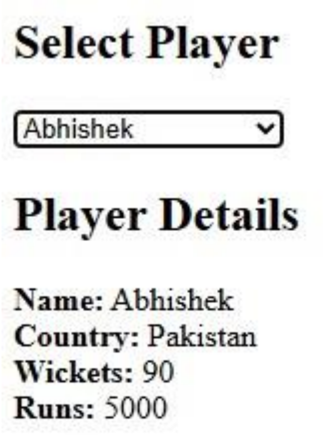
```php
<?php
if (isset($_GET['name'])) {
    $name = $_GET['name'];

    // Load XML file
    $xml = simplexml_load_file("2.xml");

    // Search for the player
    foreach ($xml->player as $player) {
        if ($player->name == $name) {
            echo "<strong>Name:</strong> " . $player->name . "<br>";
            echo "<strong>Country:</strong> " . $player->country . "<br>";
            echo "<strong>Wickets:</strong> " . $player->wickets . "<br>";
            echo "<strong>Runs:</strong> " . $player->runs . "<br>";
            exit;
        }
    }
    echo "Player not found!";
}
?>
```

XML File:-

```xml
<?xml version="1.0" encoding="UTF-8"?>
<players>
    <player>
        <name>Nikhil</name>
        <country>India</country>
        <wickets>50</wickets>
        <runs>3000</runs>
    </player>
    <player>
        <name>Aman</name>
        <country>Australia</country>
        <wickets>80</wickets>
        <runs>4500</runs>
    </player>
    <player>
        <name>Kunal</name>
        <country>England</country>
        <wickets>40</wickets>
```



```
        <runs>2700</runs>
    </player>
    <player>
        <name>Ayush</name>
        <country>South Africa</country>
        <wickets>60</wickets>
        <runs>3200</runs>
    </player>
    <player>
        <name>Abhishek</name>
        <country>Pakistan</country>
        <wickets>90</wickets>
        <runs>5000</runs>
    </player>
</players>
```



Select Player

Abhishek

Player Details

Name: Abhishek
Country: Pakistan
Wickets: 90
Runs: 5000

Q. 3) Lab Book **[Marks 10]**

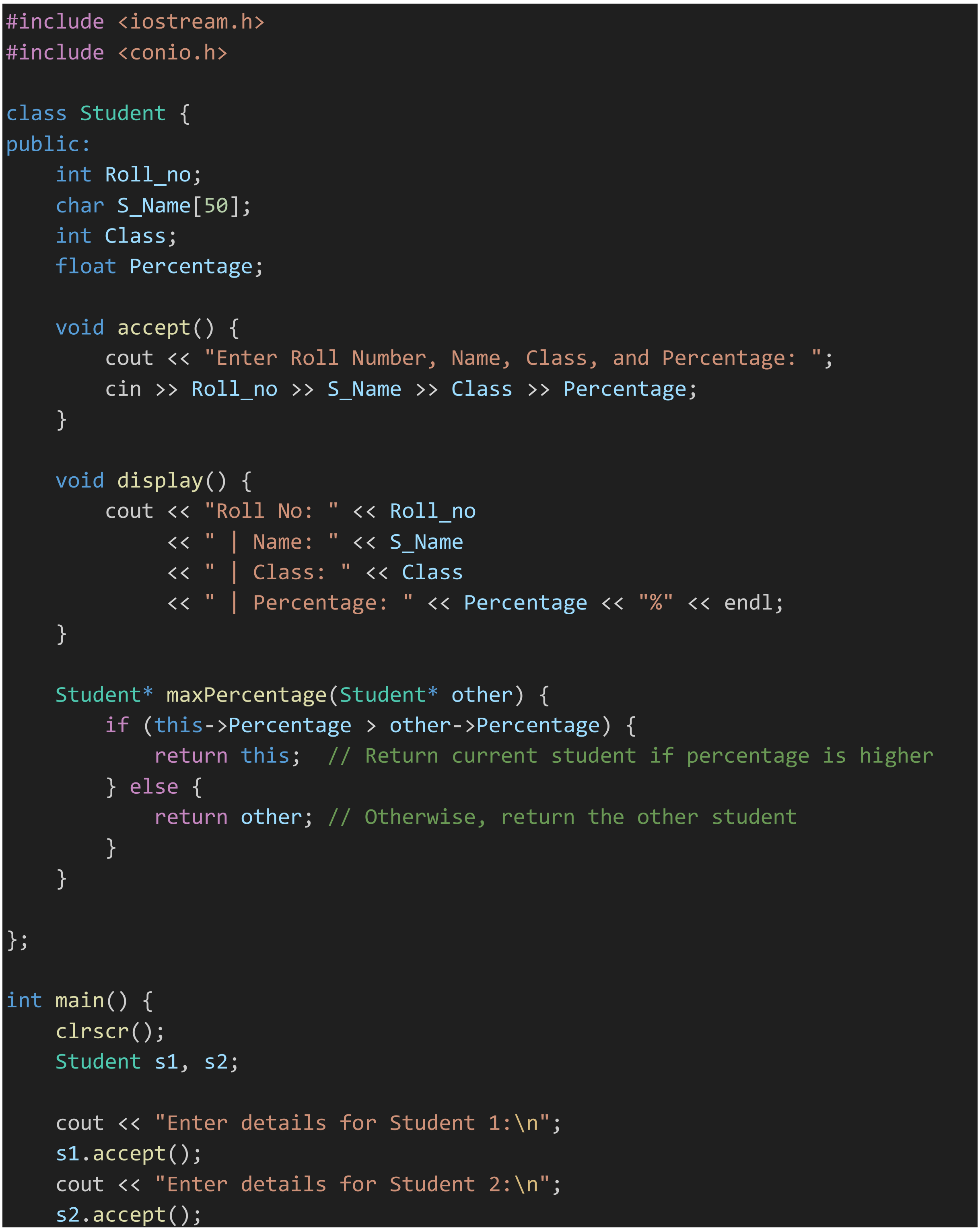
Q. 4) Viva **[Marks 10]**

Slip -5

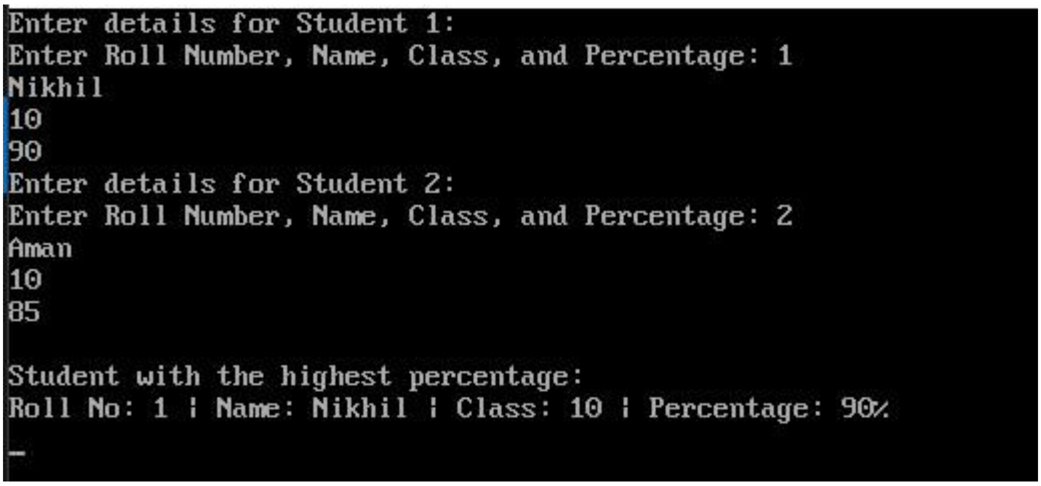
Q. 1)

a) Create a C++ Class Student with data members Rollno, Sname, Class, Percentage. Accept two students information and display information of students having maximum percentage. (Use this pointer.

[Marks 15]



```cpp
#include <iostream.h>
#include <conio.h>

class Student {
public:
    int Roll_no;
    char S_Name[50];
    int Class;
    float Percentage;

    void accept() {
        cout << "Enter Roll Number, Name, Class, and Percentage: ";
        cin >> Roll_no >> S_Name >> Class >> Percentage;
    }

    void display() {
        cout << "Roll No: " << Roll_no
             << " | Name: " << S_Name
             << " | Class: " << Class
             << " | Percentage: " << Percentage << "%" << endl;
    }

    Student* maxPercentage(Student* other) {
        if (this->Percentage > other->Percentage) {
            return this;  // Return current student if percentage is higher
        } else {
            return other; // Otherwise, return the other student
        }
    }

};

int main() {
    clrscr();
    Student s1, s2;

    cout << "Enter details for Student 1:\n";
    s1.accept();
    cout << "Enter details for Student 2:\n";
    s2.accept();
    cout << "\nStudent with the highest percentage:\n";
    s1.maxPercentage(&s2)->display();

    getch();
    return 0;
}
```

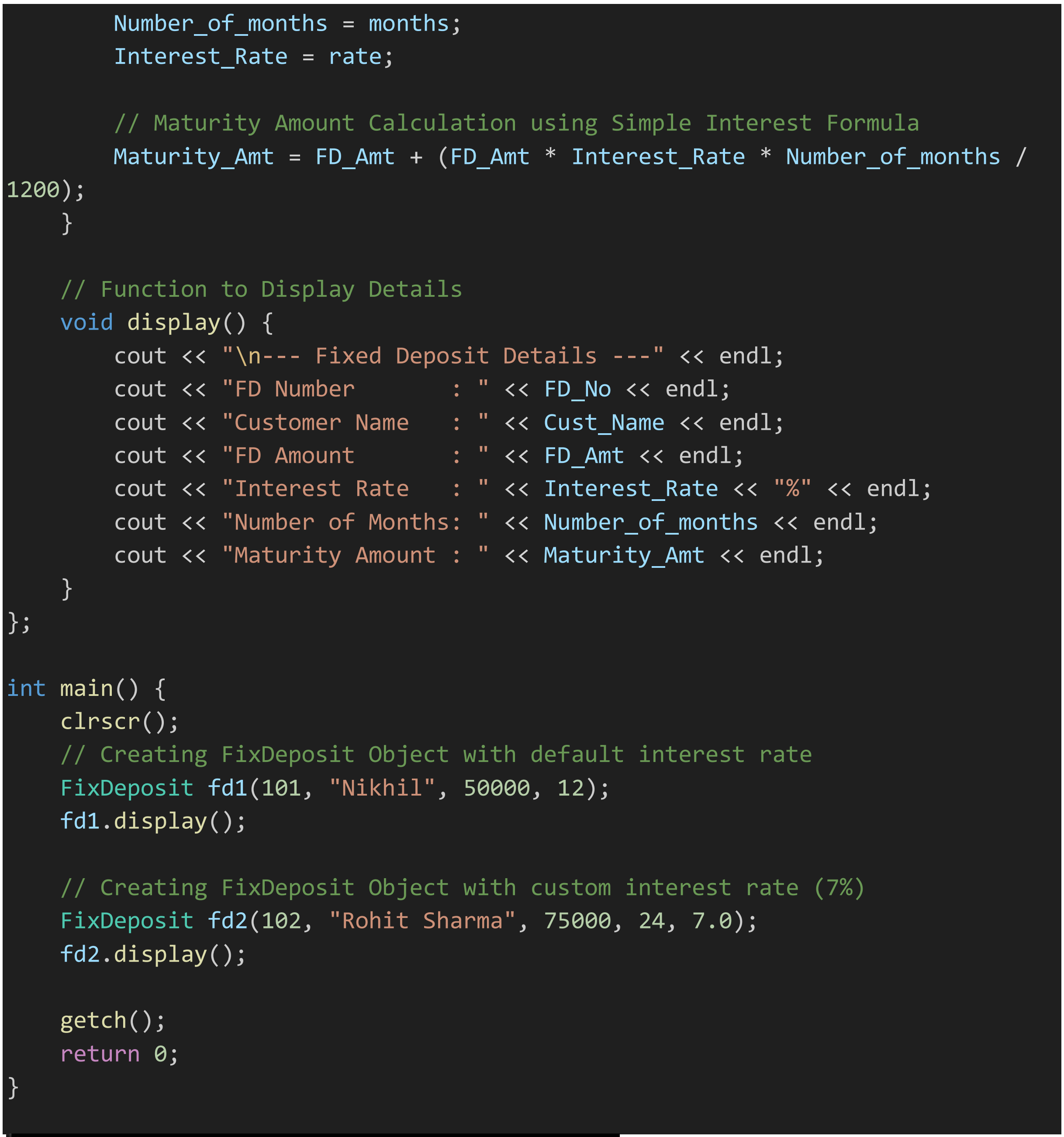


```
Enter details for Student 1:
Enter Roll Number, Name, Class, and Percentage: 1
Nikhil
10
90
Enter details for Student 2:
Enter Roll Number, Name, Class, and Percentage: 2
Aman
10
85

Student with the highest percentage:
Roll No: 1 | Name: Nikhil | Class: 10 | Percentage: 90%
_
```

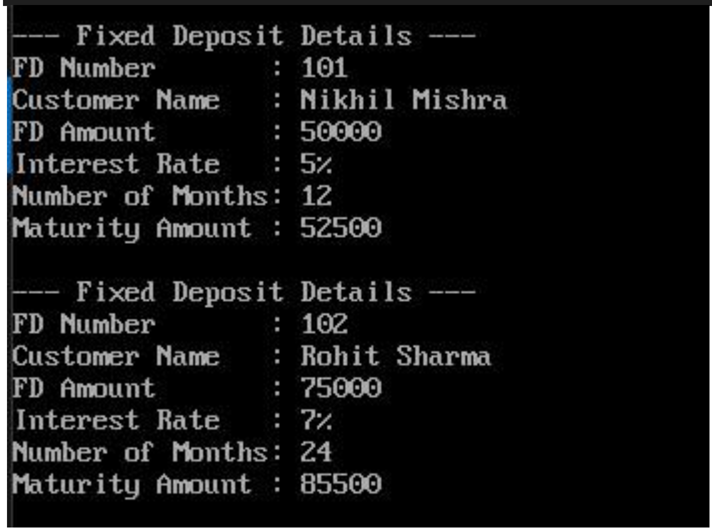
b) Create a C++ Class FixDeposite with data members FD_No, Cust_Name, FD_Amt, Interest rate, Maturity_Amt, No_Of_Months. Create and initialize all values of FixDeposit object by using parameterized constructor default value for interest rate. Calculate maturity amt using interest rate and display all the details.

[Marks 25]

```
#include <iostream.h>
#include <conio.h>
#include <string.h>

class FixDeposit {
private:
    int FD_No;
    char Cust_Name[50];
    double FD_Amt;
    double Interest_Rate;
    double Maturity_Amt;
    int Number_of_months;

public:
    // Parameterized Constructor with Default Interest Rate = 5%
    FixDeposit(int fd_no, char name[], double amt, int months, double rate =
5.0) {
        FD_No = fd_no;
        strcpy(Cust_Name,name);
        FD_Amt = amt;
```



```
        Number_of_months = months;
        Interest_Rate = rate;

        // Maturity Amount Calculation using Simple Interest Formula
        Maturity_Amt = FD_Amt + (FD_Amt * Interest_Rate * Number_of_months /
1200);
    }

    // Function to Display Details
    void display() {
        cout << "\n--- Fixed Deposit Details ---" << endl;
        cout << "FD Number        : " << FD_No << endl;
        cout << "Customer Name    : " << Cust_Name << endl;
        cout << "FD Amount        : " << FD_Amt << endl;
        cout << "Interest Rate    : " << Interest_Rate << "%" << endl;
        cout << "Number of Months: " << Number_of_months << endl;
        cout << "Maturity Amount : " << Maturity_Amt << endl;
    }
};

int main() {
    clrscr();
    // Creating FixDeposit Object with default interest rate
    FixDeposit fd1(101, "Nikhil", 50000, 12);
    fd1.display();

    // Creating FixDeposit Object with custom interest rate (7%)
    FixDeposit fd2(102, "Rohit Sharma", 75000, 24, 7.0);
    fd2.display();

    getch();
    return 0;
}
```

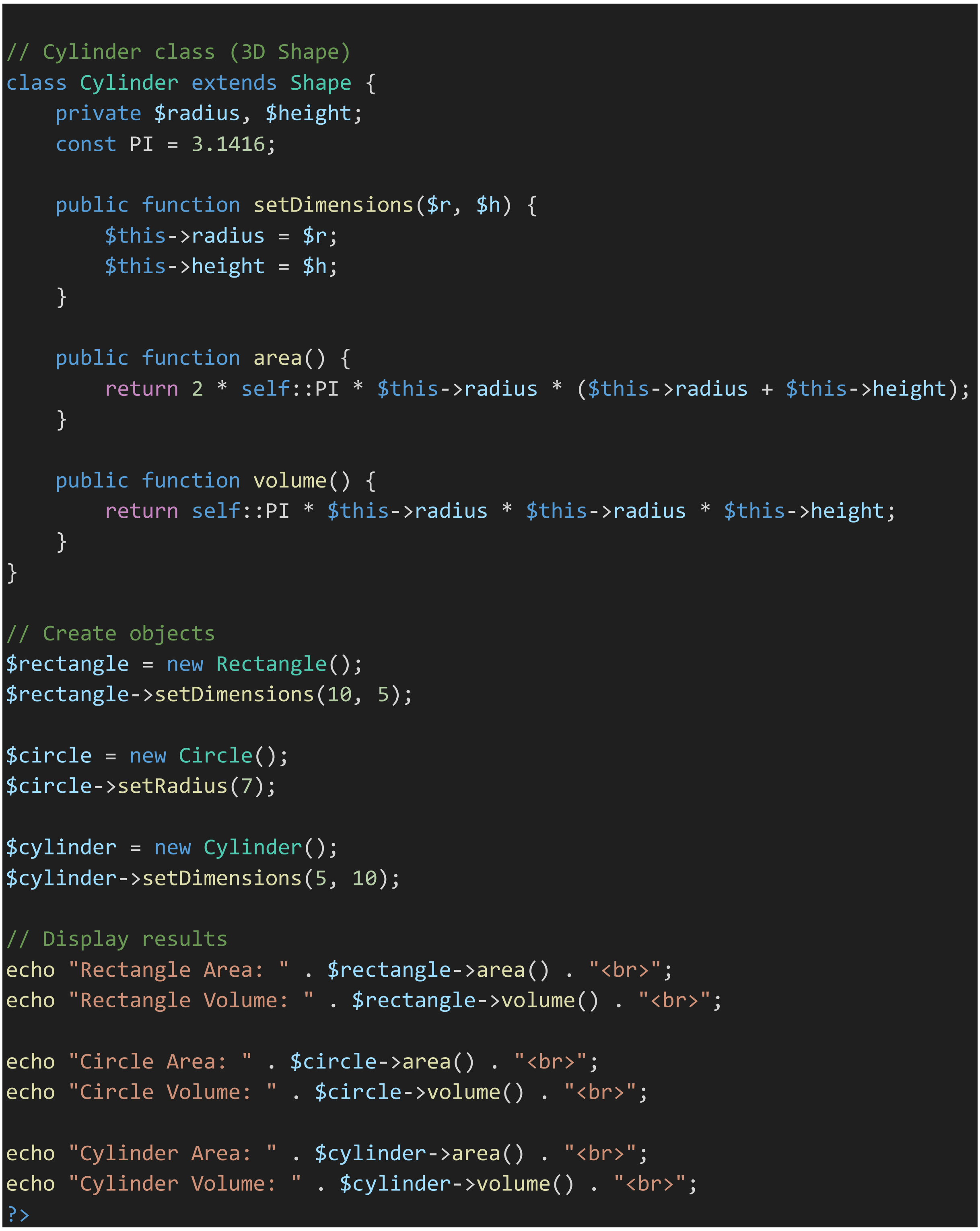


```
--- Fixed Deposit Details ---
FD Number         : 101
Customer Name     : Nikhil Mishra
FD Amount         : 50000
Interest Rate     : 5%
Number of Months: 12
Maturity Amount : 52500

--- Fixed Deposit Details ---
FD Number         : 102
Customer Name     : Rohit Sharma
FD Amount         : 75000
Interest Rate     : 7%
Number of Months: 24
Maturity Amount : 85500
```

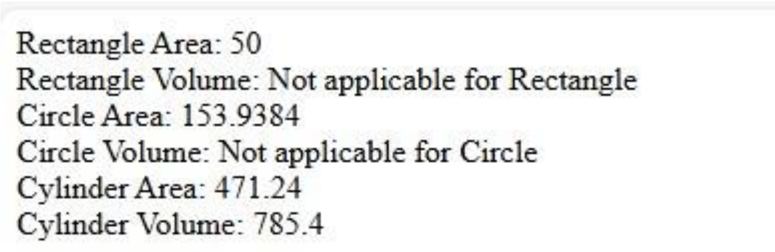
Q. 2)

a) Create an abstract class Shape with methods area( ) and volume( ). Derive three classes rectangle (length, breath), Circle(radius) and Cylinder(radius, height), Calculate area and volume
of all. (Use Method overriding). [Marks l5]

```php
<?php
// Abstract class Shape
abstract class Shape {
    abstract public function area();
    abstract public function volume();
}

// Rectangle class (2D Shape, volume is not applicable)
class Rectangle extends Shape {
    private $length, $breadth;

    public function setDimensions($l, $b) {
        $this->length = $l;
        $this->breadth = $b;
    }

    public function area() {
        return $this->length * $this->breadth;
    }

    public function volume() {
        return "Not applicable for Rectangle";
    }
}

// Circle class (2D Shape, volume is not applicable)
class Circle extends Shape {
    private $radius;
    const PI = 3.1416;

    public function setRadius($r) {
        $this->radius = $r;
    }

    public function area() {
        return self::PI * $this->radius * $this->radius;
    }

    public function volume() {
        return "Not applicable for Circle";
    }
}
```



```php
// Cylinder class (3D Shape)
class Cylinder extends Shape {
    private $radius, $height;
    const PI = 3.1416;

    public function setDimensions($r, $h) {
        $this->radius = $r;
        $this->height = $h;
    }

    public function area() {
        return 2 * self::PI * $this->radius * ($this->radius + $this->height);
    }

    public function volume() {
        return self::PI * $this->radius * $this->radius * $this->height;
    }
}

// Create objects
$rectangle = new Rectangle();
$rectangle->setDimensions(10, 5);

$circle = new Circle();
$circle->setRadius(7);

$cylinder = new Cylinder();
$cylinder->setDimensions(5, 10);

// Display results
echo "Rectangle Area: " . $rectangle->area() . "<br>";
echo "Rectangle Volume: " . $rectangle->volume() . "<br>";

echo "Circle Area: " . $circle->area() . "<br>";
echo "Circle Volume: " . $circle->volume() . "<br>";

echo "Cylinder Area: " . $cylinder->area() . "<br>";
echo "Cylinder Volume: " . $cylinder->volume() . "<br>";
?>
```



```
Rectangle Area: 50
Rectangle Volume: Not applicable for Rectangle
Circle Area: 153.9384
Circle Volume: Not applicable for Circle
Cylinder Area: 471.24
Cylinder Volume: 785.4
```

b) Create student table as follows: Student(sno, sname, standard, Marks, per).
Write AJAX script to select the student name and print the student's details of particular standard. [Marks 25]

HTML / AJAX :-

```
<html>
<head>
    <script>
        function showadd(str) {
            if (str == "") {
                document.getElementById("myDiv").innerHTML = "";
                return;
            }

            var XHRobj = new XMLHttpRequest();
            XHRobj.onreadystatechange = function () {
                if (XHRobj.readyState == 4 && XHRobj.status == 200) {
                    document.getElementById("myDiv").innerHTML =
XHRobj.responseText;
                }
            };

            XHRobj.open("GET", "2.php?q=" + encodeURIComponent(str), true);
            XHRobj.send();
        }
    </script>
</head>
<body>
    <form>
        Enter Student Name: <input type="text" name="n1" id="studentName"><br>
        <input type="button" value="Details"
onclick="showadd(document.getElementById('studentName').value)">
    </form>
    <div id="myDiv"></div>
</body>
</html>
```

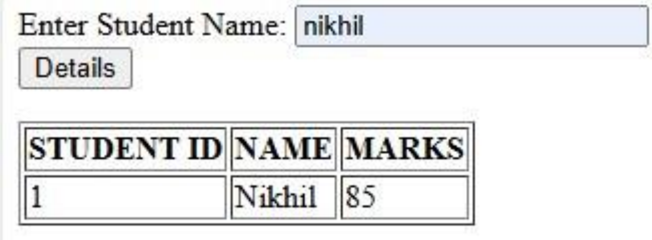
PHP File:-

```
<?php
$q = $_GET["q"];

$conn = mysqli_connect("localhost", "root", "", "studdb") or
die(mysqli_error($conn));

$sql = "SELECT * FROM student WHERE sname='" . $q . "'";
$result = mysqli_query($conn, $sql);
?>

<table border='1'>
<tr>
    <th>STUDENT ID</th>
    <th>NAME</th>
    <th>MARKS</th>
</tr>

<?php
while ($r = mysqli_fetch_array($result)) {
    echo "<tr>";
    echo "<td>" . $r['sno'] . "</td>";
    echo "<td>" . $r['sname'] . "</td>";
    echo "<td>" . $r['marks'] . "</td>";
    echo "</tr>";
}
?>
</table>

<?php
mysqli_close($conn);
?>
```

SQL Query:-

```
CREATE DATABASE studdb;

USE studdb;

CREATE TABLE student (
   sno INT PRIMARY KEY AUTO_INCREMENT,
   sname VARCHAR(100) NOT NULL,
   standard VARCHAR(20) NOT NULL,
   marks INT NOT NULL,
   per FLOAT NOT NULL
);
```

```
INSERT INTO student (sname, standard, marks, per)
VALUES
('Nikhil', '10th', 85, 85),
('Aman', '9th', 78, 78),
('Kunal', '10th', 90, 90),
('Ayush', '8th', 88, 88),
('Abhishek', '9th', 80, 80);
```



Enter Student Name: nikhil

STUDENT ID	NAME	MARKS
1	Nikhil	85

Q. 3) Lab Book **[Marks 10]**

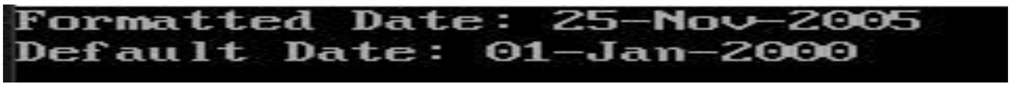
Q. 4) Viva **[Marks 10]**

Slip -6

Q. 1)

a) Write a C++ program to create a class date with data members day, month and year. Use default and parameterized constructor to initialize date and display date in dd-mon-yyyy format (Example: input: 04-01-2021 Output : 04-jan-2021)
[Marks 15]

```
#include <iostream.h>
#include <conio.h>

// Define month names globally
const char* months[12] = {"Jan", "Feb", "Mar", "Apr", "May", "Jun",
                          "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"};

class Date {
private:
    int day, month, year;

public:
    // Default constructor
    Date() {
        day = 1;
        month = 1;
        year = 2000;
    }

    // Parameterized constructor
    Date(int d, int m, int y) {
        day = d;
        month = m;
        year = y;
    }

    // Function to display date in dd-Mon-yyyy format
    void displayDate() {
        cout << (day < 10 ? "0" : "") << day << "-" << months[month - 1] << "-
" << year << endl;
    }
};

int main() {
    clrscr();

    // Using parameterized constructor
    Date d1(25, 11, 2005);
    cout << "Formatted Date: ";
    d1.displayDate();
    // Using default constructor
    Date d2;
    cout << "Default Date: ";
    d2.displayDate();

    getch();
    return 0;
}
```

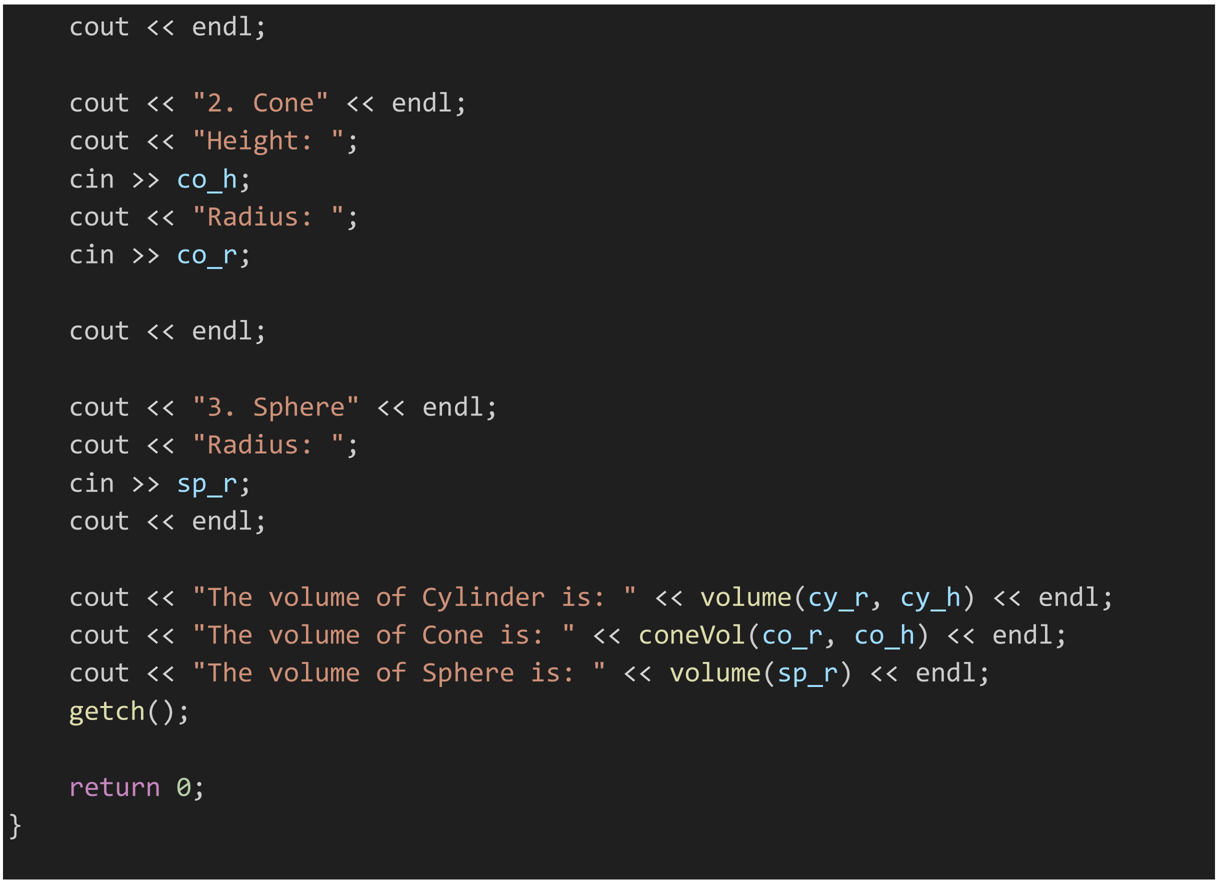


```
Formatted Date: 25-Nov-2005
Default Date: 01-Jan-2000
```

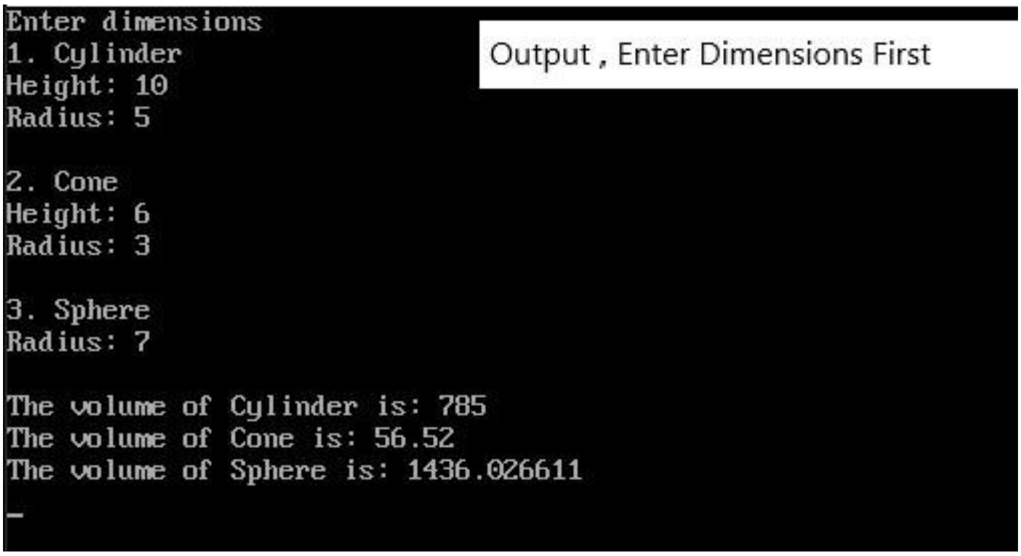
b) Write a C++ Program to find volume of cylinder, cone and sphere. (Use function overloading)

[Marks 25]

```
#include <iostream.h>
#include <conio.h>


float volume(float r, float h) // Cylinder Volume
{
    return (3.14 * r * r * h);
}

float coneVol(float r, float h) // Cone Volume
{
    return (3.14 * r * r * h / 3);
}

float volume(float r) // Sphere Volume
{
    return (4.0 / 3.0 * 3.14 * r * r * r);
}

int main() {
    clrscr();
    float cy_h, cy_r, co_h, co_r, sp_r;
    cout << "Enter dimensions" << endl;

    cout << "1. Cylinder" << endl;
    cout << "Height: ";
    cin >> cy_h;
    cout << "Radius: ";
    cin >> cy_r;
```



```
    cout << endl;

    cout << "2. Cone" << endl;
    cout << "Height: ";
    cin >> co_h;
    cout << "Radius: ";
    cin >> co_r;

    cout << endl;

    cout << "3. Sphere" << endl;
    cout << "Radius: ";
    cin >> sp_r;
    cout << endl;

    cout << "The volume of Cylinder is: " << volume(cy_r, cy_h) << endl;
    cout << "The volume of Cone is: " << coneVol(co_r, co_h) << endl;
    cout << "The volume of Sphere is: " << volume(sp_r) << endl;
    getch();

    return 0;
}
```



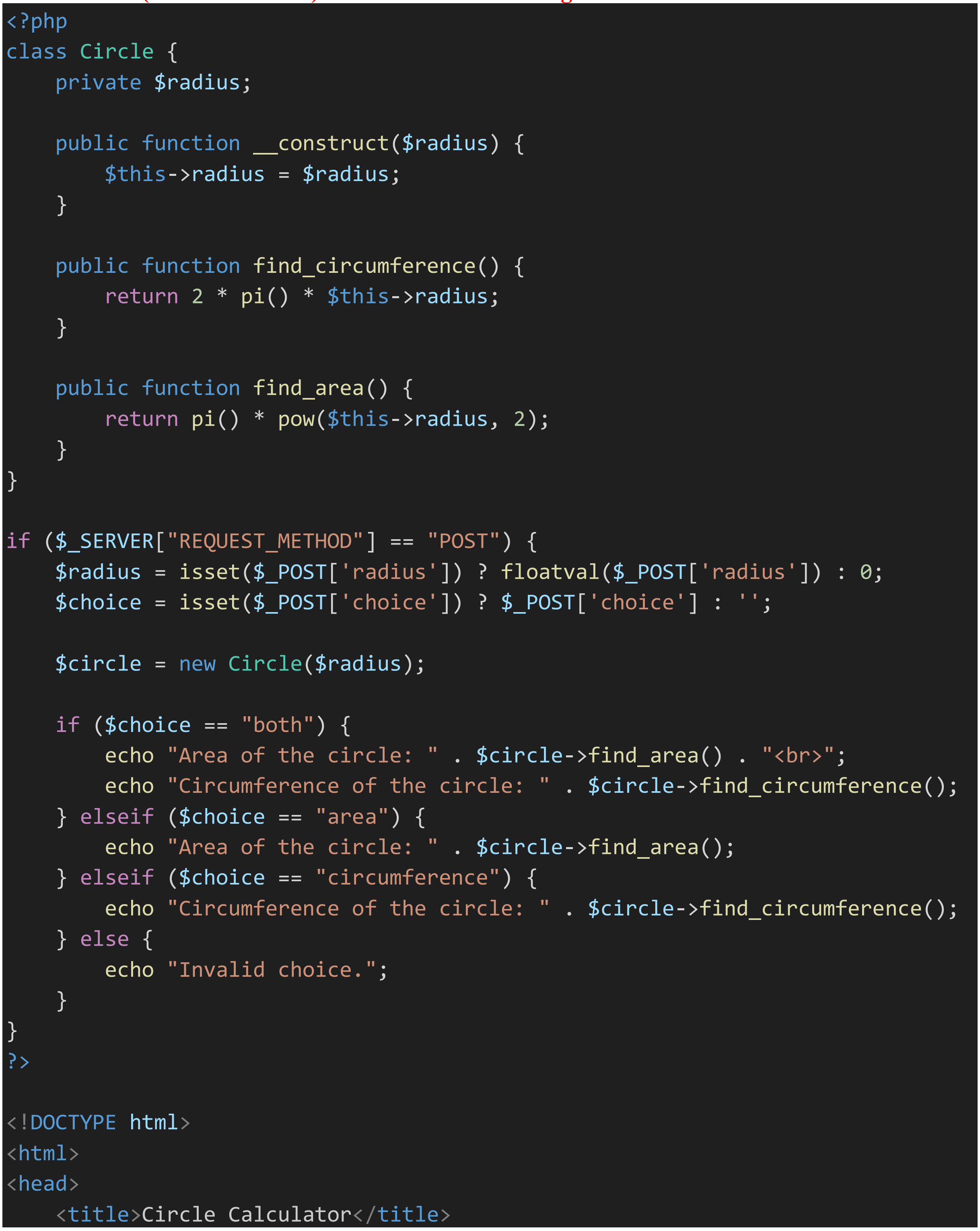
Output , Enter Dimensions First

```
Enter dimensions
1. Cylinder
Height: 10
Radius: 5

2. Cone
Height: 6
Radius: 3

3. Sphere
Radius: 7

The volume of Cylinder is: 785
The volume of Cone is: 56.52
The volume of Sphere is: 1436.026611
_
```

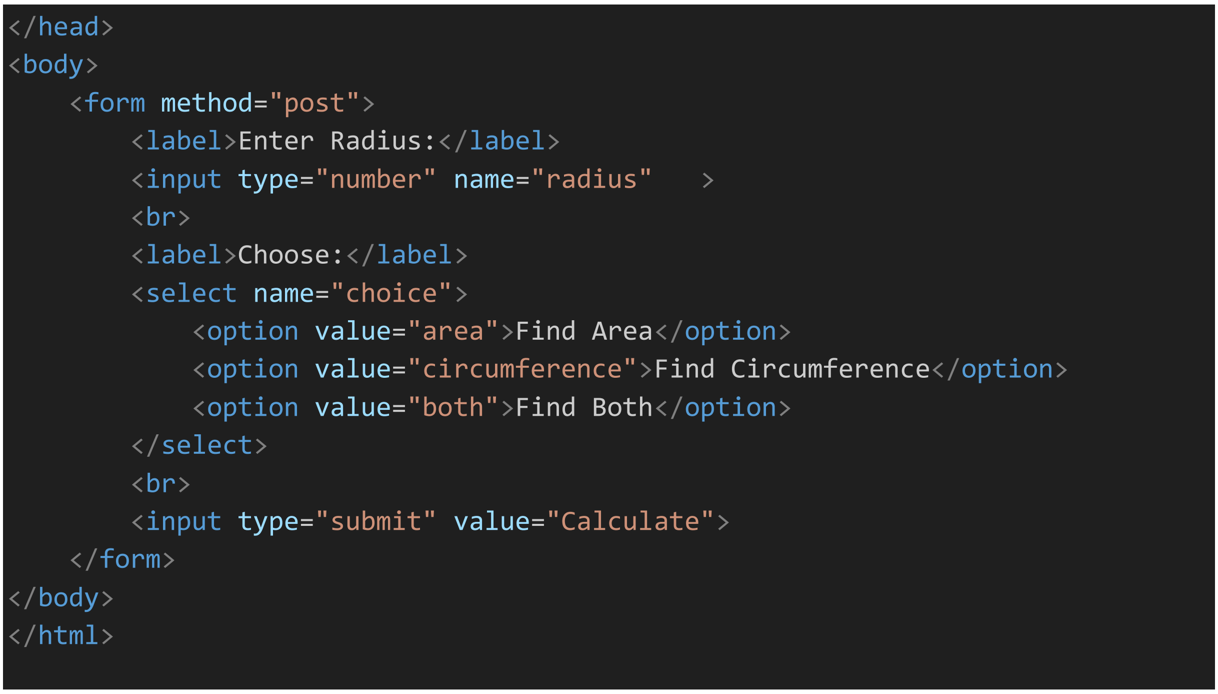
Q. 2)

a) Write a PHP program to create class circle having radius data member and two member functions find circumference() and find area(). Display area and circumference depending on user's
preference. [Marks l5]

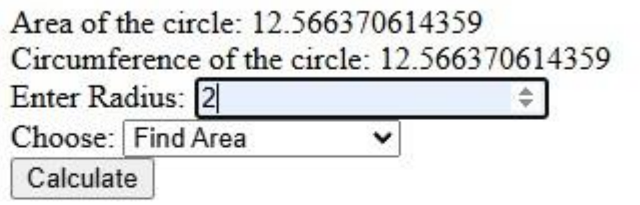
Both the File (HTML & PHP) should be written in single file ……..



```php
<?php
class Circle {
    private $radius;

    public function __construct($radius) {
        $this->radius = $radius;
    }

    public function find_circumference() {
        return 2 * pi() * $this->radius;
    }

    public function find_area() {
        return pi() * pow($this->radius, 2);
    }
}

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $radius = isset($_POST['radius']) ? floatval($_POST['radius']) : 0;
    $choice = isset($_POST['choice']) ? $_POST['choice'] : '';

    $circle = new Circle($radius);

    if ($choice == "both") {
        echo "Area of the circle: " . $circle->find_area() . "<br>";
        echo "Circumference of the circle: " . $circle->find_circumference();
    } elseif ($choice == "area") {
        echo "Area of the circle: " . $circle->find_area();
    } elseif ($choice == "circumference") {
        echo "Circumference of the circle: " . $circle->find_circumference();
    } else {
        echo "Invalid choice.";
    }
}
?>

<!DOCTYPE html>
<html>
<head>
    <title>Circle Calculator</title>
```

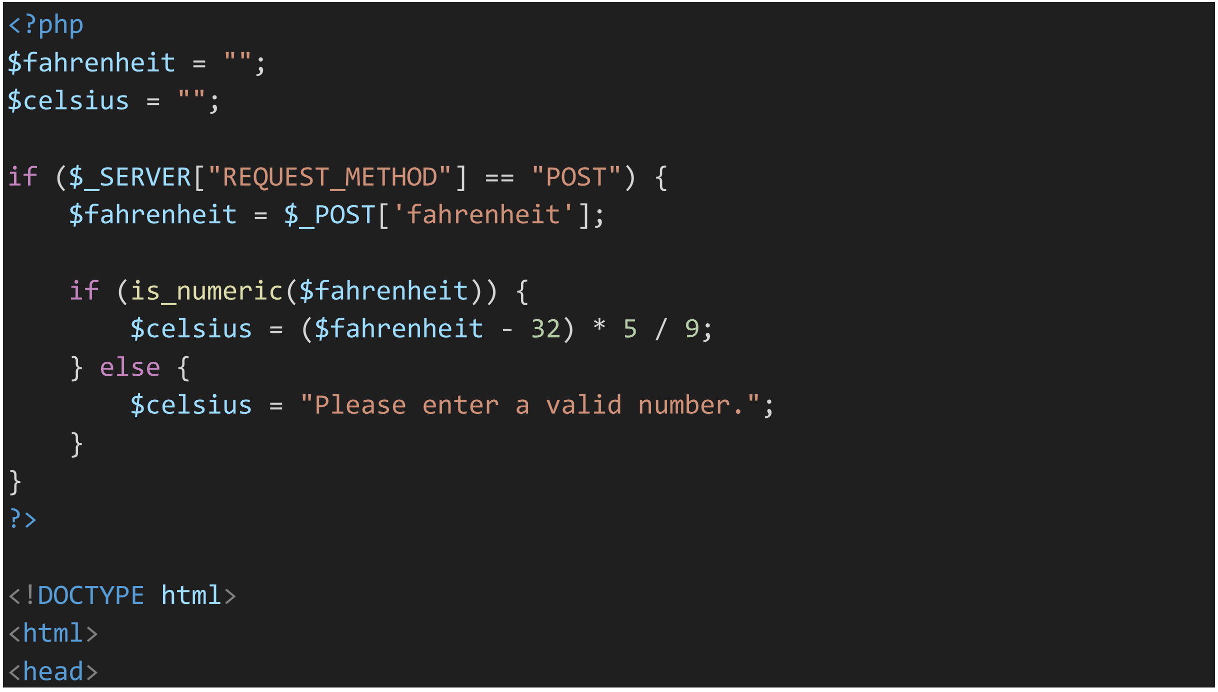


```
</head>
<body>
    <form method="post">
        <label>Enter Radius:</label>
        <input type="number" name="radius"    >
        <br>
        <label>Choose:</label>
        <select name="choice">
            <option value="area">Find Area</option>
            <option value="circumference">Find Circumference</option>
            <option value="both">Find Both</option>
        </select>
        <br>
        <input type="submit" value="Calculate">
    </form>
</body>
</html>
```

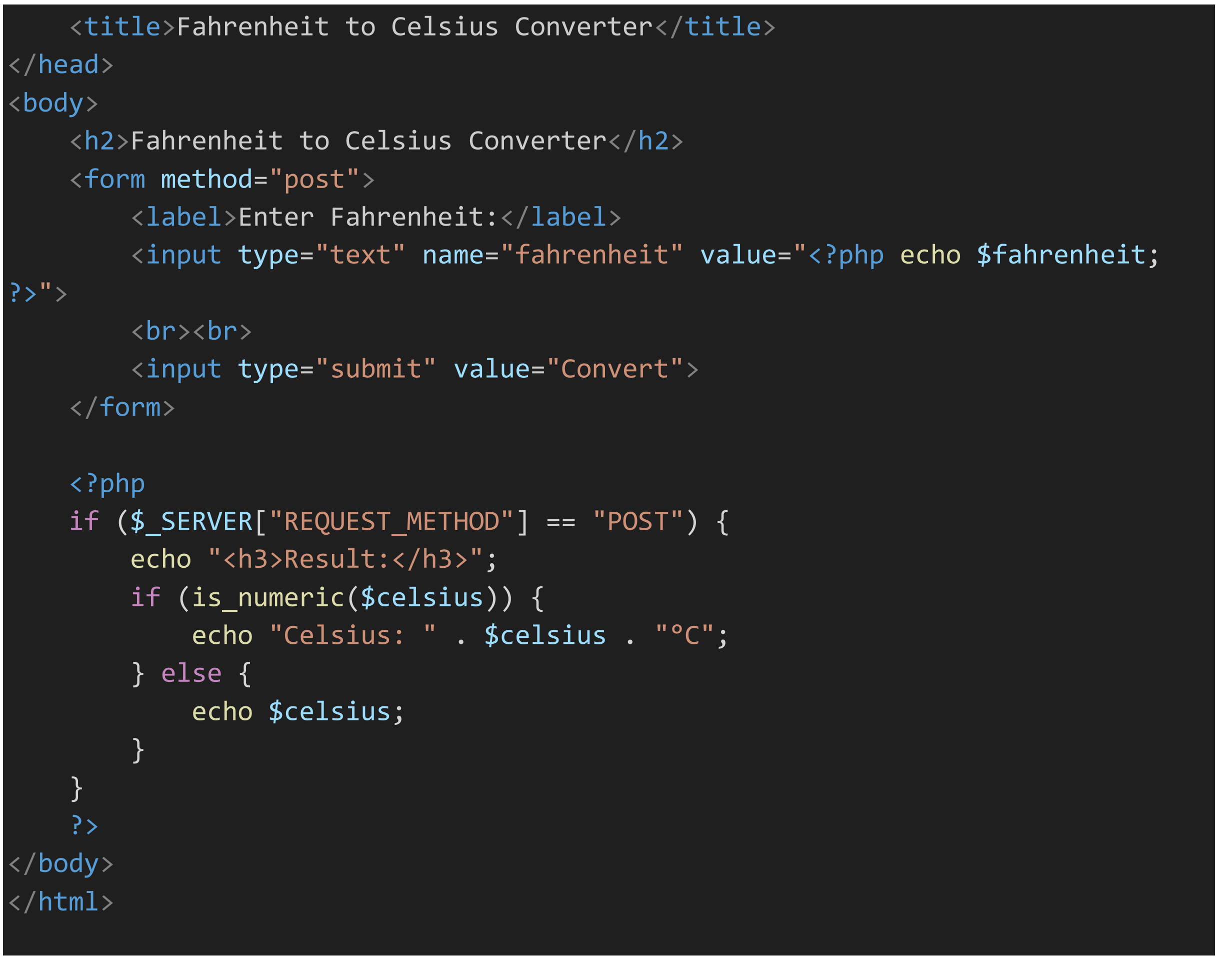


```
Area of the circle: 12.566370614359
Circumference of the circle: 12.566370614359
Enter Radius: 2
Choose: Find Area
```

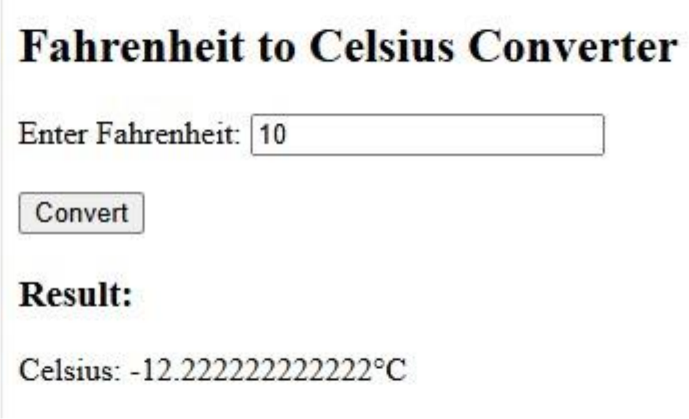
b) Write a PHP program to convert temperature Fahrenheit to Celsius using sticky form. [Marks 25]



```
<?php
$fahrenheit = "";
$celsius = "";

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $fahrenheit = $_POST['fahrenheit'];

    if (is_numeric($fahrenheit)) {
        $celsius = ($fahrenheit - 32) * 5 / 9;
    } else {
        $celsius = "Please enter a valid number.";
    }
}
?>

<!DOCTYPE html>
<html>
<head>
```



```
    <title>Fahrenheit to Celsius Converter</title>
</head>
<body>
    <h2>Fahrenheit to Celsius Converter</h2>
    <form method="post">
        <label>Enter Fahrenheit:</label>
        <input type="text" name="fahrenheit" value="<?php echo $fahrenheit;
?>">
        <br><br>
        <input type="submit" value="Convert">
    </form>

    <?php
    if ($_SERVER["REQUEST_METHOD"] == "POST") {
        echo "<h3>Result:</h3>";
        if (is_numeric($celsius)) {
            echo "Celsius: " . $celsius . "°C";
        } else {
            echo $celsius;
        }
    }
    ?>
</body>
</html>
```



Fahrenheit to Celsius Converter

Enter Fahrenheit: 10

Result:

Celsius: -12.222222222222°C

Q. 3) Lab Book **[Marks 10]**

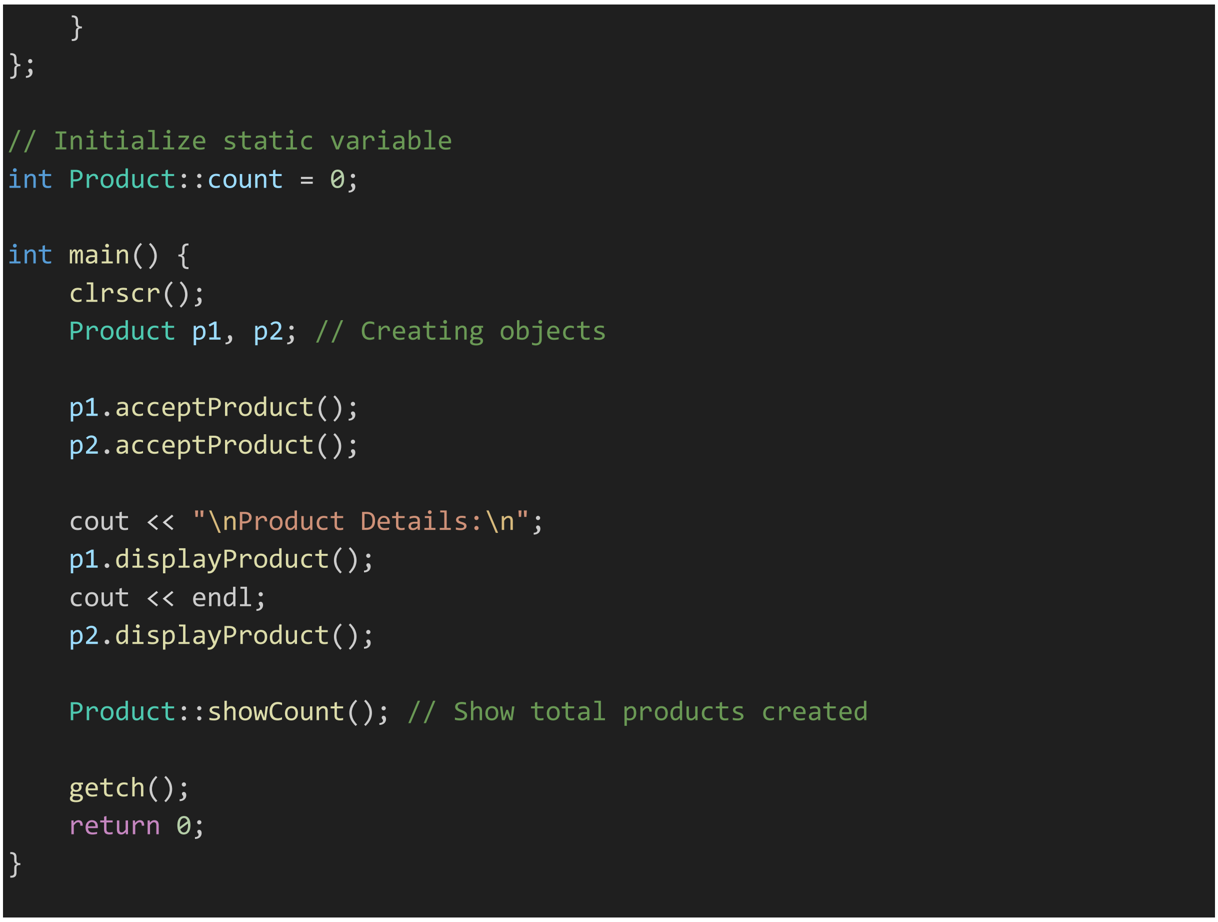
Q. 4) Viva **[Marks 10]**

Slip -7

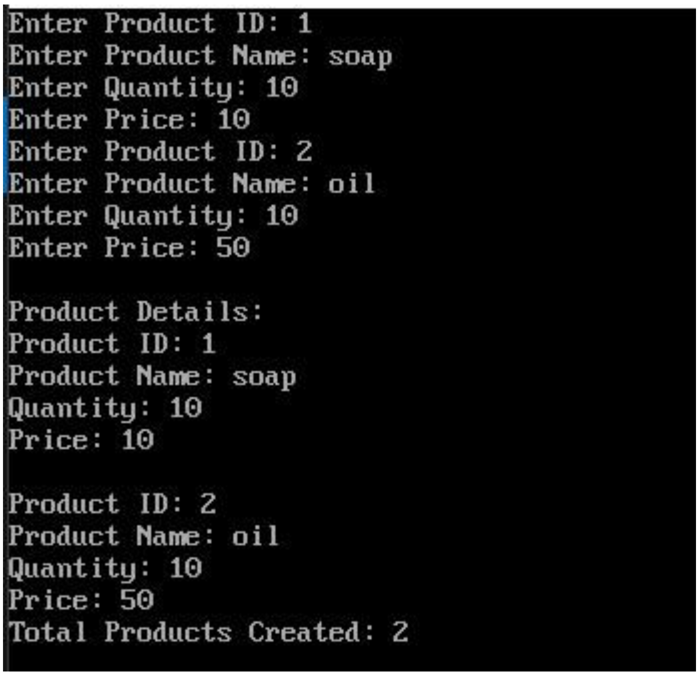
Q. 1)

a) Write a c++ program to create a class product with data members product_id, product_name, qty, price. Write necessary function to accept and display product information also display number of objects created for product class.
[Marks 15]

```
#include <iostream.h>
#include <conio.h>

class Product {
private:
    int product_id;
    char product_name[50];
    int qty;
    float price;
    static int count; // Static variable to count objects

public:
    // Constructor to increment count
    Product() {
        count++;
    }

    // Function to accept product details
    void acceptProduct() {
        cout << "Enter Product ID: ";
        cin >> product_id;
        cout << "Enter Product Name: ";
        cin >> product_name;
        cout << "Enter Quantity: ";
        cin >> qty;
        cout << "Enter Price: ";
        cin >> price;
    }

    // Function to display product details
    void displayProduct() {
        cout << "Product ID: " << product_id << endl;
        cout << "Product Name: " << product_name << endl;
        cout << "Quantity: " << qty << endl;
        cout << "Price: " << price << endl;
    }

    // Static function to show total products created
    static void showCount() {
        cout << "Total Products Created: " << count << endl;
```



```cpp
    }
};

// Initialize static variable
int Product::count = 0;

int main() {
    clrscr();
    Product p1, p2; // Creating objects

    p1.acceptProduct();
    p2.acceptProduct();

    cout << "\nProduct Details:\n";
    p1.displayProduct();
    cout << endl;
    p2.displayProduct();

    Product::showCount(); // Show total products created

    getch();
    return 0;
}
```

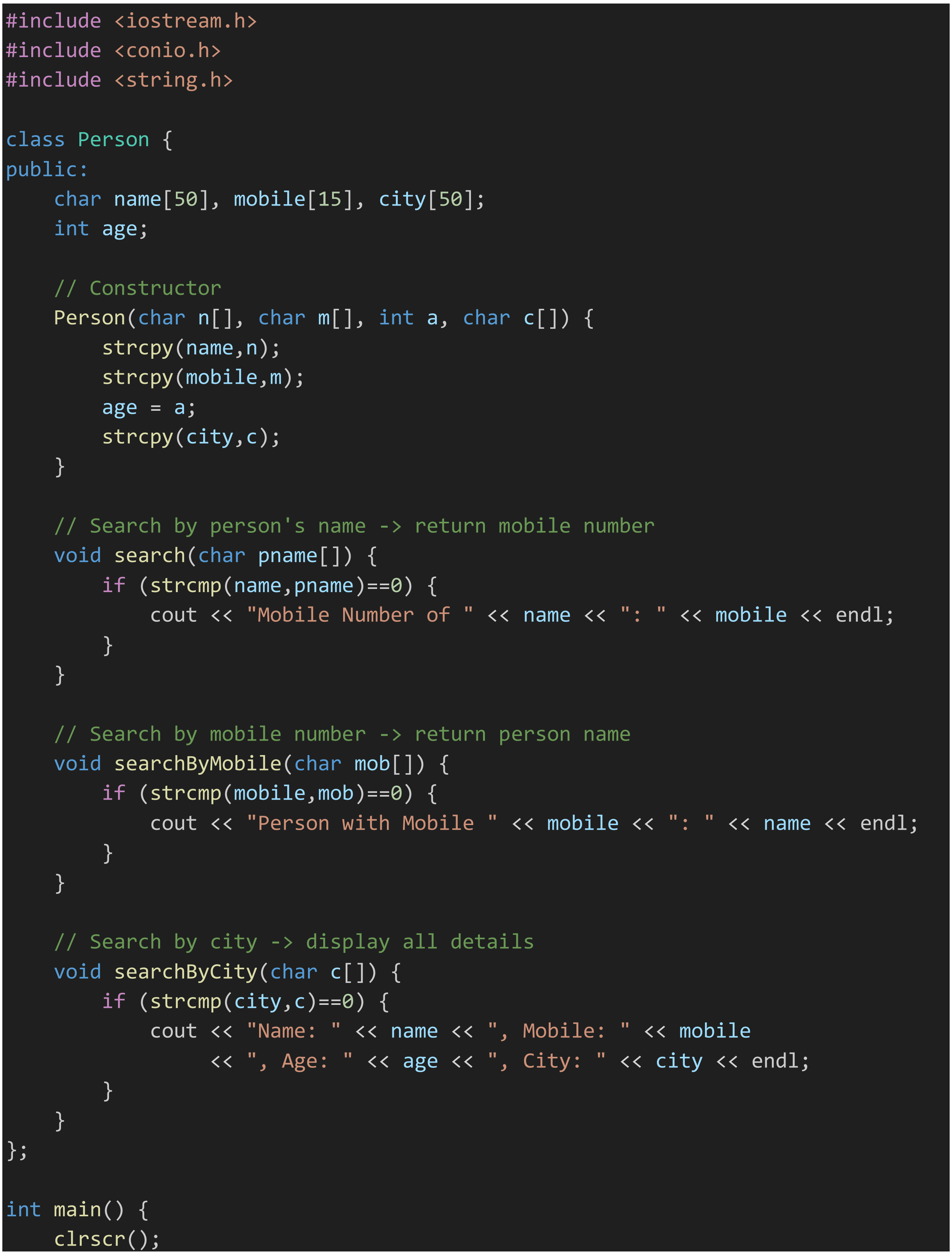


```
Enter Product ID: 1
Enter Product Name: soap
Enter Quantity: 10
Enter Price: 10
Enter Product ID: 2
Enter Product Name: oil
Enter Quantity: 10
Enter Price: 50

Product Details:
Product ID: 1
Product Name: soap
Quantity: 10
Price: 10

Product ID: 2
Product Name: oil
Quantity: 10
Price: 50
Total Products Created: 2
```

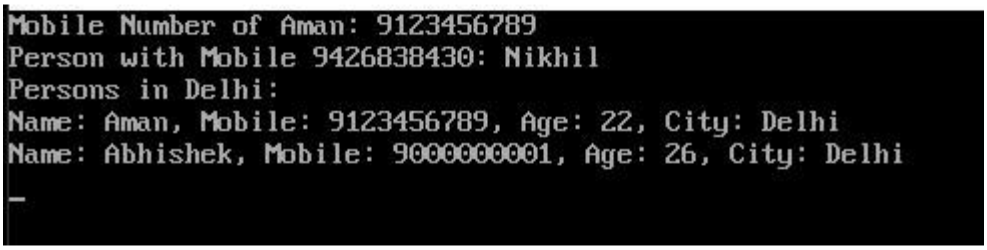
b) Create a C++ class person with data members person_name, mobile_number, age, city. Write necessary member functions for the following.

1. **accept person details**
2. **display person details**
3. **search person detail with mobile number**

[Marks 25]



```cpp
#include <iostream.h>
#include <conio.h>
#include <string.h>

class Person {
public:
    char name[50], mobile[15], city[50];
    int age;

    // Constructor
    Person(char n[], char m[], int a, char c[]) {
        strcpy(name,n);
        strcpy(mobile,m);
        age = a;
        strcpy(city,c);
    }

    // Search by person's name -> return mobile number
    void search(char pname[]) {
        if (strcmp(name,pname)==0) {
            cout << "Mobile Number of " << name << ": " << mobile << endl;
        }
    }

    // Search by mobile number -> return person name
    void searchByMobile(char mob[]) {
        if (strcmp(mobile,mob)==0) {
            cout << "Person with Mobile " << mobile << ": " << name << endl;
        }
    }

    // Search by city -> display all details
    void searchByCity(char c[]) {
        if (strcmp(city,c)==0) {
            cout << "Name: " << name << ", Mobile: " << mobile
                 << ", Age: " << age << ", City: " << city << endl;
        }
    }
};

int main() {
    clrscr();
    // Array of Persons
    Person persons[] = {
        Person("Nikhil", "9426838430", 25, "Mumbai"),
        Person("Aman", "9123456789", 22, "Delhi"),
        Person("Kunal", "9988776655", 24, "Mumbai"),
        Person("Ayush", "9090909090", 23, "Pune"),
        Person("Abhishek", "9000000001", 26, "Delhi")
    };

    char searchName[] = "Aman";
    char searchMobile[] = "9426838430";
    char searchCity[] = "Delhi";

    // Searching mobile number by name
    for(int i=0;i<5;i++)
    persons[i].search(searchName);

    // Searching person name by mobile number
    for(int j=0;j<5;j++)
    persons[j].searchByMobile(searchMobile);

    // Searching all persons in a given city
    cout << "Persons in " << searchCity << ":" << endl;
    for(int k=0;k<5;k++)
    persons[k].searchByCity(searchCity);

        getch();
    return 0;
}
```

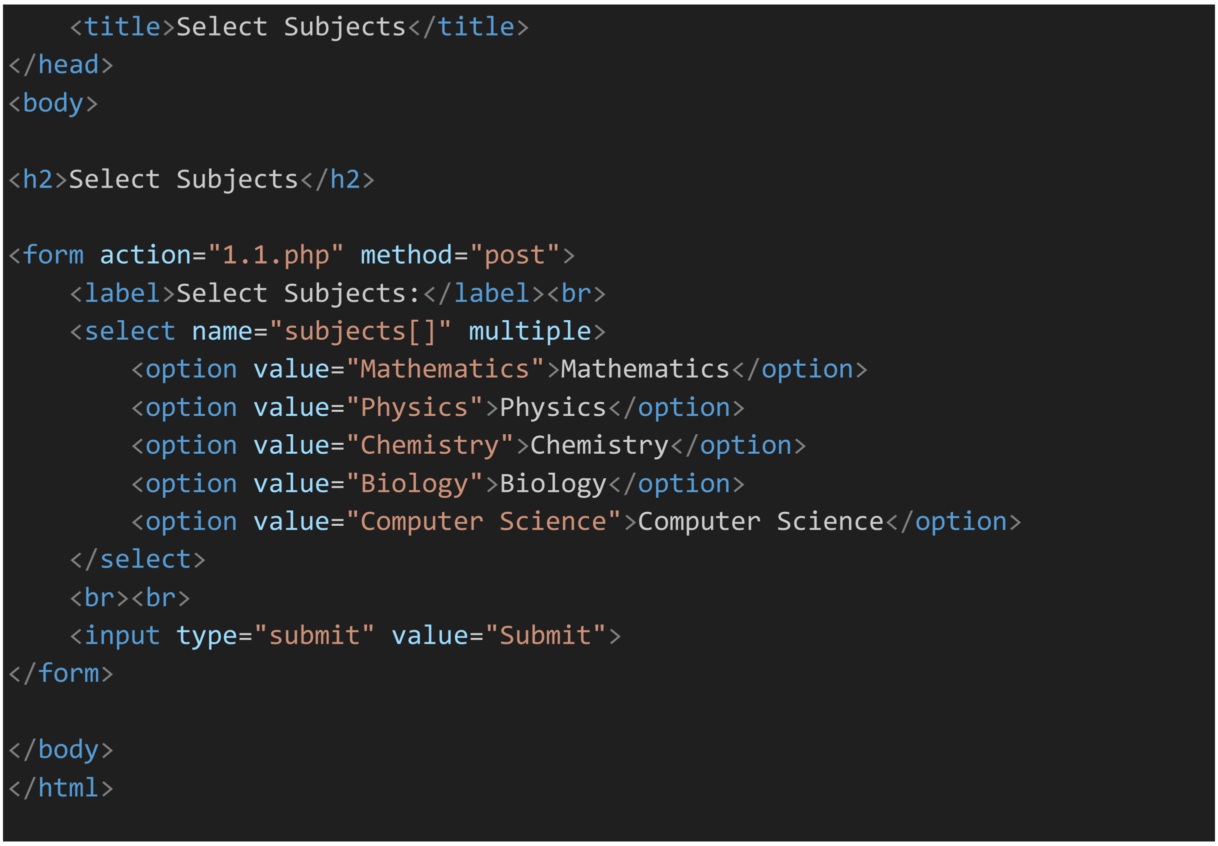


```
Mobile Number of Aman: 9123456789
Person with Mobile 9426838430: Nikhil
Persons in Delhi:
Name: Aman, Mobile: 9123456789, Age: 22, City: Delhi
Name: Abhishek, Mobile: 9000000001, Age: 26, City: Delhi
_
```

Q. 2)

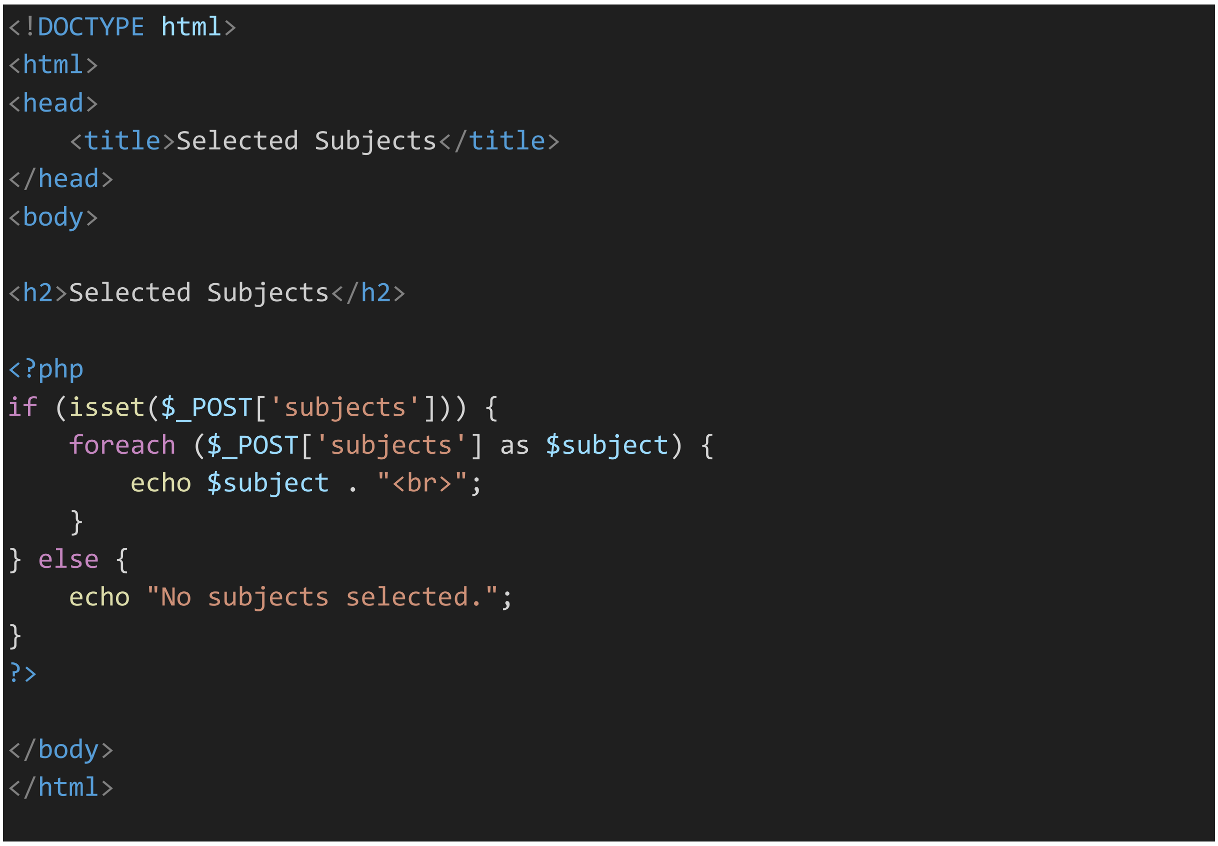
a) Write PHP program to select list of subjects front list box and displays the selected subject information on next page. (Use sticky Multivalued parameter). [Marks l5]

```
<!DOCTYPE html>
<html>
<head>
```

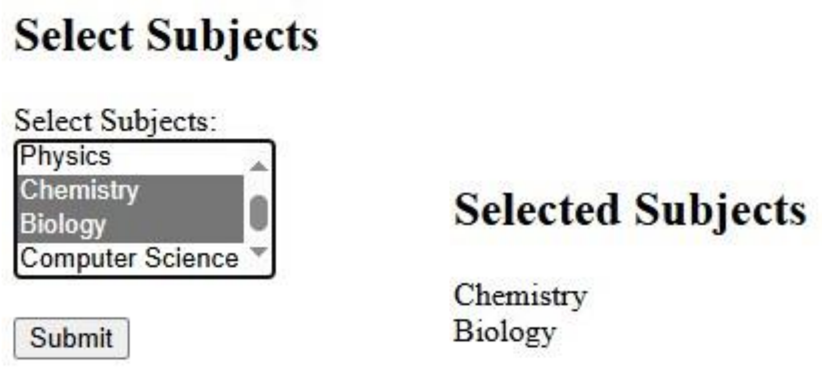


```
    <title>Select Subjects</title>
</head>
<body>

<h2>Select Subjects</h2>

<form action="1.1.php" method="post">
    <label>Select Subjects:</label><br>
    <select name="subjects[]" multiple>
        <option value="Mathematics">Mathematics</option>
        <option value="Physics">Physics</option>
        <option value="Chemistry">Chemistry</option>
        <option value="Biology">Biology</option>
        <option value="Computer Science">Computer Science</option>
    </select>
    <br><br>
    <input type="submit" value="Submit">
</form>

</body>
</html>
```

PHP File:-



```
<!DOCTYPE html>
<html>
<head>
    <title>Selected Subjects</title>
</head>
<body>

<h2>Selected Subjects</h2>

<?php
if (isset($_POST['subjects'])) {
    foreach ($_POST['subjects'] as $subject) {
        echo $subject . "<br>";
    }
} else {
    echo "No subjects selected.";
}
?>

</body>
</html>
```



Select Subjects

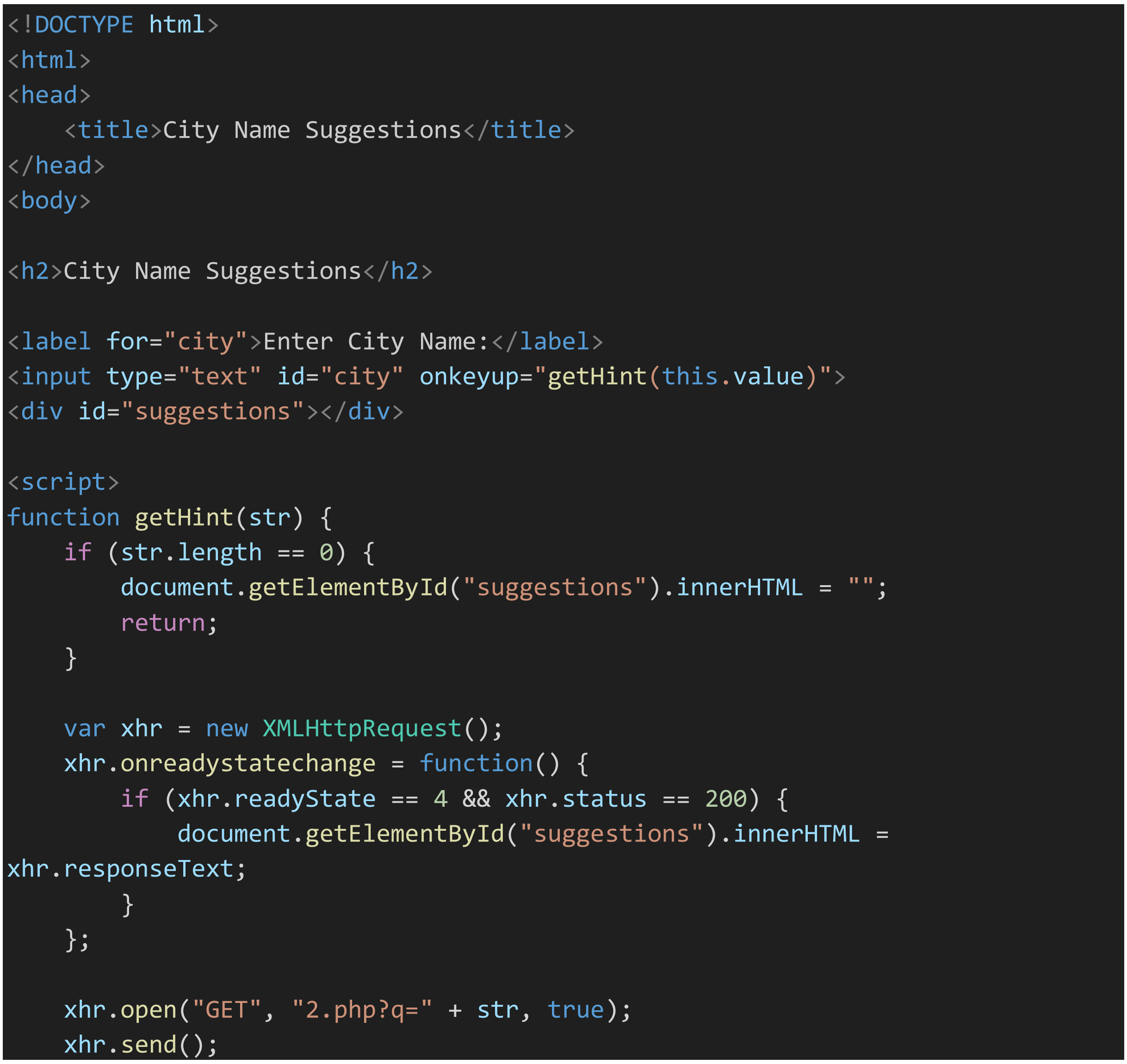
Select Subjects:

Selected Subjects

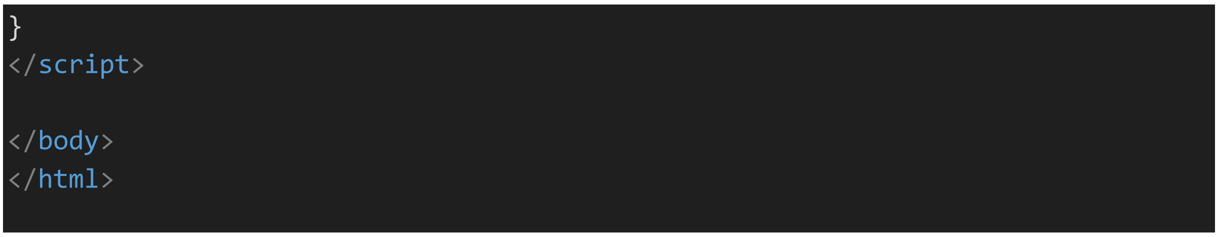
Chemistry
Biology

b) Write a PHP script using AJAX concept, to give hint to user when he/she type city name in the text field.
[Marks 25]

HTML / AJAX File:-

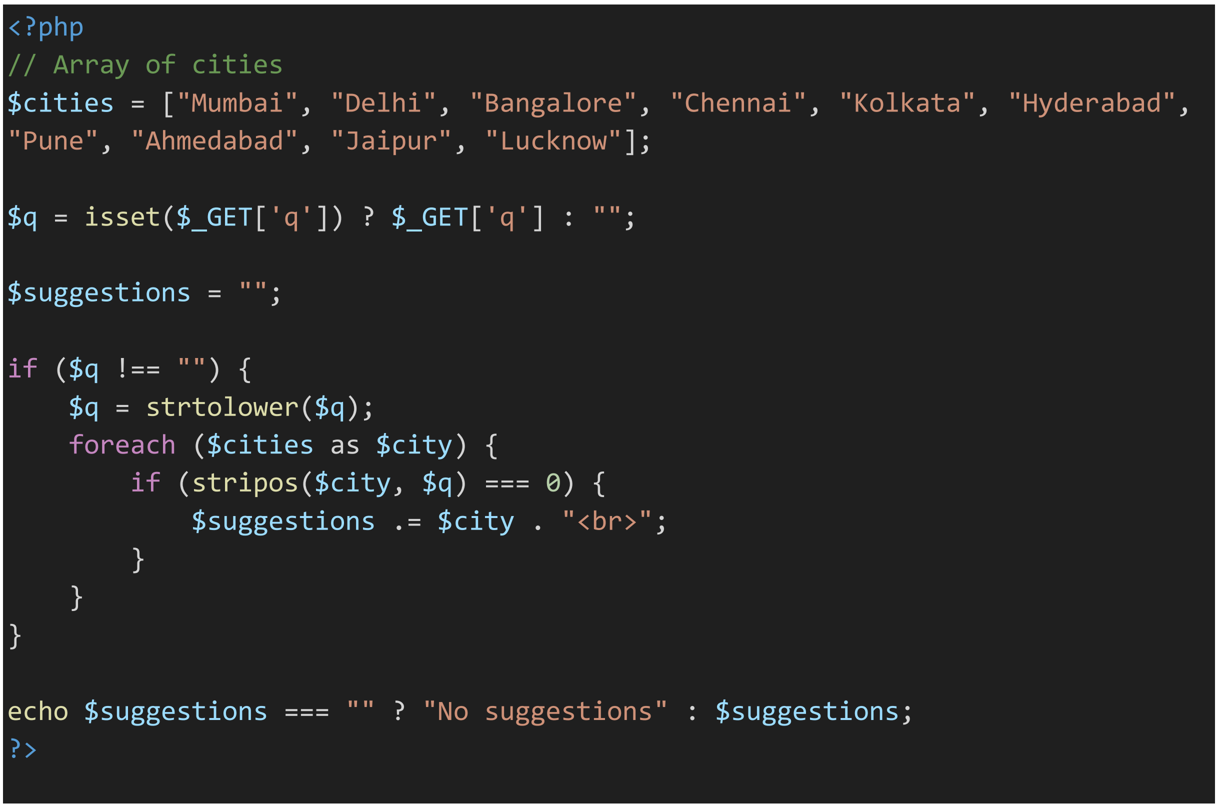


```
<!DOCTYPE html>
<html>
<head>
    <title>City Name Suggestions</title>
</head>
<body>

<h2>City Name Suggestions</h2>

<label for="city">Enter City Name:</label>
<input type="text" id="city" onkeyup="getHint(this.value)">
<div id="suggestions"></div>

<script>
function getHint(str) {
    if (str.length == 0) {
        document.getElementById("suggestions").innerHTML = "";
        return;
    }

    var xhr = new XMLHttpRequest();
    xhr.onreadystatechange = function() {
        if (xhr.readyState == 4 && xhr.status == 200) {
            document.getElementById("suggestions").innerHTML =
xhr.responseText;
        }
    };

    xhr.open("GET", "2.php?q=" + str, true);
    xhr.send();
```

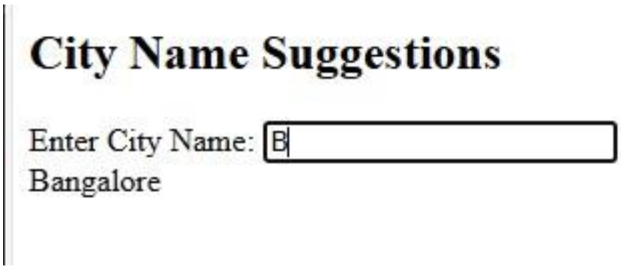


```
}
</script>

</body>
</html>
```

PHP File:-



```
<?php
// Array of cities
$cities = ["Mumbai", "Delhi", "Bangalore", "Chennai", "Kolkata", "Hyderabad",
"Pune", "Ahmedabad", "Jaipur", "Lucknow"];

$q = isset($_GET['q']) ? $_GET['q'] : "";

$suggestions = "";

if ($q !== "") {
    $q = strtolower($q);
    foreach ($cities as $city) {
        if (stripos($city, $q) === 0) {
            $suggestions .= $city . "<br>";
        }
    }
}

echo $suggestions === "" ? "No suggestions" : $suggestions;
?>
```



City Name Suggestions

Enter City Name: B

Bangalore

Q. 3) Lab Book **[Marks 10]**

Q. 4) Viva **[Marks 10]**

Slip -8

Q. 1)

a) Write a C++ program to accept Worker information worker name, no of hours worked, pay rate and salary. Write necessary functions to calculate and display the salary of worker. (Use default value for pay_rate)
[Marks 15]

Already Done Above { Kisi Ek Slip me tha }

b) Write a C++ Program to create a base class Employee(EmpCode, Empname, EmpSalary). Derive two classes as fulltime(DailyWages, No_Of_Days) and Parttime (No_of_Hrs, Hr_Wages).

Write a menu driven program to perform following functions.

1. **Accept the details of 'n' employees and calculate salary.**
2. **Display the details of 'n' employees.**
3. **Display the details of employee having maximum salary for both types of employees**

[Marks 25]

Already Done Above { Kisi Ek Slip me tha }

Q. 2)

a) Write PHP program to select list of subjects front list box and displays the selected subject information on next page. (Use sticky Multivalued parameter). **[Marks l5]**

Already Done Above { Kisi Ek Slip me tha }

b) Write a PHP script using AJAX concept, to give hint to user when he/she type city name in the text field. **[Marks 25]**

Already Done Above { Kisi Ek Slip me tha }

Q. 3) Lab Book **[Marks 10]**

Q. 4) Viva **[Marks 10]**

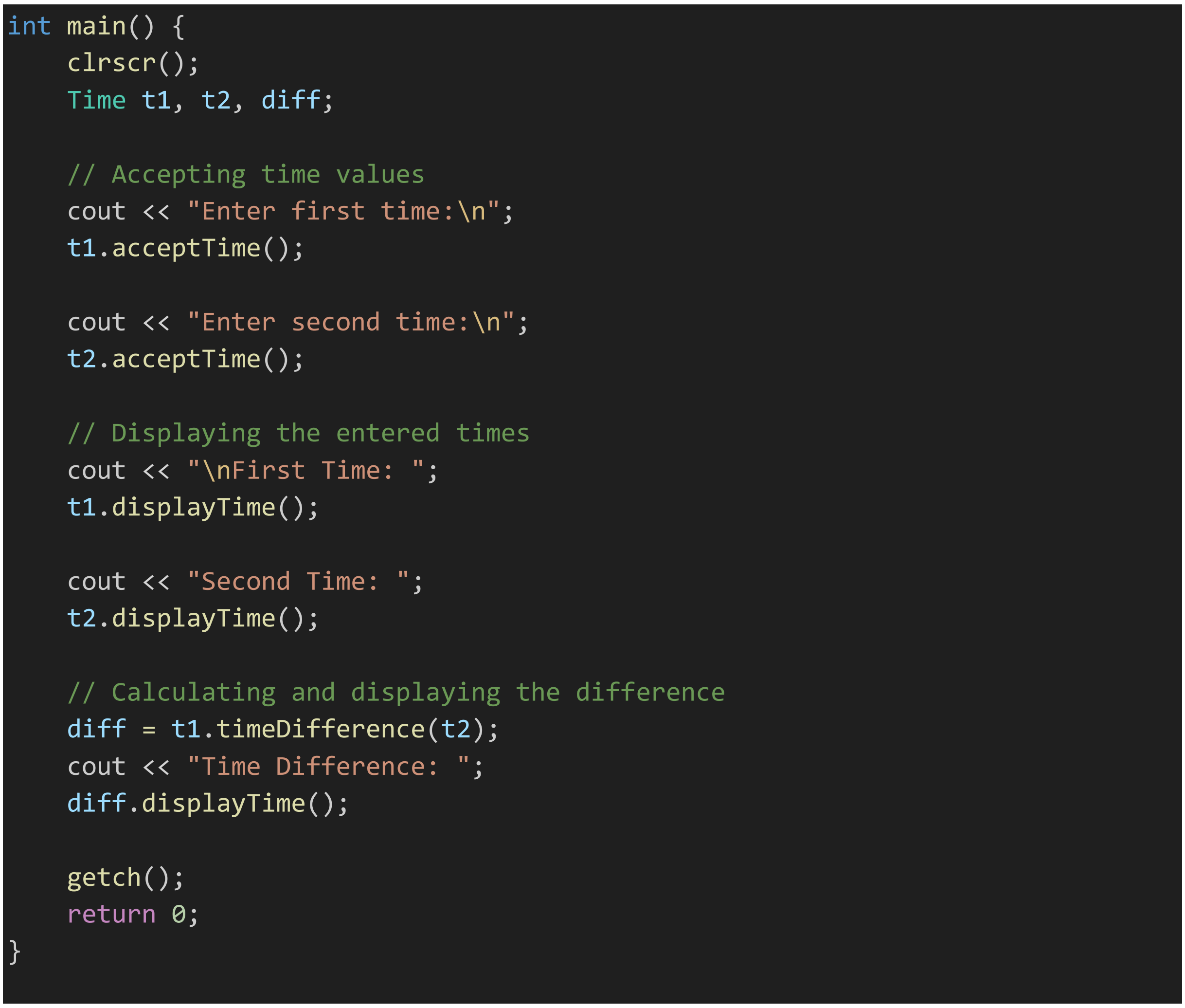
Slip -9

Q. 1)

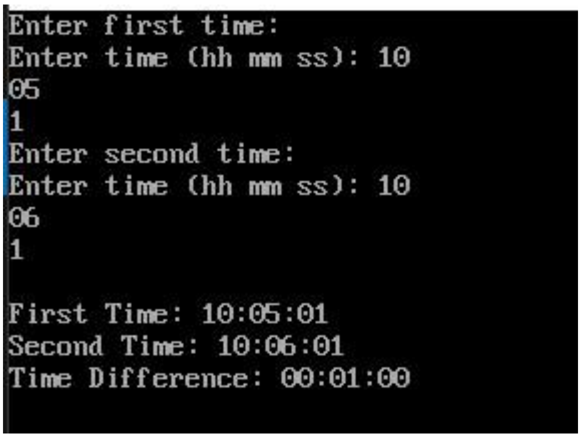
a) Create a C++ class time with data members Hours, Minutes and Seconds. Write necessary member functions for the following (Use Object as arguments)

1. to accept time
2. to display time in hh:mm:ss
3. to find difference between two time and display difference in hh:mm:ss [Marks 15]

```
#include <iostream.h>
#include <conio.h>
#include <stdlib.h>

class Time {
    int hours, minutes, seconds;

public:
    // Constructor to initialize time
    Time(int h = 0, int m = 0, int s = 0) {
        hours = h;
        minutes = m;
        seconds = s;
    }

    // Function to accept time from the user
    void acceptTime() {
        cout << "Enter time (hh mm ss): ";
        cin >> hours >> minutes >> seconds;
    }

    // Function to display time in hh:mm:ss format
    void displayTime() {
        cout << (hours < 10 ? "0" : "") << hours << ":"
             << (minutes < 10 ? "0" : "") << minutes << ":"
             << (seconds < 10 ? "0" : "") << seconds << endl;
    }

    // Function to find the difference between two time objects
    Time timeDifference(Time t) {
        int totalSec1 = hours * 3600 + minutes * 60 + seconds;
        int totalSec2 = t.hours * 3600 + t.minutes * 60 + t.seconds;
        int diffSec = abs(totalSec1 - totalSec2);

        return Time(diffSec / 3600, (diffSec % 3600) / 60, diffSec % 60);
    }
};
```



```
int main() {
    clrscr();
    Time t1, t2, diff;

    // Accepting time values
    cout << "Enter first time:\n";
    t1.acceptTime();

    cout << "Enter second time:\n";
    t2.acceptTime();

    // Displaying the entered times
    cout << "\nFirst Time: ";
    t1.displayTime();

    cout << "Second Time: ";
    t2.displayTime();

    // Calculating and displaying the difference
    diff = t1.timeDifference(t2);
    cout << "Time Difference: ";
    diff.displayTime();

    getch();
    return 0;
}
```



```
Enter first time:
Enter time (hh mm ss): 10
05
1
Enter second time:
Enter time (hh mm ss): 10
06
1

First Time: 10:05:01
Second Time: 10:06:01
Time Difference: 00:01:00
```

b) Design two base classes student(sid,sname,class) and Competition(Cid, Cname). Derive a class Stud_Comp(Rank) from it. Write a menu driven program to perform following functions.

i. Accept Information

ii. Display Information

iii. Display student details in the ascending order of rank of a specified competition.

(Use array of objects) [Marks 25]

Already Done Above { Kisi Ek Slip me tha }

Q. 2)

a) Write a PHP program to accept two string from user and check whether entered strings are matching or not. (Use sticky form concept) [Marks 15]

Already Done Above { Kisi Ek Slip me tha }

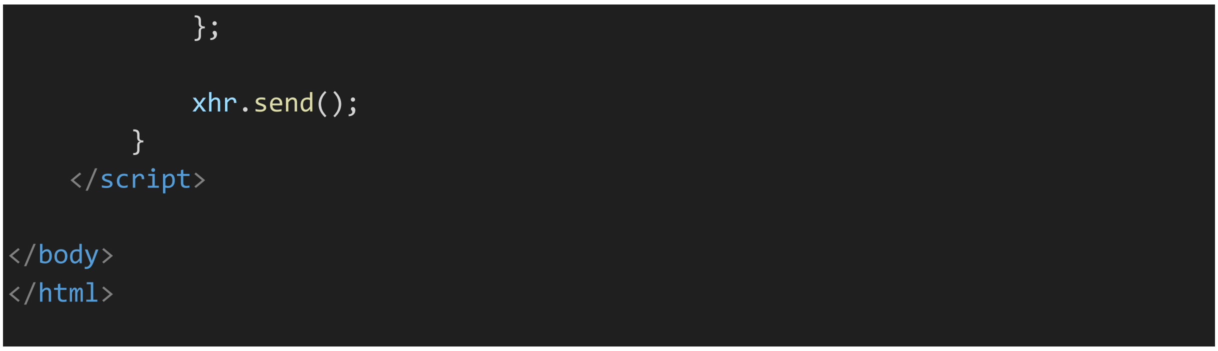
b) Write a Ajax program to get book details from XML file when user select book name. Create XML file to store details of book(name, author, year and price) [Marks 25]

PHP File:-

```php
<?php
if (isset($_GET['name'])) {
    $bookName = $_GET['name'];
    $xml = simplexml_load_file("2.1.xml") or die("XML file not found");

    foreach ($xml->Book as $book) {
        if ((string)$book->Name == $bookName) {
            echo "<p><strong>Name:</strong> " . $book->Name . "</p>";
            echo "<p><strong>Author:</strong> " . $book->Author . "</p>";
            echo "<p><strong>Year:</strong> " . $book->Year . "</p>";
            echo "<p><strong>Price:</strong> $" . $book->Price . "</p>";
            exit;
        }
    }

    echo "<p>Book not found</p>";
}
?>
```

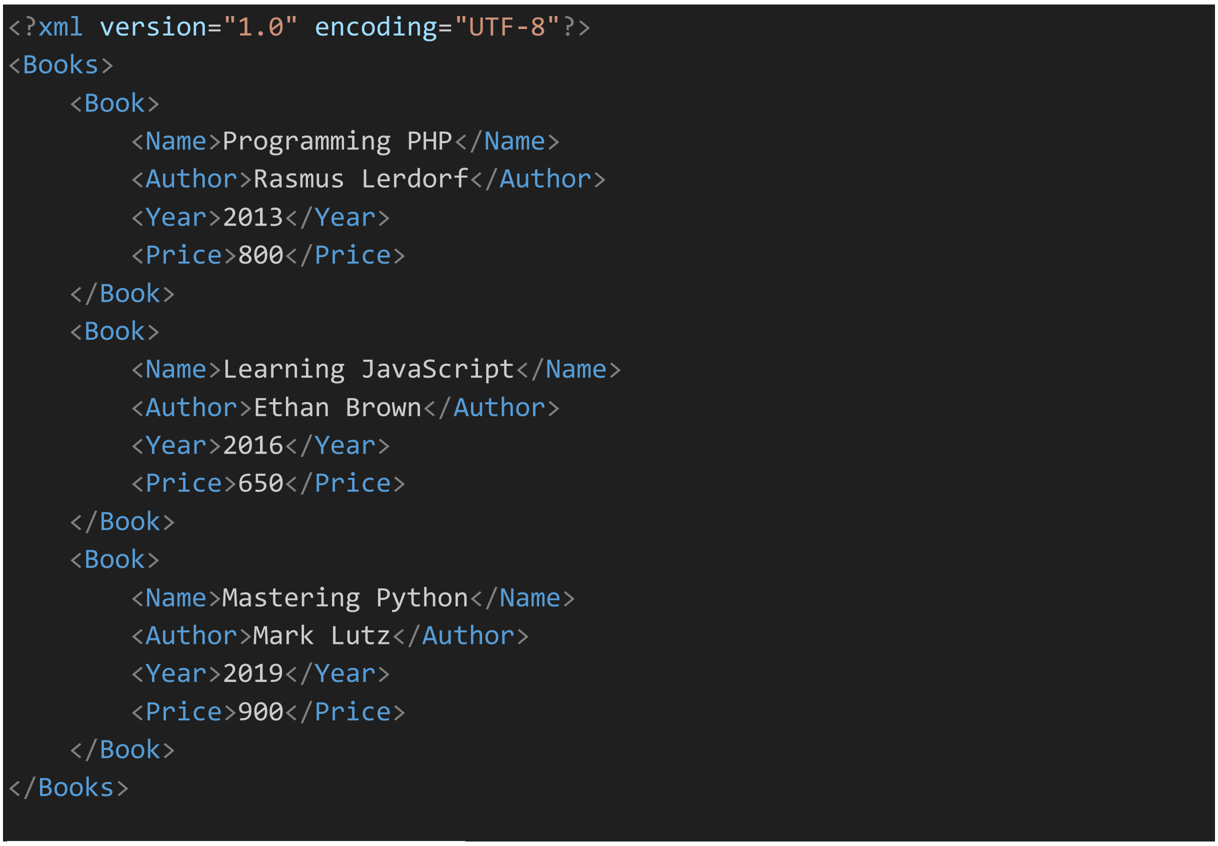
HTML / AJAX :-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Book Details</title>
</head>
<body>

    <h2>Select a Book:</h2>
    <select id="bookSelect" onchange="getBookDetails()">
        <option value="">--Select Book--</option>
        <option value="Programming PHP">Programming PHP</option>
        <option value="Learning JavaScript">Learning JavaScript</option>
        <option value="Mastering Python">Mastering Python</option>
    </select>

    <h3>Book Details:</h3>
    <div id="bookDetails"></div>

    <script>
        function getBookDetails() {
            var bookName = document.getElementById("bookSelect").value;
            var bookDetailsDiv = document.getElementById("bookDetails");

            if (bookName === "") {
                bookDetailsDiv.innerHTML = "";
                return;
            }

            var xhr = new XMLHttpRequest();
            xhr.open("GET", "2.php?name=" + encodeURIComponent(bookName),
true);

            xhr.onreadystatechange = function () {
                if (xhr.readyState == 4 && xhr.status == 200) {
                    bookDetailsDiv.innerHTML = xhr.responseText; // Directly
display response
                }
```

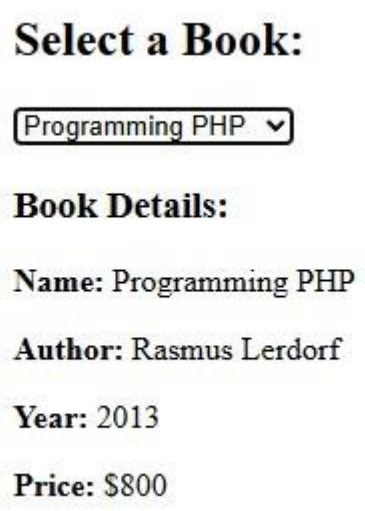


```
            };

            xhr.send();
        }
    </script>

</body>
</html>
```

XML File:-



```
<?xml version="1.0" encoding="UTF-8"?>
<Books>
    <Book>
        <Name>Programming PHP</Name>
        <Author>Rasmus Lerdorf</Author>
        <Year>2013</Year>
        <Price>800</Price>
    </Book>
    <Book>
        <Name>Learning JavaScript</Name>
        <Author>Ethan Brown</Author>
        <Year>2016</Year>
        <Price>650</Price>
    </Book>
    <Book>
        <Name>Mastering Python</Name>
        <Author>Mark Lutz</Author>
        <Year>2019</Year>
        <Price>900</Price>
    </Book>
</Books>
```



Select a Book:

Programming PHP ▾

Book Details:

Name: Programming PHP

Author: Rasmus Lerdorf

Year: 2013

Price: $800

Q. 3) Lab Book	**[Marks 10]**
Q. 4) Viva	**[Marks 10]**

Slip -10

Q. 1)

a) Create a C++ Class Student with data members Rollno, Sname, Class, Percentage. Accept two students information and display information of students having maximum percentage. (Use this pointer.
[Marks 15]

Already Done Above { Kisi Ek Slip me tha }

b) Create a C++ Class FixDeposite with data members FD_No, Cust_Name, FD_Amt, Interest rate, Maturity_Amt, No_Of_Months. Create and initialize all values of FixDeposit object by using parameterized constructor default value for interest rate. Calculate maturity amt using interest rate and display all the details.
[Marks 25]

Already Done Above { Kisi Ek Slip me tha }

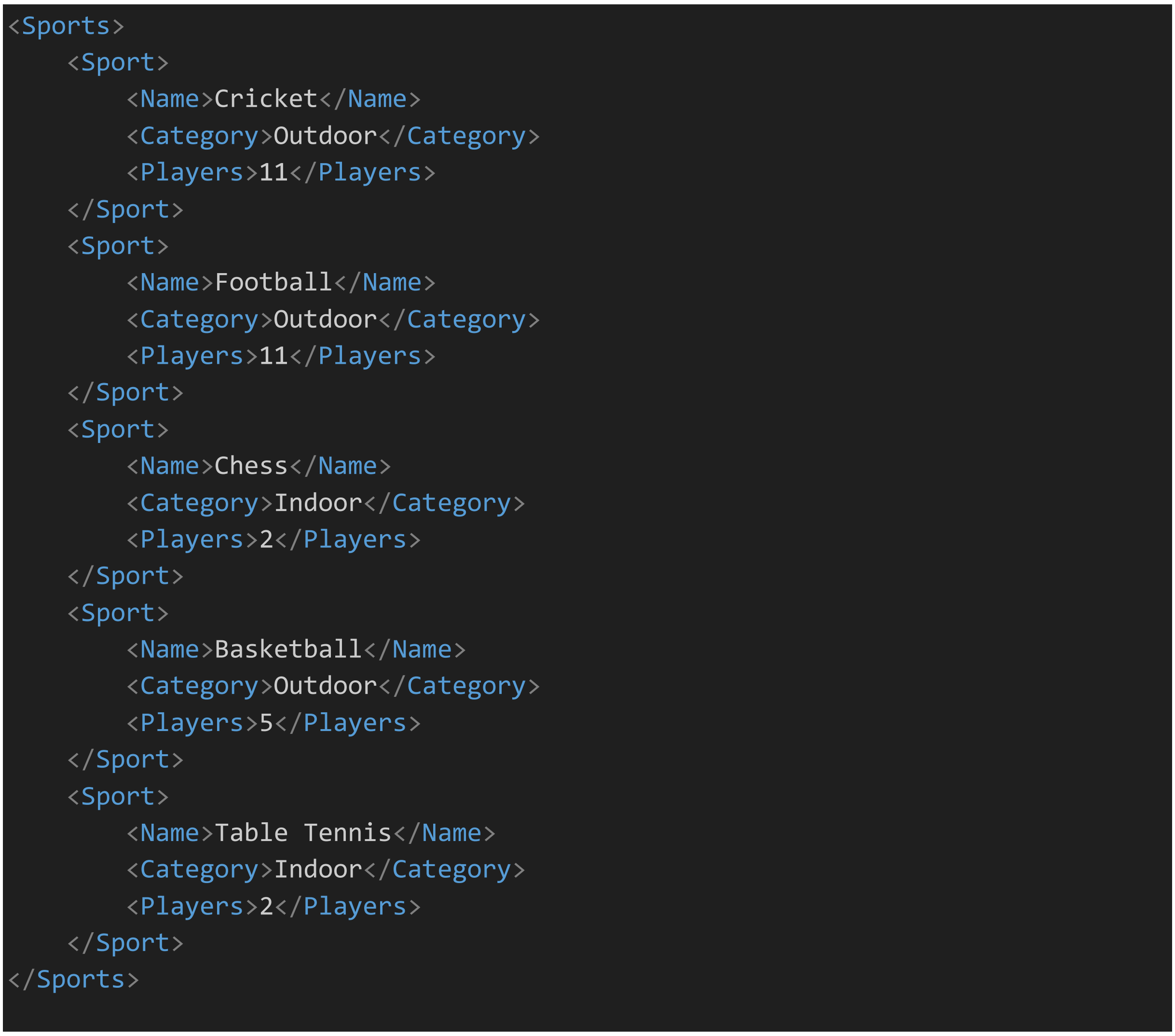
Q. 2)

a) Create an application that reads “Sports.xml” file into sample XML object. Display attributes and elements. (Hint : Use simple_xml_load_file() function)
[Marks l5]

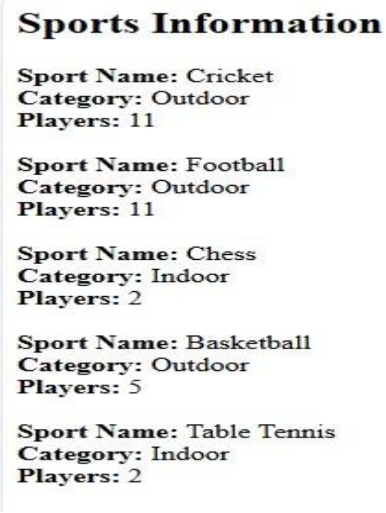
PHP File:-

```
<?php
// Load XML file
$xml = simplexml_load_file("Sports.xml") or die("Error: Cannot load
Sports.xml");

// Display XML elements
echo "<h2>Sports Information</h2>";

foreach ($xml->Sport as $sport) {
    echo "<b>Sport Name:</b> " . $sport->Name . "<br>";
    echo "<b>Category:</b> " . $sport->Category . "<br>";
    echo "<b>Players:</b> " . $sport->Players . "<br><br>";
}
?>
```

XML File:-



```xml
<Sports>
    <Sport>
        <Name>Cricket</Name>
        <Category>Outdoor</Category>
        <Players>11</Players>
    </Sport>
    <Sport>
        <Name>Football</Name>
        <Category>Outdoor</Category>
        <Players>11</Players>
    </Sport>
    <Sport>
        <Name>Chess</Name>
        <Category>Indoor</Category>
        <Players>2</Players>
    </Sport>
    <Sport>
        <Name>Basketball</Name>
        <Category>Outdoor</Category>
        <Players>5</Players>
    </Sport>
    <Sport>
        <Name>Table Tennis</Name>
        <Category>Indoor</Category>
        <Players>2</Players>
    </Sport>
</Sports>
```



Sports Information

Sport Name: Cricket
Category: Outdoor
Players: 11

Sport Name: Football
Category: Outdoor
Players: 11

Sport Name: Chess
Category: Indoor
Players: 2

Sport Name: Basketball
Category: Outdoor
Players: 5

Sport Name: Table Tennis
Category: Indoor
Players: 2

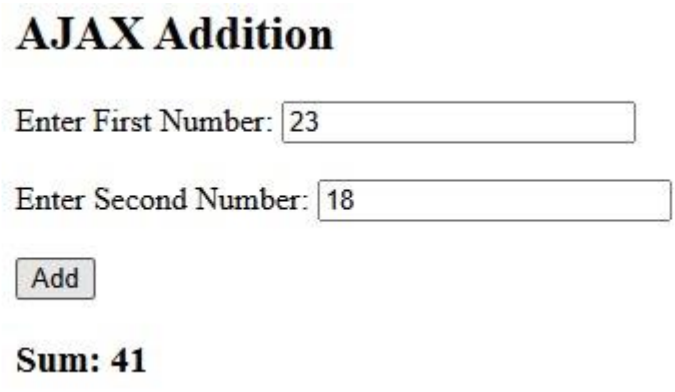
b) Write a simple PHP program which implements Ajax for addition of two numbers. [Marks 25]

HTML / AJAX :-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>AJAX Addition</title>
    <script>
        function addNumbers() {
            let num1 = document.getElementById("num1").value.trim();
            let num2 = document.getElementById("num2").value.trim();

            // Convert input values to numbers and handle empty values
            let n1 = num1 !== "" ? parseFloat(num1) : 0;
            let n2 = num2 !== "" ? parseFloat(num2) : 0;

            let xhr = new XMLHttpRequest();
            xhr.open("POST", "1.php", true); // Request sent to add.php
            xhr.setRequestHeader("Content-Type", "application/x-www-form-
urlencoded");

            xhr.onreadystatechange = function() {
                if (xhr.readyState == 4 && xhr.status == 200) {
                    console.log("Server Response:", xhr.responseText); //
Debugging
                    document.getElementById("result").innerHTML = "Sum: " +
xhr.responseText;
                }
            };

            xhr.send("num1=" + n1 + "&num2=" + n2 + "&ajax=1");
        }
    </script>
</head>
<body>

    <h2>AJAX Addition</h2>
    <label>Enter First Number: </label>
    <input type="number" id="num1"><br><br>

    <label>Enter Second Number: </label>
    <input type="number" id="num2"><br><br>

    <button onclick="addNumbers()">Add</button>
    <h3 id="result"></h3>

</body>
</html>
```

PHP File:-

```
<?php
// Handle AJAX request
if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST["ajax"])) {
    $num1 = isset($_POST["num1"]) && $_POST["num1"] !== "" ?
(float)$_POST["num1"] : 0;
    $num2 = isset($_POST["num2"]) && $_POST["num2"] !== "" ?
(float)$_POST["num2"] : 0;

    echo $num1 + $num2;
    exit; // Stop execution after sending AJAX response
}
?>
```



AJAX Addition

Enter First Number: 23

Enter Second Number: 18

Sum: 41

Q. 3) Lab Book **[Marks 10]**

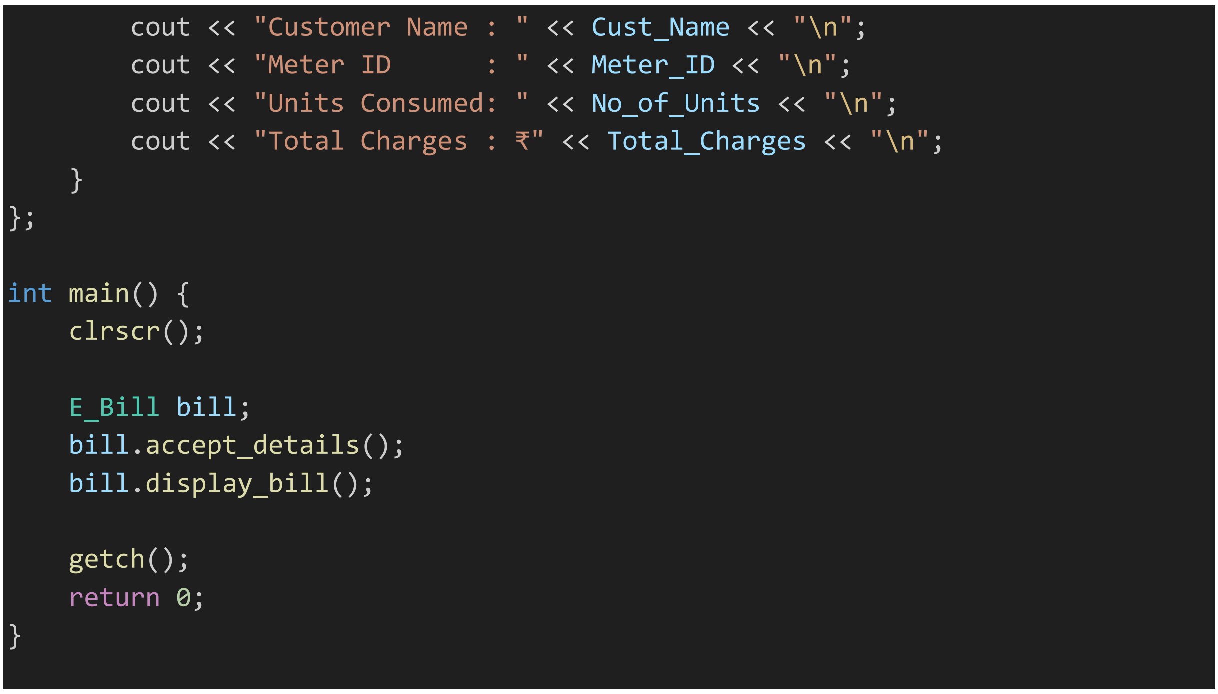
Q. 4) Viva **[Marks 10]**

Slip -11

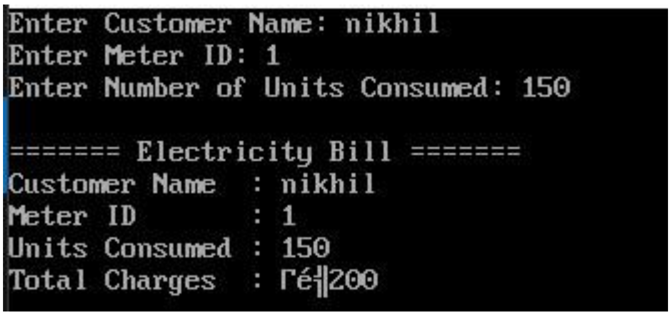
Q. 1)

a) Write a C++ program to create a class E_Bill with data members Cust_name, Meter_ID, No_of_Units and Total_Charges. Write member function to accept and display customer information by calculating charges. (Rules to calculate electricity board charges) [Marks 15]

```
#include <iostream.h>
#include <conio.h>

class E_Bill {
private:
    char Cust_Name[50];
    int Meter_ID, No_of_Units;
    float Total_Charges;

public:
    void accept_details() {
        cout << "Enter Customer Name: ";
        cin >> Cust_Name;

        cout << "Enter Meter ID: ";
        cin >> Meter_ID;

        cout << "Enter Number of Units Consumed: ";
        cin >> No_of_Units;

        calculate_charges();
    }

    void calculate_charges() {
        if (No_of_Units <= 100)
            Total_Charges = No_of_Units * 1;
        else if (No_of_Units <= 300)
            Total_Charges = (100 * 1) + ((No_of_Units - 100) * 2);
        else
            Total_Charges = (100 * 1) + (200 * 2) + ((No_of_Units - 300) * 5);

        if (Total_Charges < 150)
            Total_Charges = 150;

        if (Total_Charges > 250)
            Total_Charges += Total_Charges * 0.15;
    }

    void display_bill() {
        cout << "\n======= Electricity Bill =======\n";
```



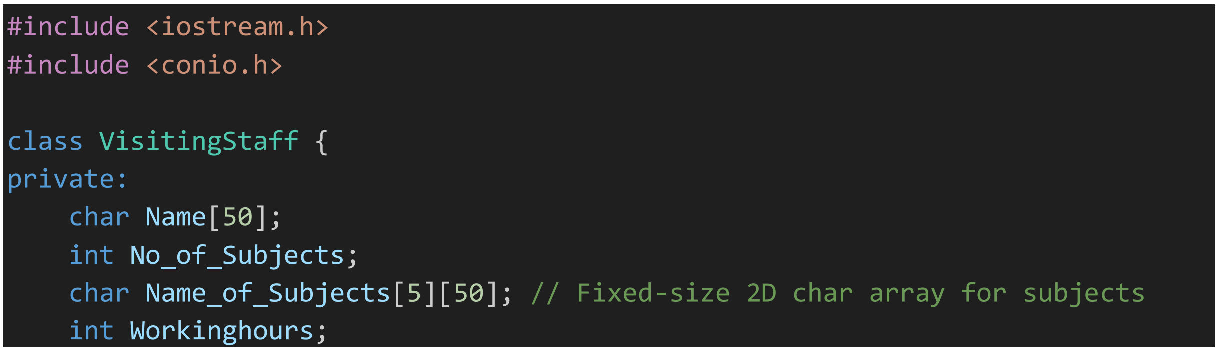
```
        cout << "Customer Name : " << Cust_Name << "\n";
        cout << "Meter ID      : " << Meter_ID << "\n";
        cout << "Units Consumed: " << No_of_Units << "\n";
        cout << "Total Charges : ₹" << Total_Charges << "\n";
    }
};

int main() {
    clrscr();

    E_Bill bill;
    bill.accept_details();
    bill.display_bill();

    getch();
    return 0;
}
```



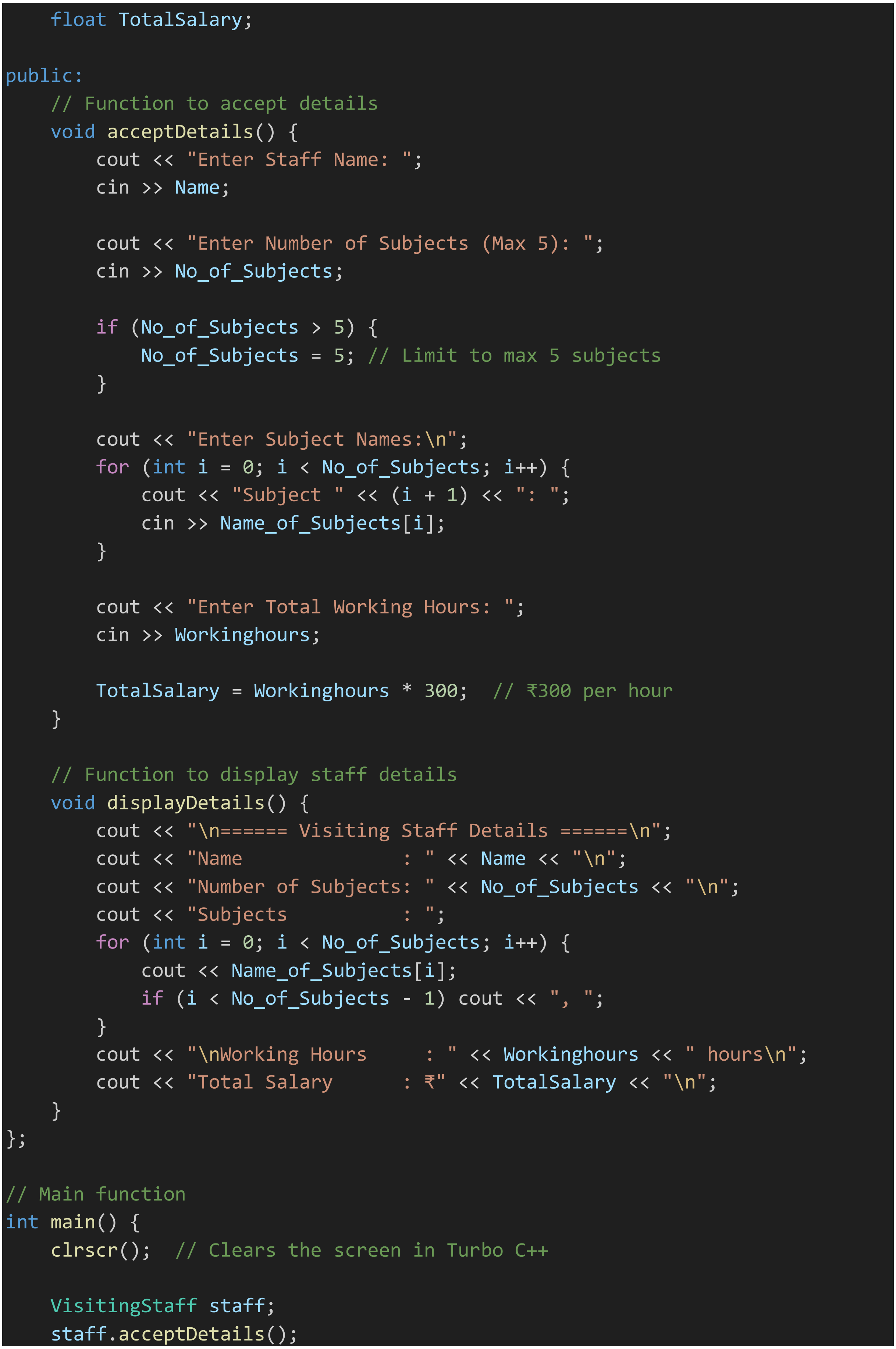
```
Enter Customer Name: nikhil
Enter Meter ID: 1
Enter Number of Units Consumed: 150

======= Electricity Bill =======
Customer Name  : nikhil
Meter ID       : 1
Units Consumed : 150
Total Charges  : Γé╣200
```

b) Create a C++ class visiting staff with data members name, no of subjects, name of subjects[], working hr, total salary. (number of subjects varies for a staff). Write a parameterized constructor to initialize the data members and create an array for name of subjects dynamically. Display visiting staff details by calculating salary. (Assume renumeration 300 per working hour)

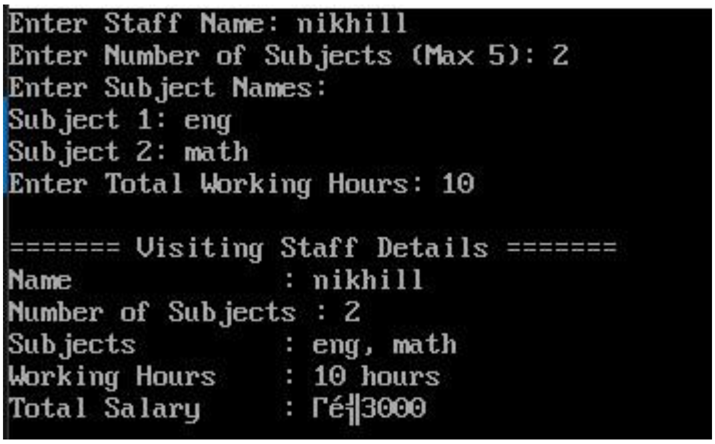
[Marks 25]



```
#include <iostream.h>
#include <conio.h>

class VisitingStaff {
private:
    char Name[50];
    int No_of_Subjects;
    char Name_of_Subjects[5][50]; // Fixed-size 2D char array for subjects
    int Workinghours;
```



```cpp
    float TotalSalary;

public:
    // Function to accept details
    void acceptDetails() {
        cout << "Enter Staff Name: ";
        cin >> Name;

        cout << "Enter Number of Subjects (Max 5): ";
        cin >> No_of_Subjects;

        if (No_of_Subjects > 5) {
            No_of_Subjects = 5; // Limit to max 5 subjects
        }

        cout << "Enter Subject Names:\n";
        for (int i = 0; i < No_of_Subjects; i++) {
            cout << "Subject " << (i + 1) << ": ";
            cin >> Name_of_Subjects[i];
        }

        cout << "Enter Total Working Hours: ";
        cin >> Workinghours;

        TotalSalary = Workinghours * 300;  // ₹300 per hour
    }

    // Function to display staff details
    void displayDetails() {
        cout << "\n====== Visiting Staff Details ======\n";
        cout << "Name              : " << Name << "\n";
        cout << "Number of Subjects: " << No_of_Subjects << "\n";
        cout << "Subjects          : ";
        for (int i = 0; i < No_of_Subjects; i++) {
            cout << Name_of_Subjects[i];
            if (i < No_of_Subjects - 1) cout << ", ";
        }
        cout << "\nWorking Hours     : " << Workinghours << " hours\n";
        cout << "Total Salary      : ₹" << TotalSalary << "\n";
    }
};

// Main function
int main() {
    clrscr();  // Clears the screen in Turbo C++

    VisitingStaff staff;
    staff.acceptDetails();
    staff.displayDetails();

    getch(); // Waits for user input before closing
    return 0;
}
```



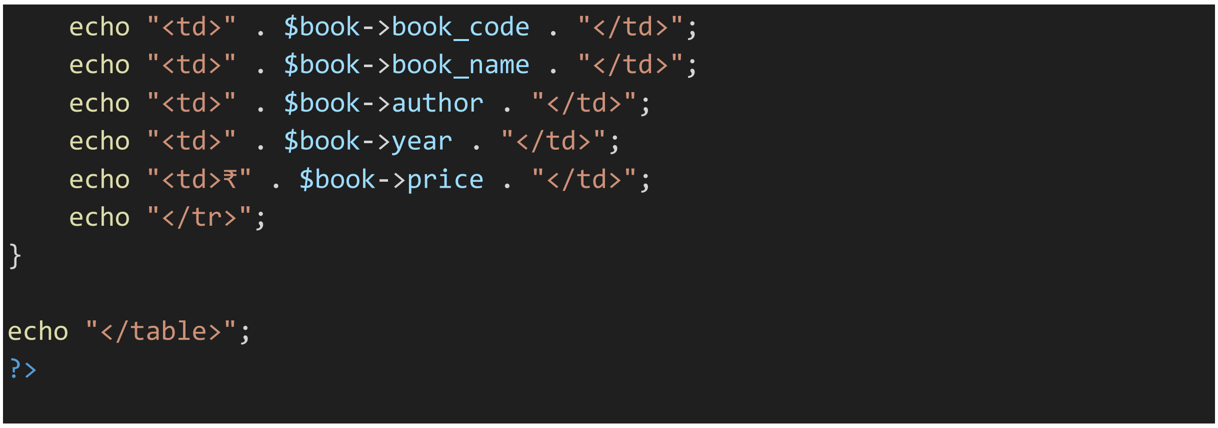
```
Enter Staff Name: nikhill
Enter Number of Subjects (Max 5): 2
Enter Subject Names:
Subject 1: eng
Subject 2: math
Enter Total Working Hours: 10

======= Visiting Staff Details =======
Name                : nikhill
Number of Subjects : 2
Subjects            : eng, math
Working Hours       : 10 hours
Total Salary        : Γé╣3000
```

Q. 2)

a) Write a PHP script to read book. XML and print book details in tabular format using simple XML(Content of book. XML are book_code, book_name, author, year, price) [Marks l5]

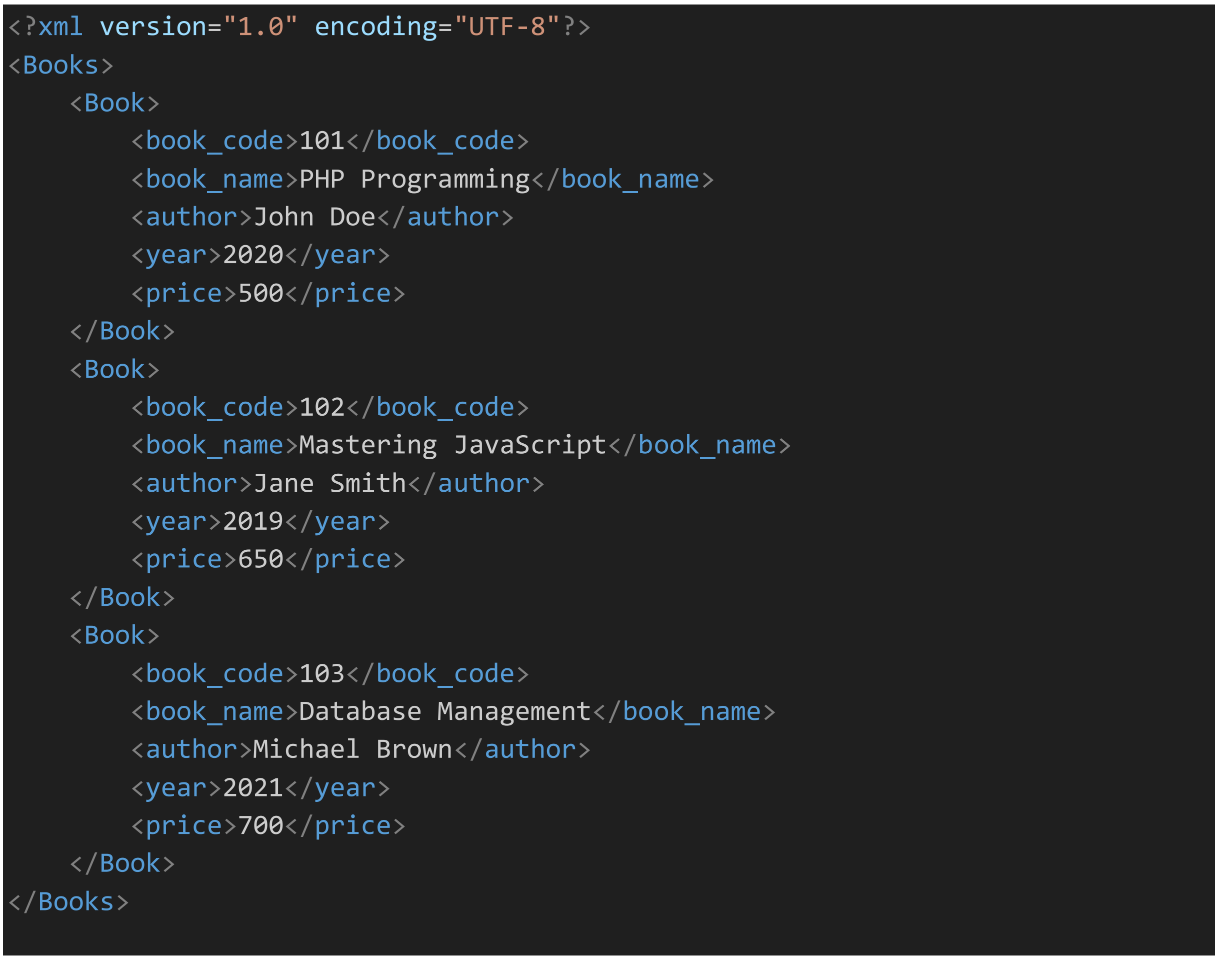
PHP File:-

```
<?php
// Load the XML file
$xml = simplexml_load_file("book.xml") or die("Error: Cannot load XML file.");

echo "<h2>Book Details</h2>";
echo "<table border='1' cellpadding='10' cellspacing='0'>";
echo "<tr>
        <th>Book Code</th>
        <th>Book Name</th>
        <th>Author</th>
        <th>Year</th>
        <th>Price</th>
      </tr>";

// Loop through each Book element
foreach ($xml->Book as $book) {
    echo "<tr>";
```

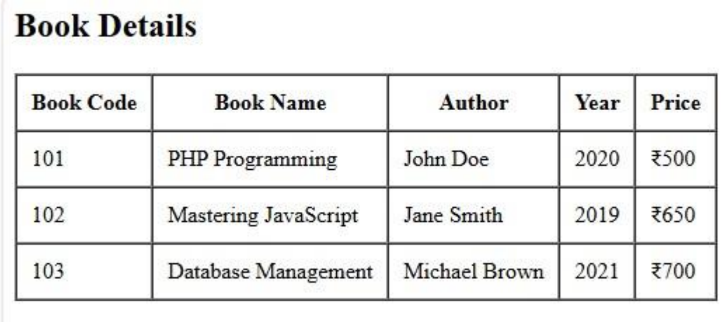


```php
    echo "<td>" . $book->book_code . "</td>";
    echo "<td>" . $book->book_name . "</td>";
    echo "<td>" . $book->author . "</td>";
    echo "<td>" . $book->year . "</td>";
    echo "<td>₹" . $book->price . "</td>";
    echo "</tr>";
}

echo "</table>";
?>
```

XML File:-



```xml
<?xml version="1.0" encoding="UTF-8"?>
<Books>
    <Book>
        <book_code>101</book_code>
        <book_name>PHP Programming</book_name>
        <author>John Doe</author>
        <year>2020</year>
        <price>500</price>
    </Book>
    <Book>
        <book_code>102</book_code>
        <book_name>Mastering JavaScript</book_name>
        <author>Jane Smith</author>
        <year>2019</year>
        <price>650</price>
    </Book>
    <Book>
        <book_code>103</book_code>
        <book_name>Database Management</book_name>
        <author>Michael Brown</author>
        <year>2021</year>
        <price>700</price>
    </Book>
</Books>
```

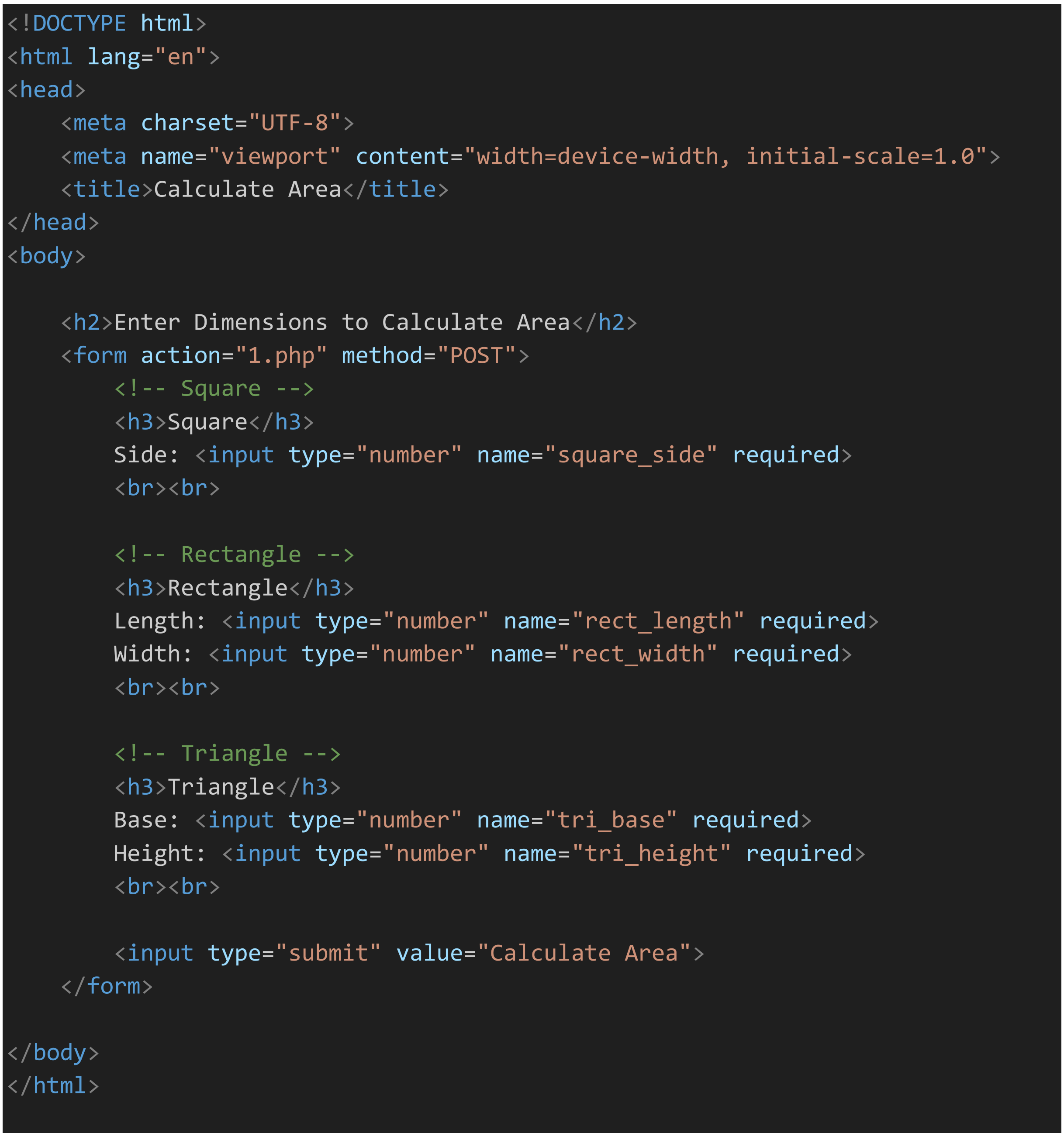


Book Details

Book Code	Book Name	Author	Year	Price
101	PHP Programming	John Doe	2020	₹500
102	Mastering JavaScript	Jane Smith	2019	₹650
103	Database Management	Michael Brown	2021	₹700

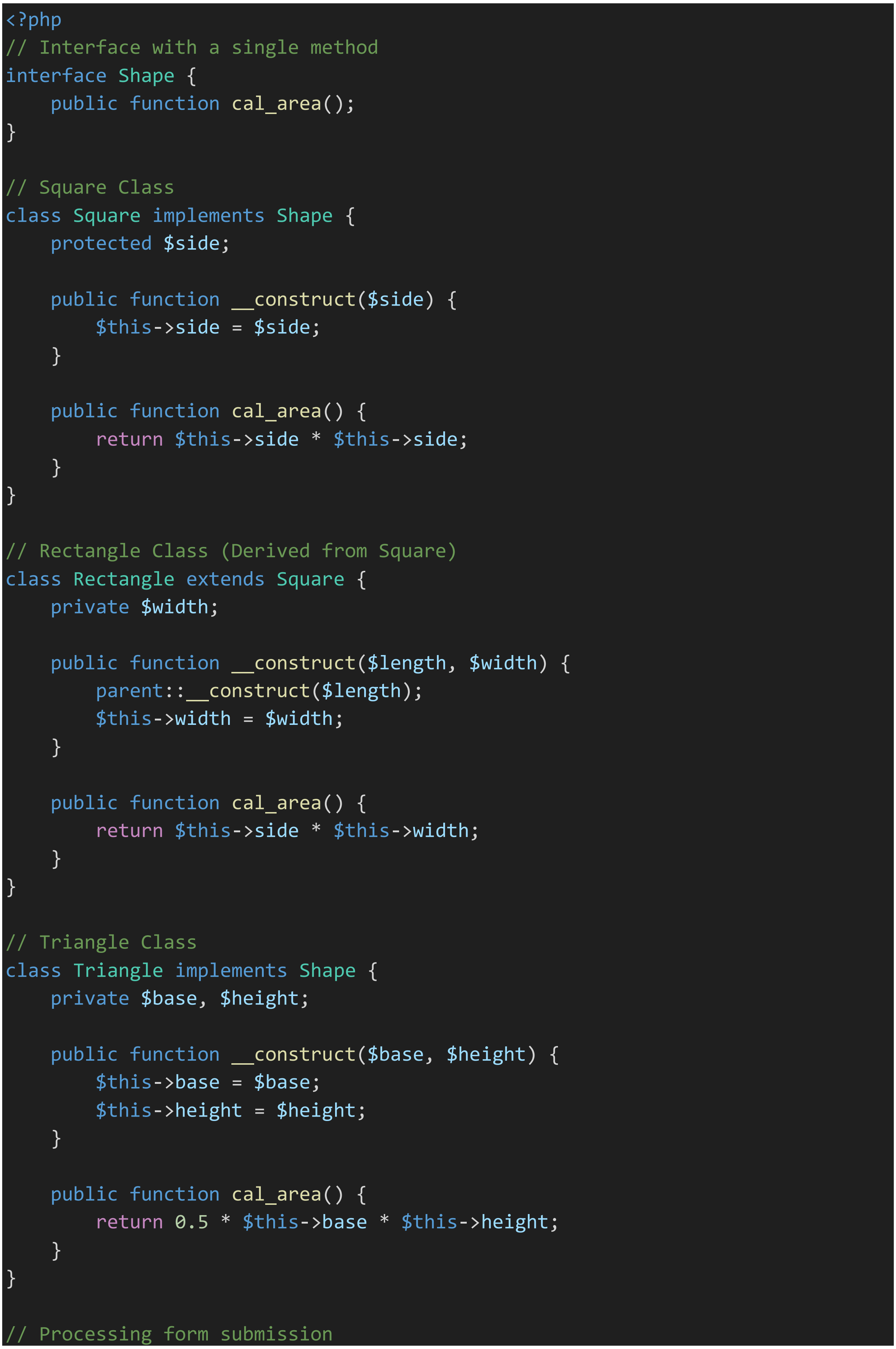
b) Derive a class Rectangle from class Square. Create one more class Triangle. Create an interface with only one method called cal_area(). Implement this interface in all the classes. Include appropriate data members and constructors in all classes. Write a program to accept details of Rectangle, Square and Triangle and display the area.

[Marks 25]

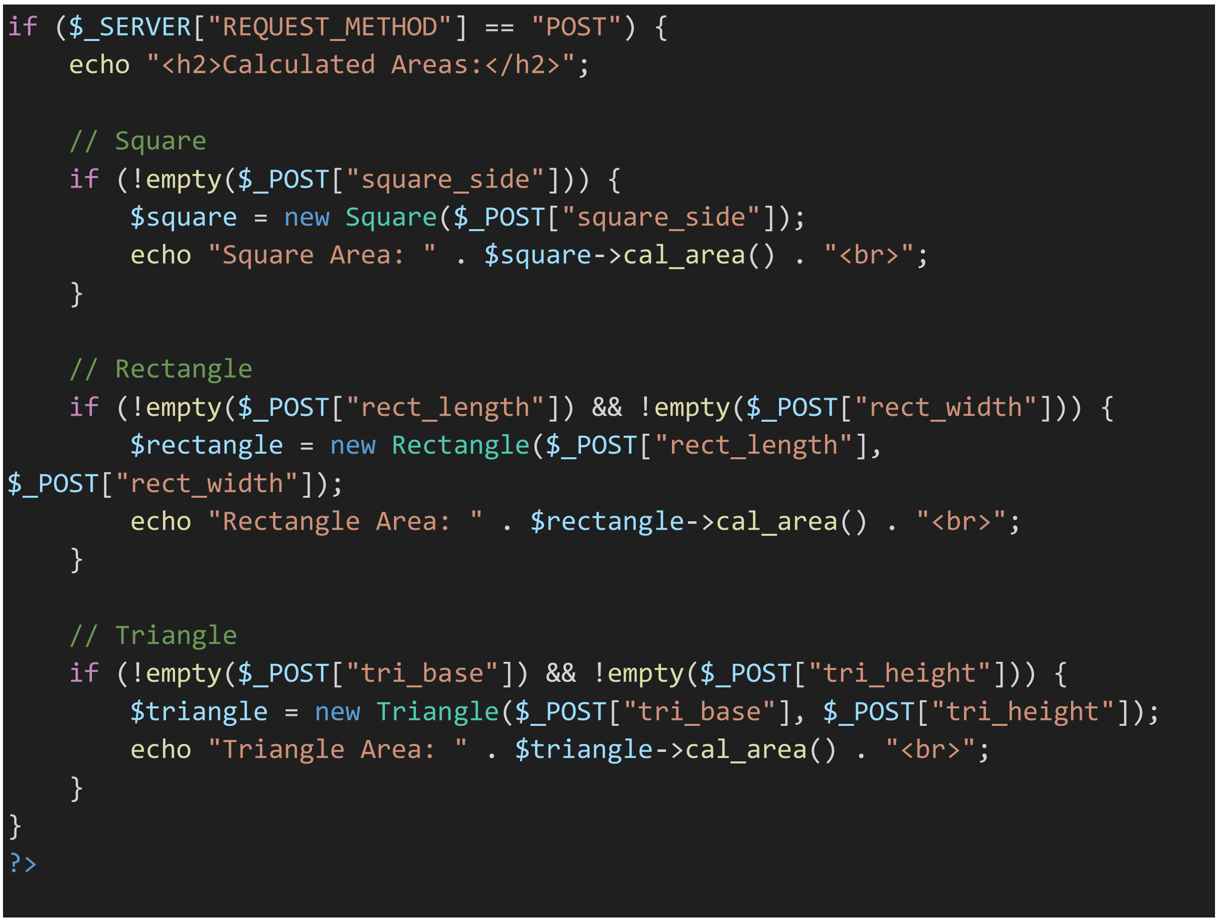


```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Calculate Area</title>
</head>
<body>

    <h2>Enter Dimensions to Calculate Area</h2>
    <form action="1.php" method="POST">
        <!-- Square -->
        <h3>Square</h3>
        Side: <input type="number" name="square_side" required>
        <br><br>

        <!-- Rectangle -->
        <h3>Rectangle</h3>
        Length: <input type="number" name="rect_length" required>
        Width: <input type="number" name="rect_width" required>
        <br><br>

        <!-- Triangle -->
        <h3>Triangle</h3>
        Base: <input type="number" name="tri_base" required>
        Height: <input type="number" name="tri_height" required>
        <br><br>

        <input type="submit" value="Calculate Area">
    </form>

</body>
</html>
```

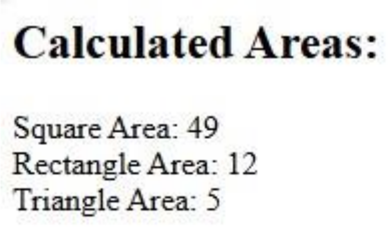
PHP File:-



```php
<?php
// Interface with a single method
interface Shape {
    public function cal_area();
}

// Square Class
class Square implements Shape {
    protected $side;

    public function __construct($side) {
        $this->side = $side;
    }

    public function cal_area() {
        return $this->side * $this->side;
    }
}

// Rectangle Class (Derived from Square)
class Rectangle extends Square {
    private $width;

    public function __construct($length, $width) {
        parent::__construct($length);
        $this->width = $width;
    }

    public function cal_area() {
        return $this->side * $this->width;
    }
}

// Triangle Class
class Triangle implements Shape {
    private $base, $height;

    public function __construct($base, $height) {
        $this->base = $base;
        $this->height = $height;
    }

    public function cal_area() {
        return 0.5 * $this->base * $this->height;
    }
}

// Processing form submission
```



```
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    echo "<h2>Calculated Areas:</h2>";

    // Square
    if (!empty($_POST["square_side"])) {
        $square = new Square($_POST["square_side"]);
        echo "Square Area: " . $square->cal_area() . "<br>";
    }

    // Rectangle
    if (!empty($_POST["rect_length"]) && !empty($_POST["rect_width"])) {
        $rectangle = new Rectangle($_POST["rect_length"],
$_POST["rect_width"]);
        echo "Rectangle Area: " . $rectangle->cal_area() . "<br>";
    }

    // Triangle
    if (!empty($_POST["tri_base"]) && !empty($_POST["tri_height"])) {
        $triangle = new Triangle($_POST["tri_base"], $_POST["tri_height"]);
        echo "Triangle Area: " . $triangle->cal_area() . "<br>";
    }
}
?>
```



```
Calculated Areas:

Square Area: 49
Rectangle Area: 12
Triangle Area: 5
```

Q. 3) Lab Book **[Marks 10]**

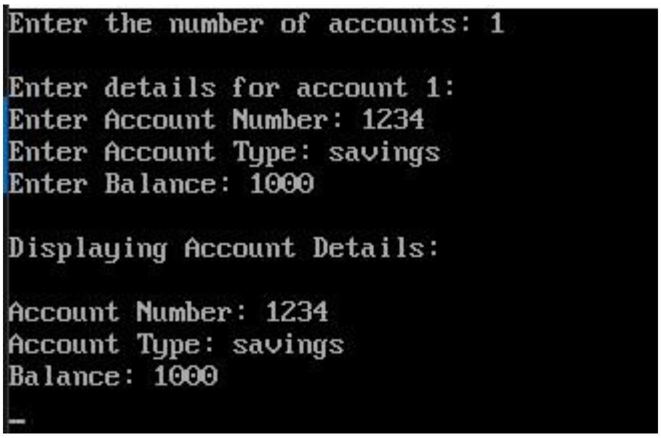
Q. 4) Viva **[Marks 10]**

Slip -12

Q. 1)

a) Write a c++ program to create a class Account with data members AccNo, AccType and Balance.write member function to accept and display ’n’ account details (Use dynamic memory allocation) [Marks 15]

```
#include <iostream.h>
#include <conio.h>


class Account {
private:
    int Acc_number;
    char Acc_type[50];
    double Balance;

public:
    void acceptDetails() {
        cout << "Enter Account Number: ";
        cin >> Acc_number;
        cout << "Enter Account Type: ";
        cin >> Acc_type;
        cout << "Enter Balance: ";
        cin >> Balance;
    }

    void displayDetails() {
        cout << "\nAccount Number: " << Acc_number;
        cout << "\nAccount Type: " << Acc_type;
        cout << "\nBalance: " << Balance << "\n";
    }
};

int main() {
    clrscr();
    int n;
    cout << "Enter the number of accounts: ";
    cin >> n;

    Account *accounts = new Account[n];

    for (int i = 0; i < n; i++) {
        cout << "\nEnter details for account " << i + 1 << ":\n";
        accounts[i].acceptDetails();
    }

    cout << "\nDisplaying Account Details:\n";
    for (int i = 0; i < n; i++) {
        accounts[i].displayDetails();
    }

    getch();
    delete[] accounts; // Free allocated memory
    return 0;
}
```



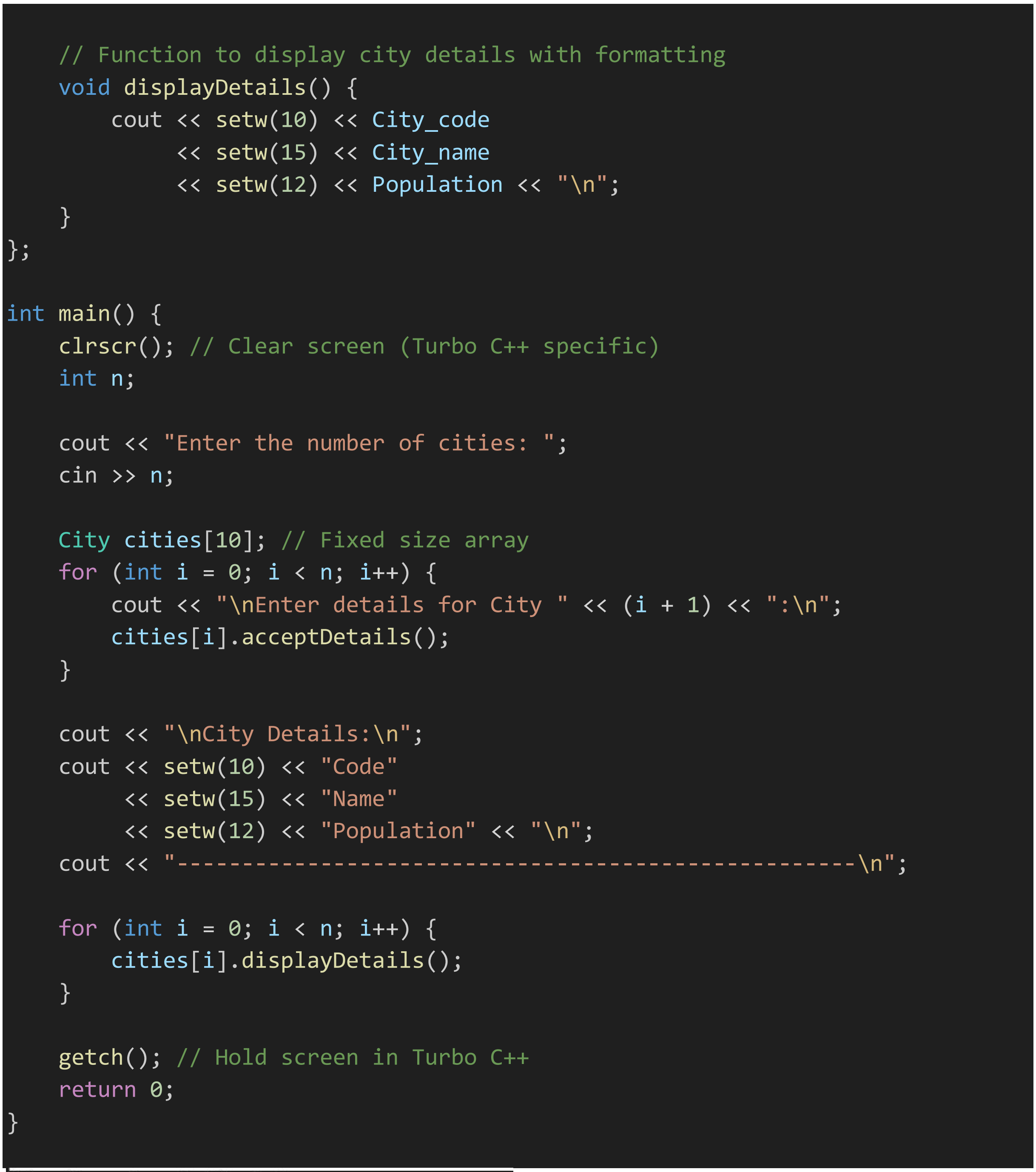
```
Enter the number of accounts: 1

Enter details for account 1:
Enter Account Number: 1234
Enter Account Type: savings
Enter Balance: 1000

Displaying Account Details:

Account Number: 1234
Account Type: savings
Balance: 1000
_
```

b) Create a C++ class city with data member City_code, City_name, Population. Write necessary member functions for following.

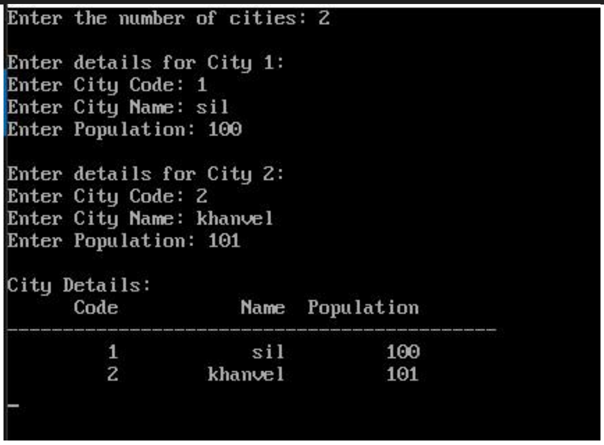
1. To accept details of n cities

2. Display Details of n cities (Use Array of Object and manipulators) [Marks 25]

```
#include <iostream.h>
#include <conio.h>
#include <iomanip.h> // For setw()

class City {
public:
    int City_code;
    char City_name[30];
    long Population;

    // Function to accept city details
    void acceptDetails() {
        cout << "Enter City Code: ";
        cin >> City_code;
        cout << "Enter City Name: ";
        cin >> City_name;
        cout << "Enter Population: ";
        cin >> Population;
    }
```



```cpp
    // Function to display city details with formatting
    void displayDetails() {
        cout << setw(10) << City_code
             << setw(15) << City_name
             << setw(12) << Population << "\n";
    }
};

int main() {
    clrscr(); // Clear screen (Turbo C++ specific)
    int n;

    cout << "Enter the number of cities: ";
    cin >> n;

    City cities[10]; // Fixed size array
    for (int i = 0; i < n; i++) {
        cout << "\nEnter details for City " << (i + 1) << ":\n";
        cities[i].acceptDetails();
    }

    cout << "\nCity Details:\n";
    cout << setw(10) << "Code"
         << setw(15) << "Name"
         << setw(12) << "Population" << "\n";
    cout << "----------------------------------------------------\n";

    for (int i = 0; i < n; i++) {
        cities[i].displayDetails();
    }

    getch(); // Hold screen in Turbo C++
    return 0;
}
```

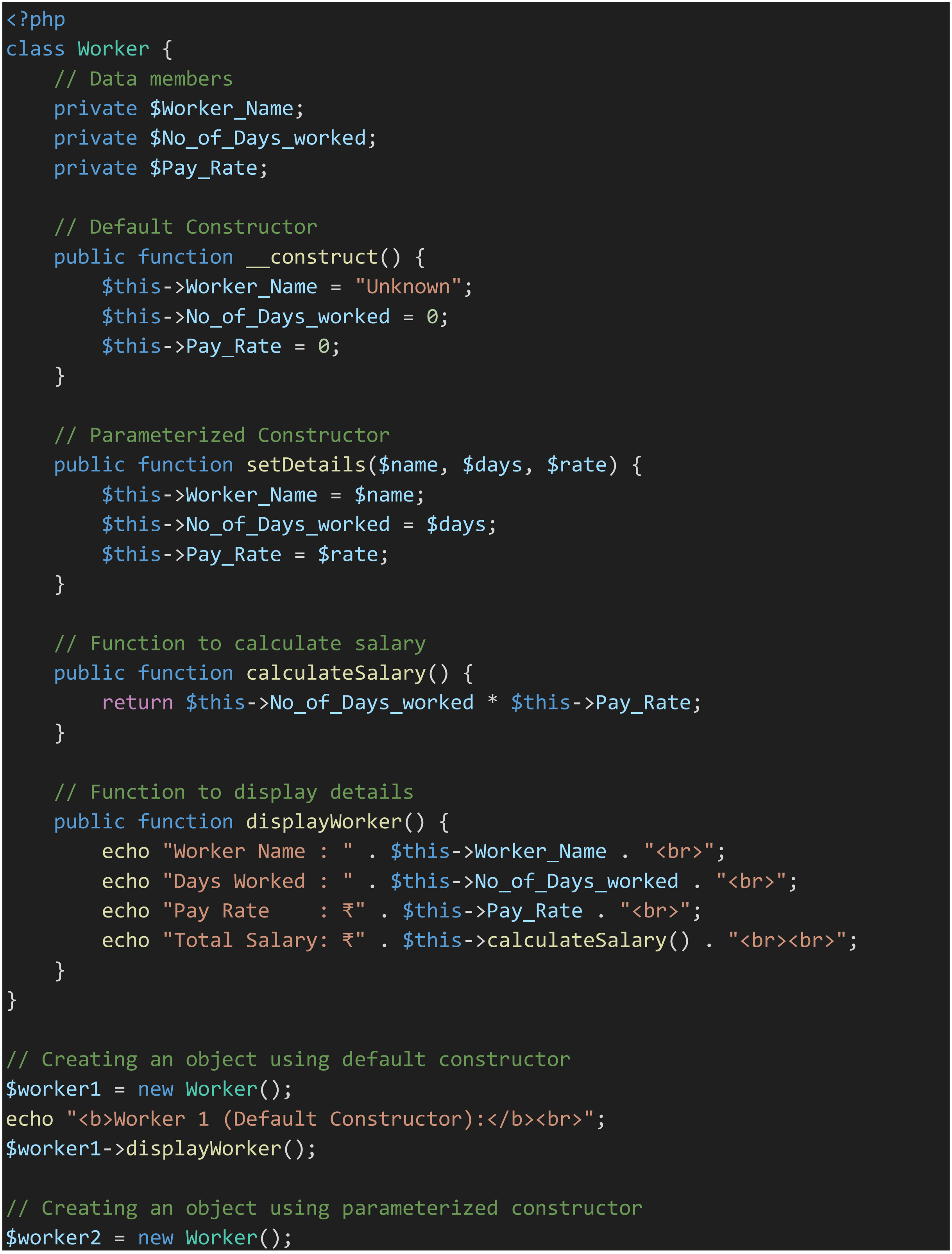


```
Enter the number of cities: 2

Enter details for City 1:
Enter City Code: 1
Enter City Name: sil
Enter Population: 100

Enter details for City 2:
Enter City Code: 2
Enter City Name: khanvel
Enter Population: 101

City Details:
      Code           Name  Population
--------------------------------------------
         1            sil         100
         2        khanvel         101
_
```

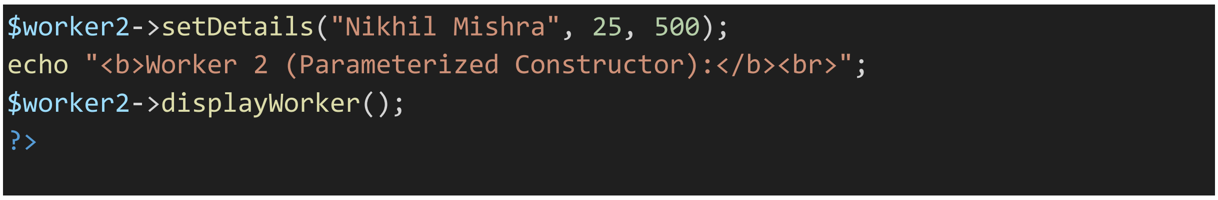
Q. 2)

a) Write a PHP program to create a class Worker that has data members as Worker_Name, No_of_Days_worked, Pay_Rate. Create and initialize the object using default constructor, Parameterized constructor. Also write necessary member function to calculate and display the salary of worker.

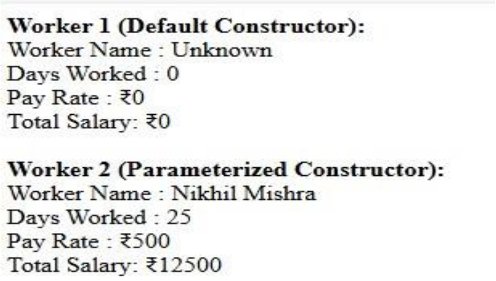
[Marks l5]



```php
<?php
class Worker {
    // Data members
    private $Worker_Name;
    private $No_of_Days_worked;
    private $Pay_Rate;

    // Default Constructor
    public function __construct() {
        $this->Worker_Name = "Unknown";
        $this->No_of_Days_worked = 0;
        $this->Pay_Rate = 0;
    }

    // Parameterized Constructor
    public function setDetails($name, $days, $rate) {
        $this->Worker_Name = $name;
        $this->No_of_Days_worked = $days;
        $this->Pay_Rate = $rate;
    }

    // Function to calculate salary
    public function calculateSalary() {
        return $this->No_of_Days_worked * $this->Pay_Rate;
    }

    // Function to display details
    public function displayWorker() {
        echo "Worker Name : " . $this->Worker_Name . "<br>";
        echo "Days Worked : " . $this->No_of_Days_worked . "<br>";
        echo "Pay Rate    : ₹" . $this->Pay_Rate . "<br>";
        echo "Total Salary: ₹" . $this->calculateSalary() . "<br><br>";
    }
}

// Creating an object using default constructor
$worker1 = new Worker();
echo "<b>Worker 1 (Default Constructor):</b><br>";
$worker1->displayWorker();

// Creating an object using parameterized constructor
$worker2 = new Worker();
```



```
$worker2->setDetails("Nikhil Mishra", 25, 500);
echo "<b>Worker 2 (Parameterized Constructor):</b><br>";
$worker2->displayWorker();
?>
```



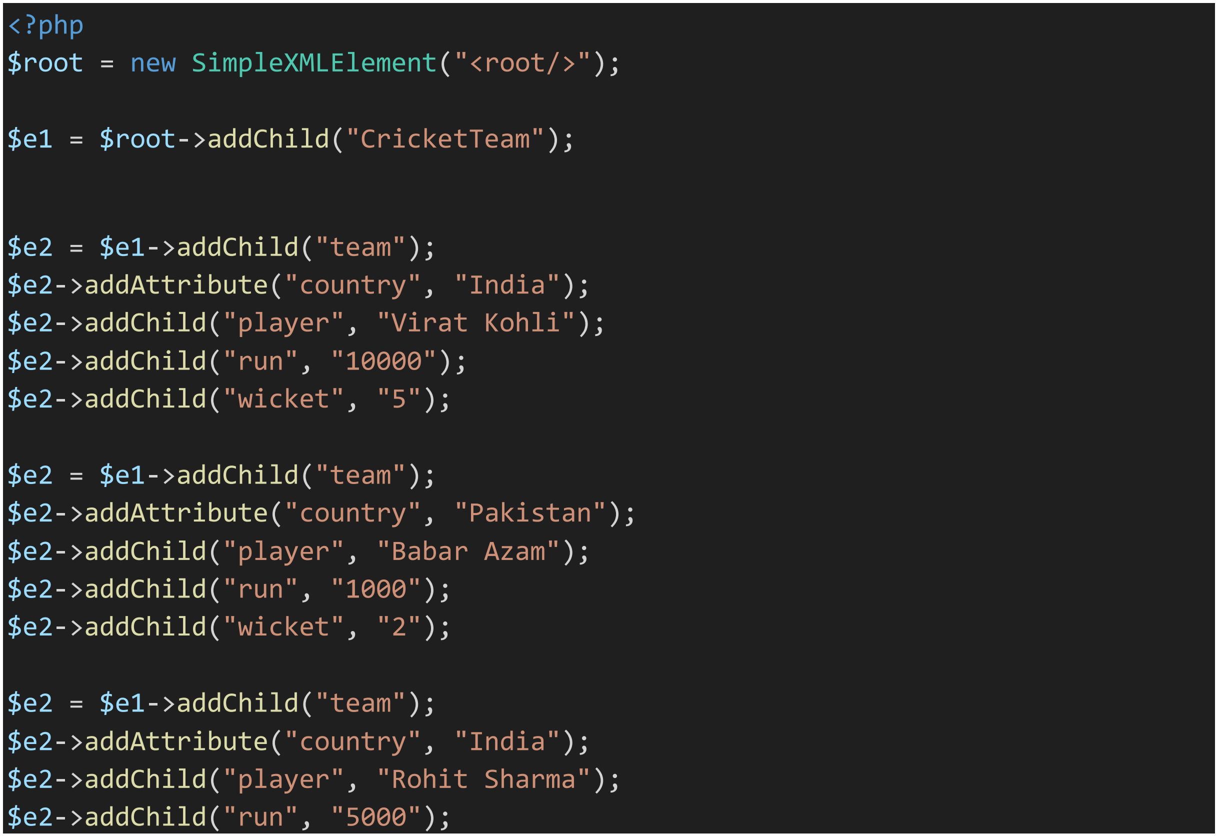
```
Worker 1 (Default Constructor):
Worker Name : Unknown
Days Worked : 0
Pay Rate : ₹0
Total Salary: ₹0

Worker 2 (Parameterized Constructor):
Worker Name : Nikhil Mishra
Days Worked : 25
Pay Rate : ₹500
Total Salary: ₹12500
```

b) Write a script to create “cricket.xml” file with multiple elements as shown below:
<CricketTeam>

```
 <Team country=”India”>
        <player>          </player>
         <runs>            </runs>
         <wicket>          </wicket>
  </Team>
  </CricketTeam>
```

Write a script to add multiple elements in “cricket.xml” file of category, country=”Australia”.

[Marks 25]



```
<?php
$root = new SimpleXMLElement("<root/>");

$e1 = $root->addChild("CricketTeam");


$e2 = $e1->addChild("team");
$e2->addAttribute("country", "India");
$e2->addChild("player", "Virat Kohli");
$e2->addChild("run", "10000");
$e2->addChild("wicket", "5");

$e2 = $e1->addChild("team");
$e2->addAttribute("country", "Pakistan");
$e2->addChild("player", "Babar Azam");
$e2->addChild("run", "1000");
$e2->addChild("wicket", "2");

$e2 = $e1->addChild("team");
$e2->addAttribute("country", "India");
$e2->addChild("player", "Rohit Sharma");
$e2->addChild("run", "5000");
```

```
$e2->addChild("wicket", "4");


$root->asXML("cricket.xml");

?>
```

**** Output :- New XML file named cricket.xml will be created ****

Q. 3) Lab Book **[Marks 10]**

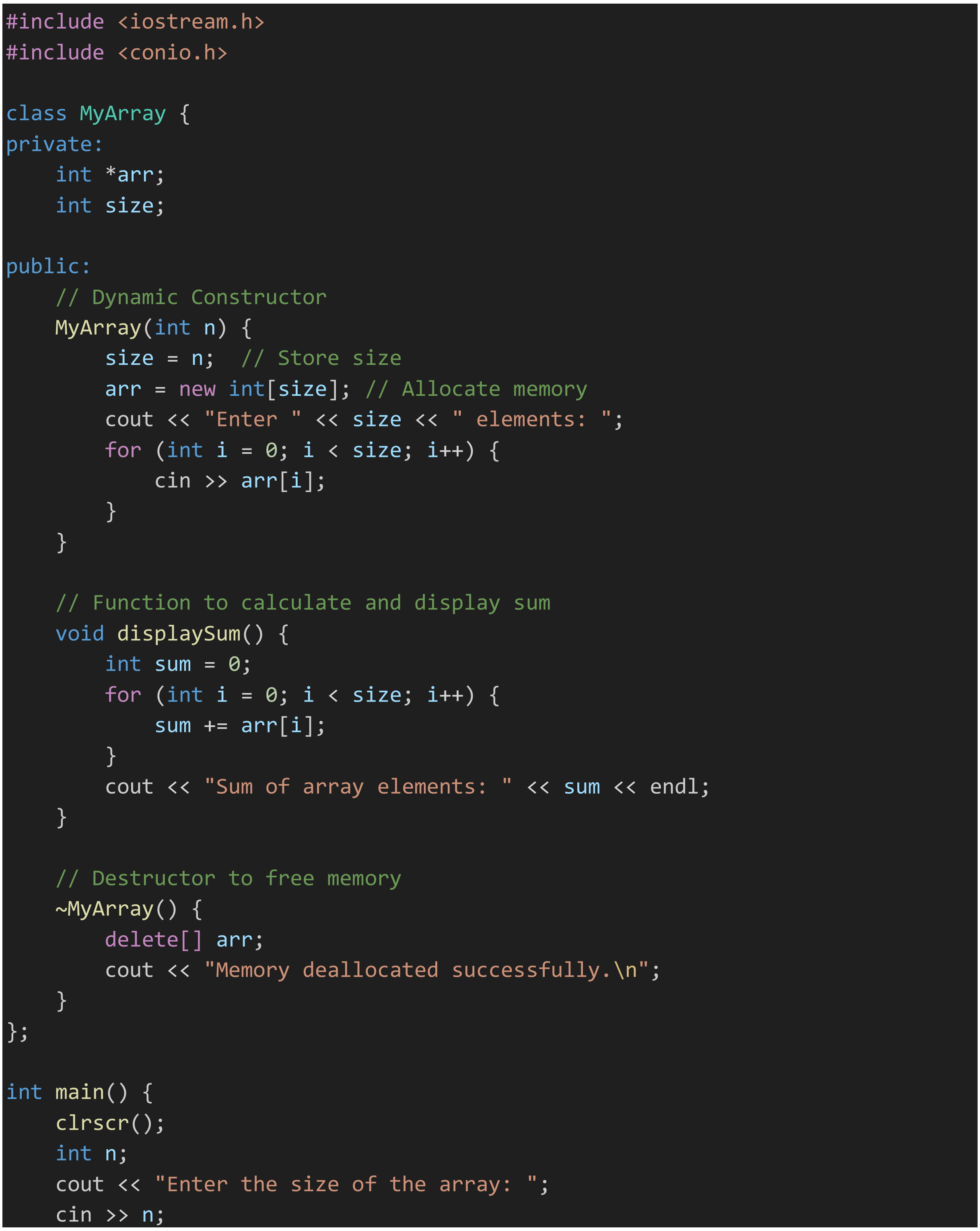
Q. 4) Viva **[Marks 10]**

Slip -13

Q. 1)

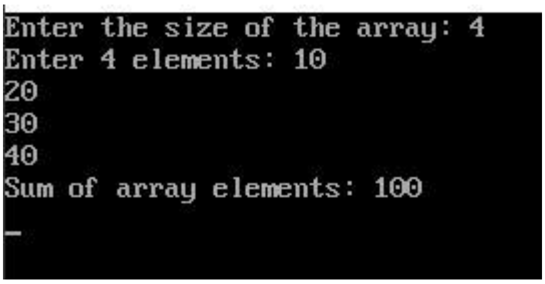
a) Create a C++ class MyArray, which contains single dimensional integer array of a given size. Write a member function to display sum of given array elements. (Use Dynamic Constructor and Destructor)

[Marks 15]



```
#include <iostream.h>
#include <conio.h>

class MyArray {
private:
    int *arr;
    int size;

public:
    // Dynamic Constructor
    MyArray(int n) {
        size = n;  // Store size
        arr = new int[size]; // Allocate memory
        cout << "Enter " << size << " elements: ";
        for (int i = 0; i < size; i++) {
            cin >> arr[i];
        }
    }

    // Function to calculate and display sum
    void displaySum() {
        int sum = 0;
        for (int i = 0; i < size; i++) {
            sum += arr[i];
        }
        cout << "Sum of array elements: " << sum << endl;
    }

    // Destructor to free memory
    ~MyArray() {
        delete[] arr;
        cout << "Memory deallocated successfully.\n";
    }
};

int main() {
    clrscr();
    int n;
    cout << "Enter the size of the array: ";
    cin >> n;
```

```
   if(n>100){
    cout<<"Size is too large! Enter a Value <=100.";
   }else{
    MyArray *obj = new MyArray(n);
    obj->displaysum();
    delete obj;
   }

    getch();
    return 0; // Destructor is automatically called when obj goes out of scope
}
```

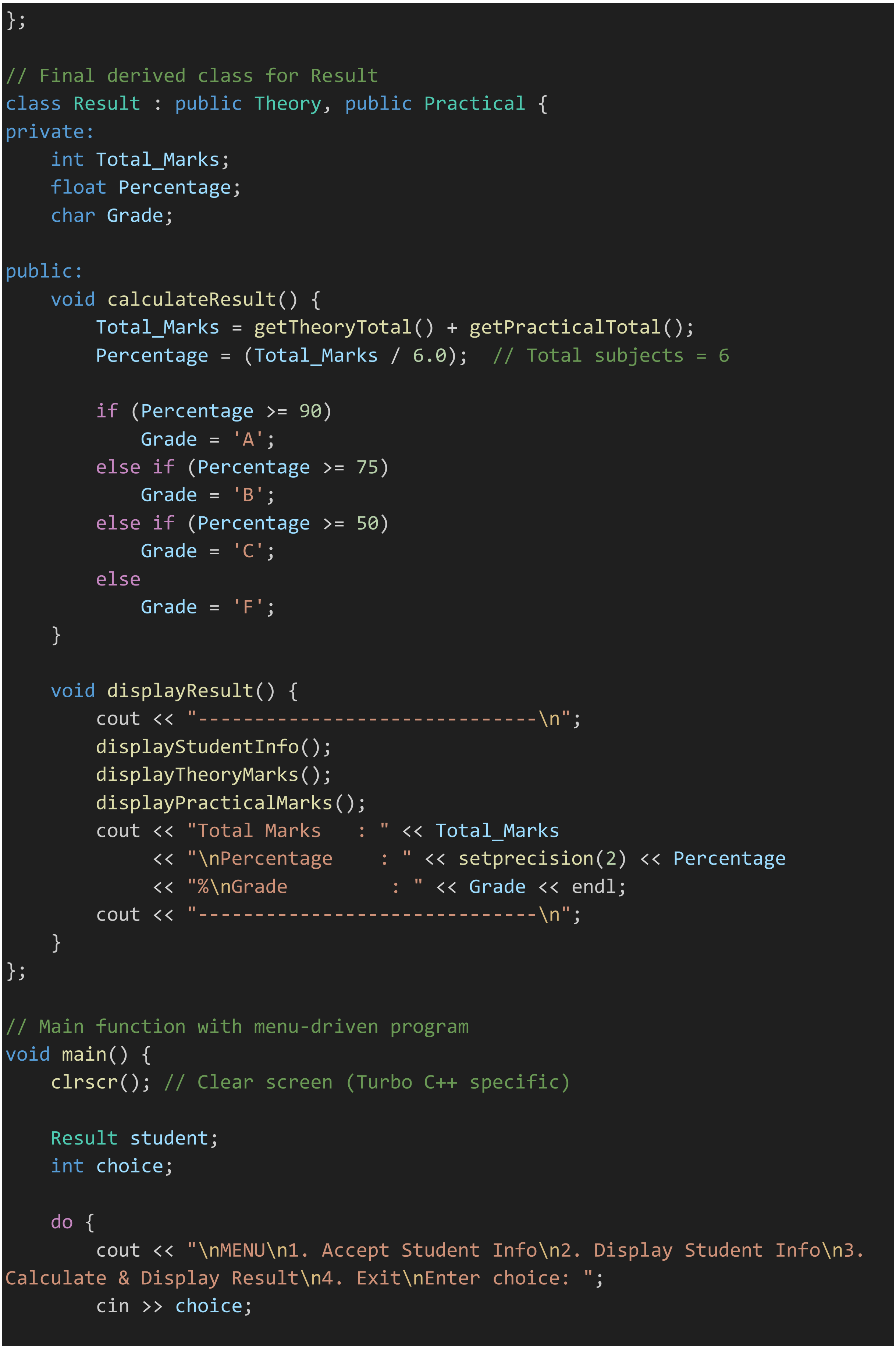


```
Enter the size of the array: 4
Enter 4 elements: 10
20
30
40
Sum of array elements: 100
_
```

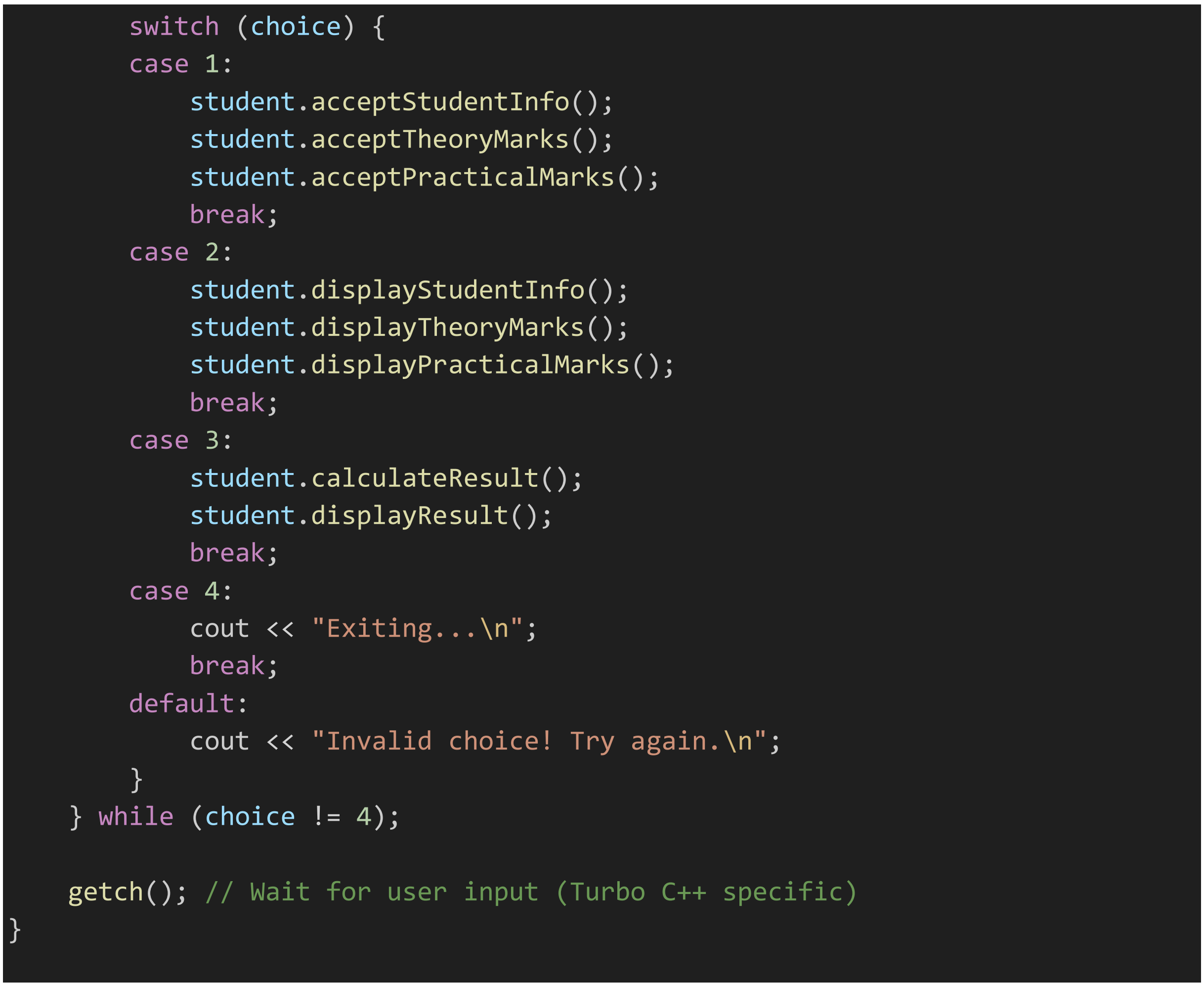
b) Create a base class Student with data members Roll No, Name. Derives two classes from it, class Theory with data members M 1, M2, M3, M4 and class Practical w ith data members P1, P2. Class Result (Total Marks, Percentage, Grade) inherits both Theory and Practical classes. (Use concept of Virtual Base Class and protected access specifiers) Write a C++ menu driven program to perform the following functions:

i. Accept Student Information
ii. Display Student Information
iii. Calculate Total marks, Percentage and Grade. [Marks 25]

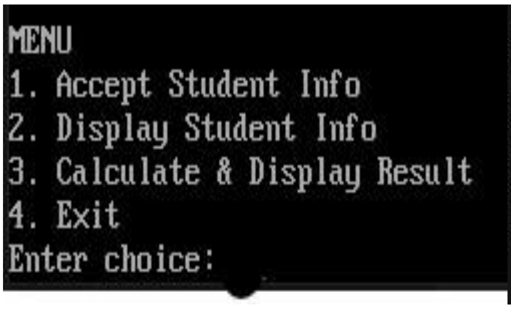
```
#include <iostream.h>
#include <conio.h>       // For clrscr() and getch()
#include <iomanip.h>     // For formatting output

// Base class
class Student {
protected:
    int Roll_No;
    char Name[50];

public:
    void acceptStudentInfo() {
        cout << "Enter Roll Number: ";
        cin >> Roll_No;
        cout << "Enter Name: ";
        cin.ignore();
        cin.getline(Name, 50);
    }

    void displayStudentInfo() {
        cout << "Roll No: " << Roll_No << "\nName: " << Name << endl;
    }
};

// Derived class for Theory marks
class Theory : virtual public Student {
protected:
    int M1, M2, M3, M4;

public:
    void acceptTheoryMarks() {
        cout << "Enter Theory Marks (M1, M2, M3, M4): ";
        cin >> M1 >> M2 >> M3 >> M4;
    }

    void displayTheoryMarks() {
        cout << "Theory Marks: " << M1 << ", " << M2 << ", " << M3 << ", " << M4 << endl;
    }

    int getTheoryTotal() {
        return M1 + M2 + M3 + M4;
    }
};

// Derived class for Practical marks
class Practical : virtual public Student {
protected:
    int P1, P2;

public:
    void acceptPracticalMarks() {
        cout << "Enter Practical Marks (P1, P2): ";
        cin >> P1 >> P2;
    }

    void displayPracticalMarks() {
        cout << "Practical Marks: " << P1 << ", " << P2 << endl;
    }

    int getPracticalTotal() {
        return P1 + P2;
    }
```



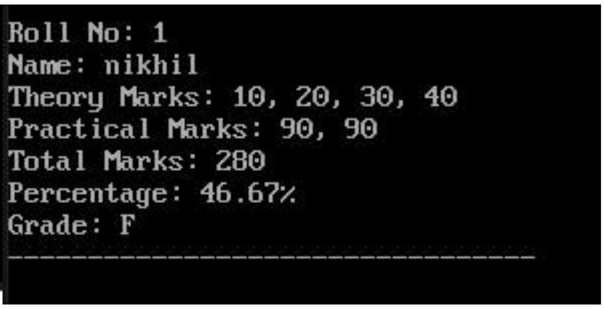
```
};

// Final derived class for Result
class Result : public Theory, public Practical {
private:
    int Total_Marks;
    float Percentage;
    char Grade;

public:
    void calculateResult() {
        Total_Marks = getTheoryTotal() + getPracticalTotal();
        Percentage = (Total_Marks / 6.0);  // Total subjects = 6

        if (Percentage >= 90)
            Grade = 'A';
        else if (Percentage >= 75)
            Grade = 'B';
        else if (Percentage >= 50)
            Grade = 'C';
        else
            Grade = 'F';
    }

    void displayResult() {
        cout << "------------------------------\n";
        displayStudentInfo();
        displayTheoryMarks();
        displayPracticalMarks();
        cout << "Total Marks   : " << Total_Marks
             << "\nPercentage    : " << setprecision(2) << Percentage
             << "%\nGrade         : " << Grade << endl;
        cout << "------------------------------\n";
    }
};

// Main function with menu-driven program
void main() {
    clrscr(); // Clear screen (Turbo C++ specific)

    Result student;
    int choice;

    do {
        cout << "\nMENU\n1. Accept Student Info\n2. Display Student Info\n3.
Calculate & Display Result\n4. Exit\nEnter choice: ";
        cin >> choice;
```



```
        switch (choice) {
        case 1:
            student.acceptStudentInfo();
            student.acceptTheoryMarks();
            student.acceptPracticalMarks();
            break;
        case 2:
            student.displayStudentInfo();
            student.displayTheoryMarks();
            student.displayPracticalMarks();
            break;
        case 3:
            student.calculateResult();
            student.displayResult();
            break;
        case 4:
            cout << "Exiting...\n";
            break;
        default:
            cout << "Invalid choice! Try again.\n";
        }
    } while (choice != 4);

    getch(); // Wait for user input (Turbo C++ specific)
}
```



```
MENU
1. Accept Student Info
2. Display Student Info
3. Calculate & Display Result
4. Exit
Enter choice:
```

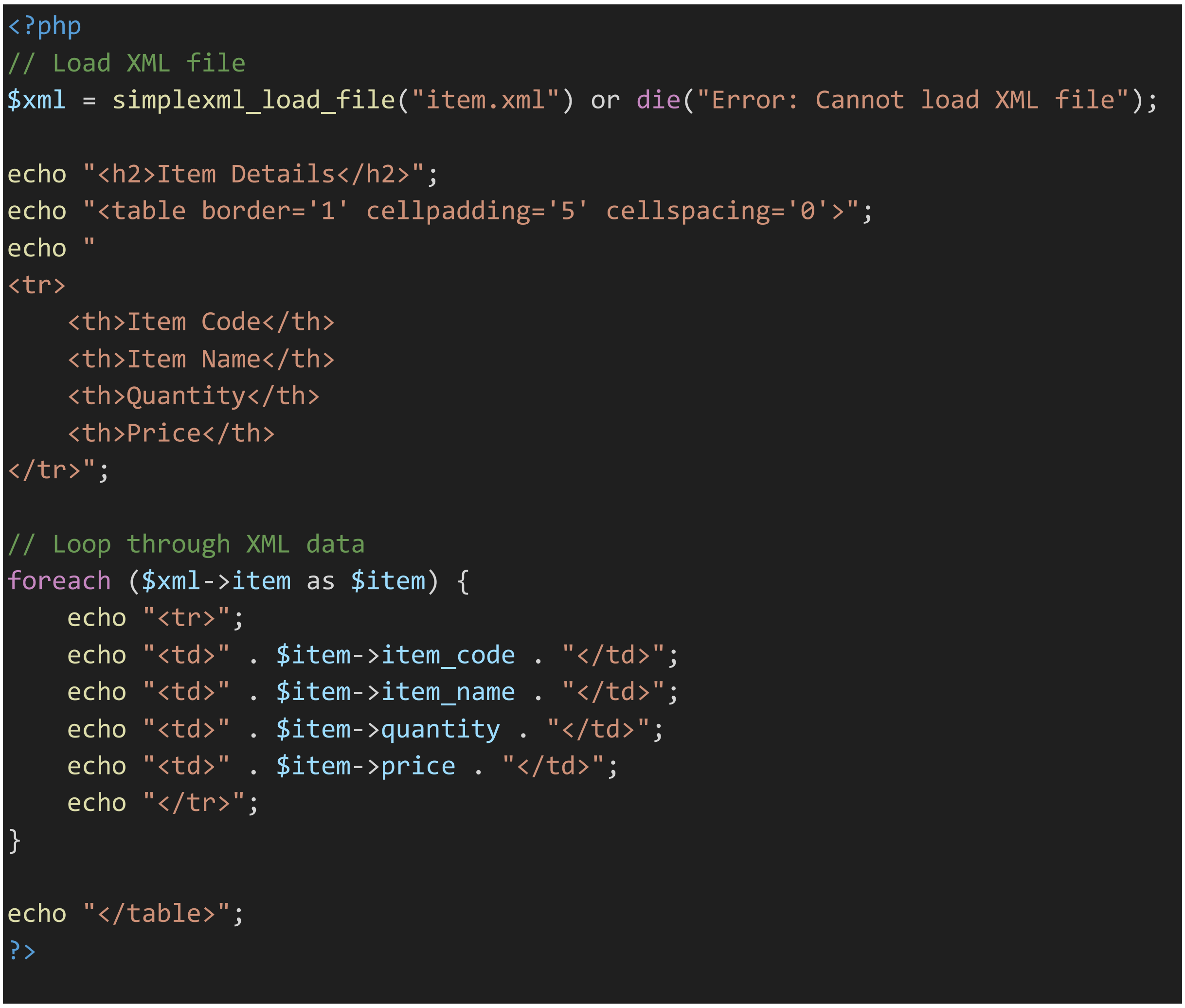


```
Roll No: 1
Name: nikhil
Theory Marks: 10, 20, 30, 40
Practical Marks: 90, 90
Total Marks: 280
Percentage: 46.67%
Grade: F
------------------------------------
```

Q. 2)

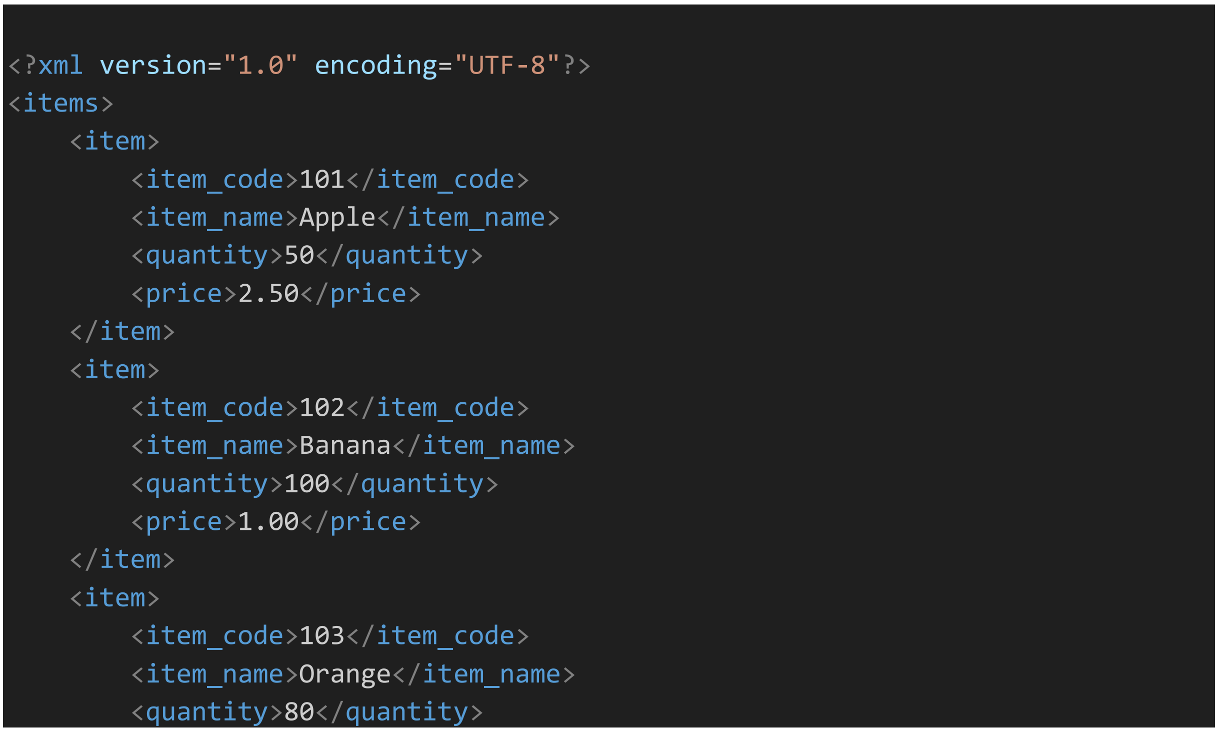
a) Write a PHP script to read item.XML and print item details in tabular format using simple XML(Content of book. XML are item_code, item_name, quantity, price) [Marks l5]

PHP File:-

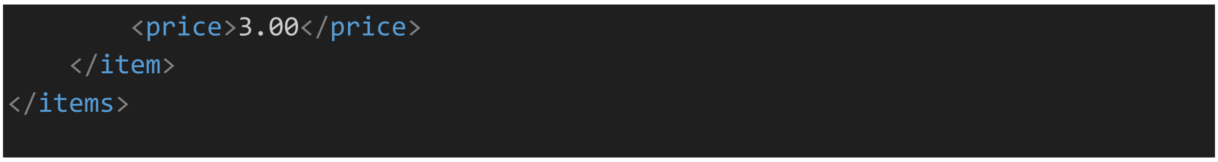


```php
<?php
// Load XML file
$xml = simplexml_load_file("item.xml") or die("Error: Cannot load XML file");

echo "<h2>Item Details</h2>";
echo "<table border='1' cellpadding='5' cellspacing='0'>";
echo "
<tr>
    <th>Item Code</th>
    <th>Item Name</th>
    <th>Quantity</th>
    <th>Price</th>
</tr>";

// Loop through XML data
foreach ($xml->item as $item) {
    echo "<tr>";
    echo "<td>" . $item->item_code . "</td>";
    echo "<td>" . $item->item_name . "</td>";
    echo "<td>" . $item->quantity . "</td>";
    echo "<td>" . $item->price . "</td>";
    echo "</tr>";
}

echo "</table>";
?>
```

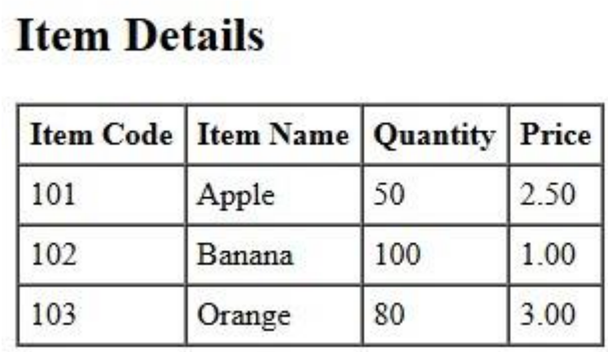
XML File:-



```xml
<?xml version="1.0" encoding="UTF-8"?>
<items>
    <item>
        <item_code>101</item_code>
        <item_name>Apple</item_name>
        <quantity>50</quantity>
        <price>2.50</price>
    </item>
    <item>
        <item_code>102</item_code>
        <item_name>Banana</item_name>
        <quantity>100</quantity>
        <price>1.00</price>
    </item>
    <item>
        <item_code>103</item_code>
        <item_name>Orange</item_name>
        <quantity>80</quantity>
```



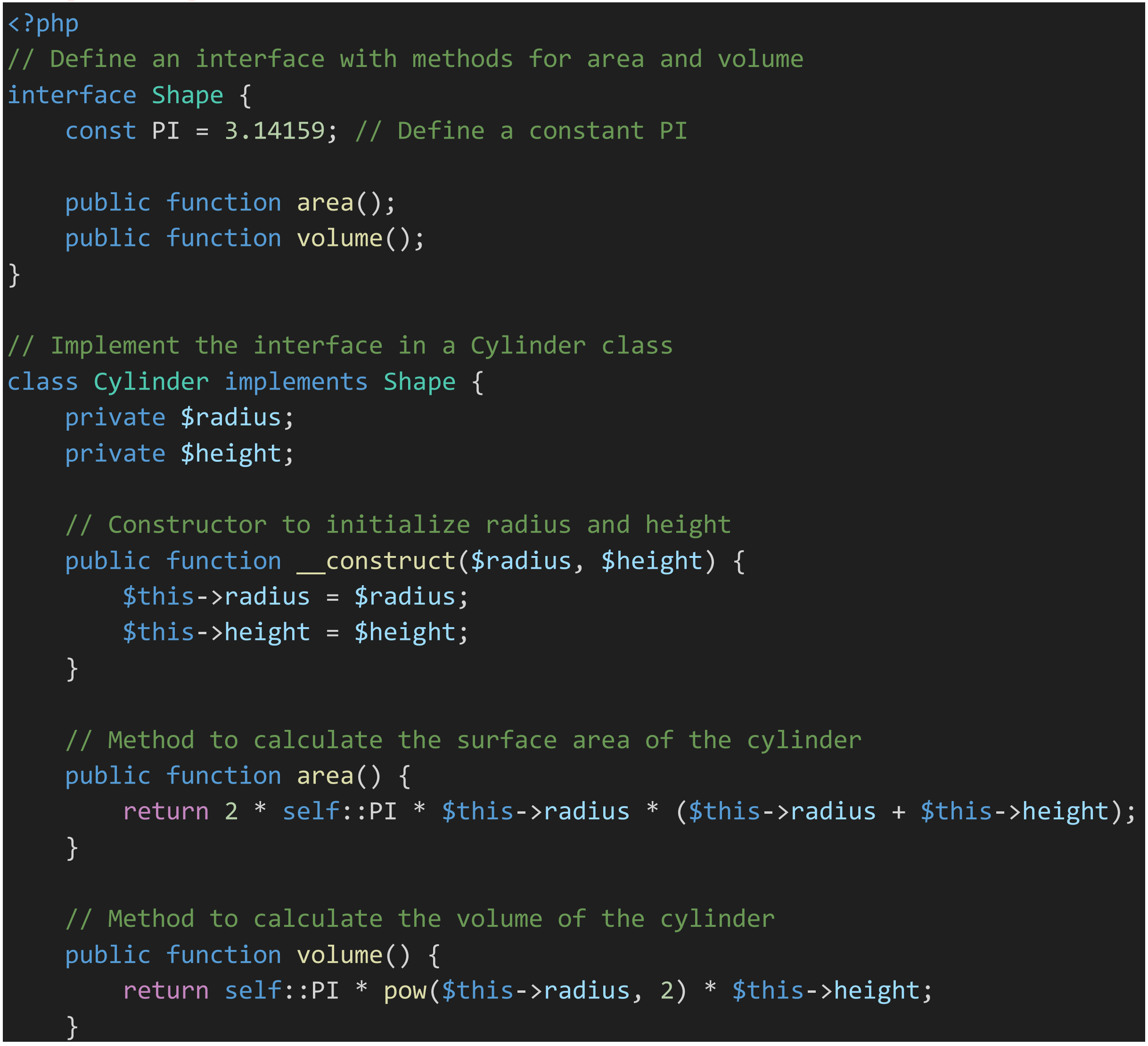
```
        <price>3.00</price>
    </item>
</items>
```



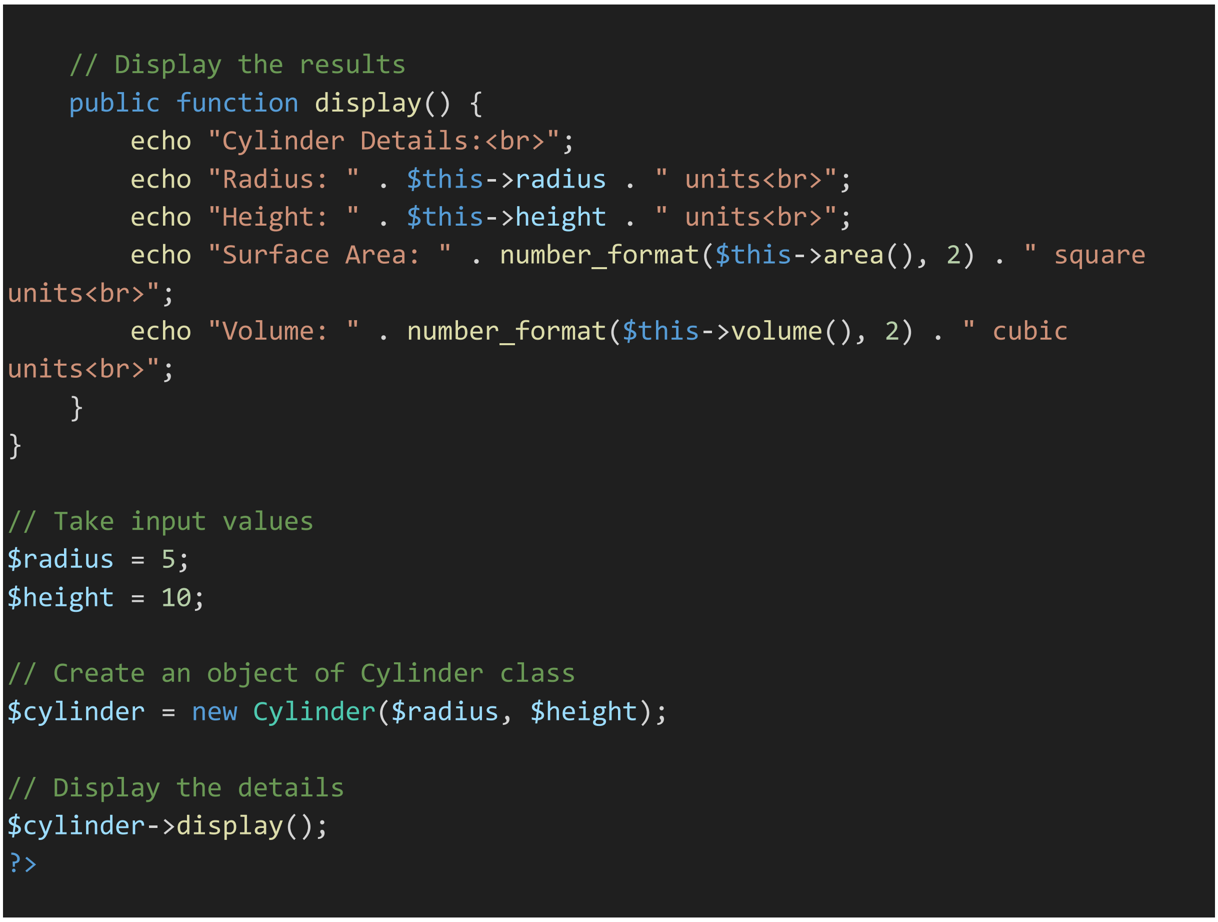
Item Details

Item Code	Item Name	Quantity	Price
101	Apple	50	2.50
102	Banana	100	1.00
103	Orange	80	3.00

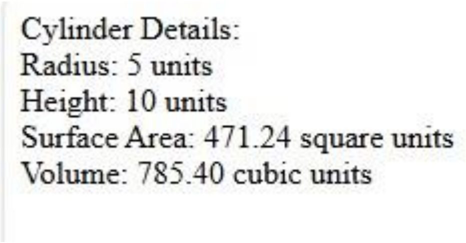
b) Define an interface which has methods area( ), volume( ). Define constant PI. Create a class cylinder which implements this interface and calculate area and volume. [Marks 25]



```
<?php
// Define an interface with methods for area and volume
interface Shape {
    const PI = 3.14159; // Define a constant PI

    public function area();
    public function volume();
}

// Implement the interface in a Cylinder class
class Cylinder implements Shape {
    private $radius;
    private $height;

    // Constructor to initialize radius and height
    public function __construct($radius, $height) {
        $this->radius = $radius;
        $this->height = $height;
    }

    // Method to calculate the surface area of the cylinder
    public function area() {
        return 2 * self::PI * $this->radius * ($this->radius + $this->height);
    }

    // Method to calculate the volume of the cylinder
    public function volume() {
        return self::PI * pow($this->radius, 2) * $this->height;
    }
```



```php
    // Display the results
    public function display() {
        echo "Cylinder Details:<br>";
        echo "Radius: " . $this->radius . " units<br>";
        echo "Height: " . $this->height . " units<br>";
        echo "Surface Area: " . number_format($this->area(), 2) . " square
units<br>";
        echo "Volume: " . number_format($this->volume(), 2) . " cubic
units<br>";
    }
}

// Take input values
$radius = 5;
$height = 10;

// Create an object of Cylinder class
$cylinder = new Cylinder($radius, $height);

// Display the details
$cylinder->display();
?>
```



```
Cylinder Details:
Radius: 5 units
Height: 10 units
Surface Area: 471.24 square units
Volume: 785.40 cubic units
```

Q. 3) Lab Book **[Marks 10]**

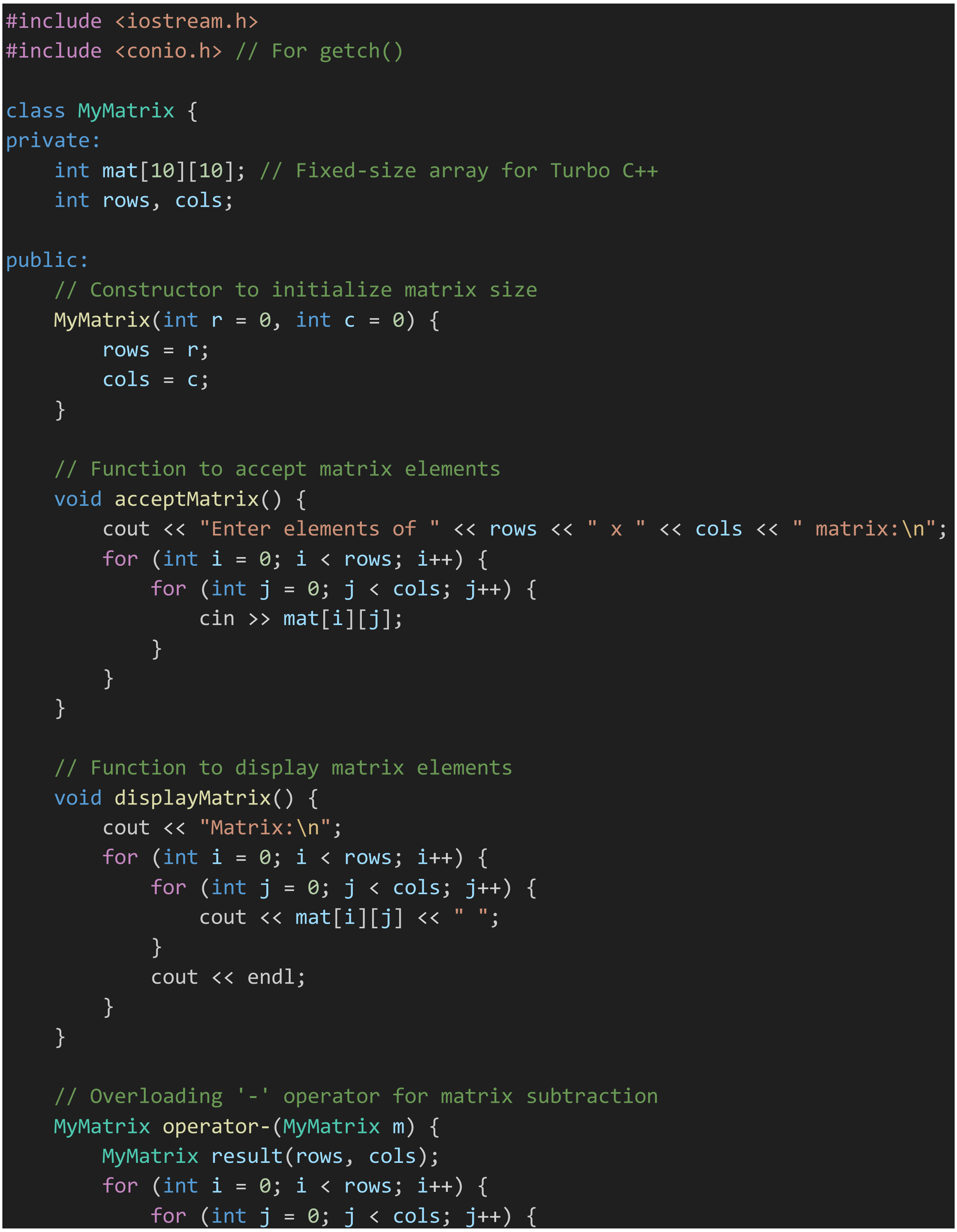
Q. 4) Viva **[Marks 10]**

Slip -14

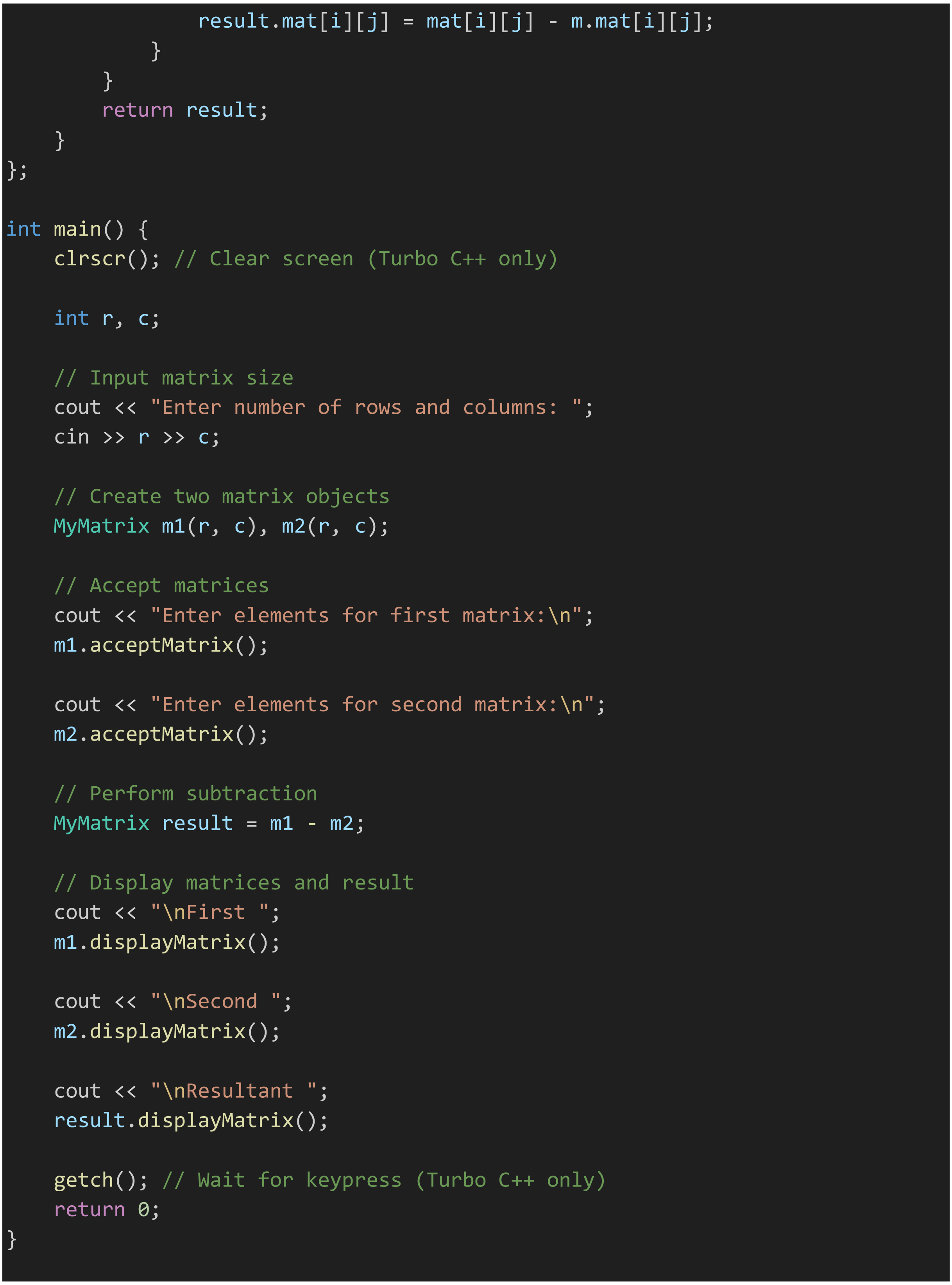
Q. 1)

a) Create a class Mymatrix. Write C++ program to accept and display a Matrix overload binary '-' operator to calculate substraction of two matrix.

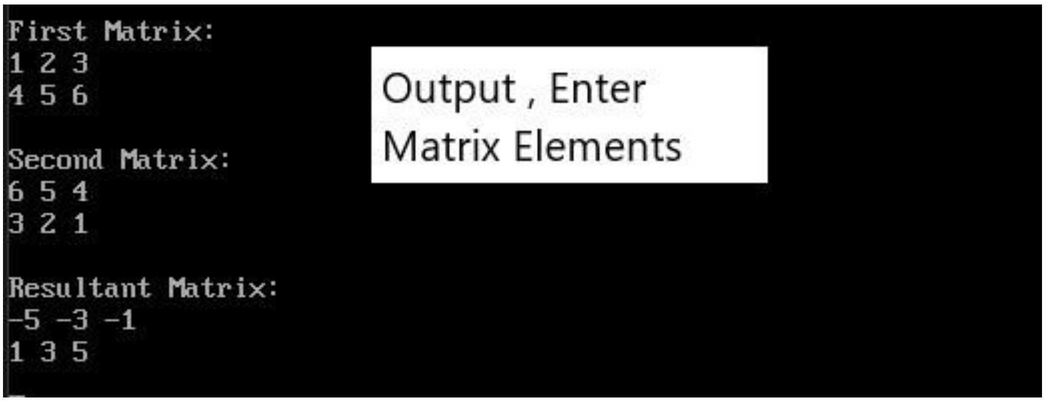
[Marks 15]



```
#include <iostream.h>
#include <conio.h> // For getch()

class MyMatrix {
private:
    int mat[10][10]; // Fixed-size array for Turbo C++
    int rows, cols;

public:
    // Constructor to initialize matrix size
    MyMatrix(int r = 0, int c = 0) {
        rows = r;
        cols = c;
    }

    // Function to accept matrix elements
    void acceptMatrix() {
        cout << "Enter elements of " << rows << " x " << cols << " matrix:\n";
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                cin >> mat[i][j];
            }
        }
    }

    // Function to display matrix elements
    void displayMatrix() {
        cout << "Matrix:\n";
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                cout << mat[i][j] << " ";
            }
            cout << endl;
        }
    }

    // Overloading '-' operator for matrix subtraction
    MyMatrix operator-(MyMatrix m) {
        MyMatrix result(rows, cols);
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
```



```
                result.mat[i][j] = mat[i][j] - m.mat[i][j];
            }
        }
        return result;
    }
};

int main() {
    clrscr(); // Clear screen (Turbo C++ only)

    int r, c;

    // Input matrix size
    cout << "Enter number of rows and columns: ";
    cin >> r >> c;

    // Create two matrix objects
    MyMatrix m1(r, c), m2(r, c);

    // Accept matrices
    cout << "Enter elements for first matrix:\n";
    m1.acceptMatrix();

    cout << "Enter elements for second matrix:\n";
    m2.acceptMatrix();

    // Perform subtraction
    MyMatrix result = m1 - m2;

    // Display matrices and result
    cout << "\nFirst ";
    m1.displayMatrix();

    cout << "\nSecond ";
    m2.displayMatrix();

    cout << "\nResultant ";
    result.displayMatrix();

    getch(); // Wait for keypress (Turbo C++ only)
    return 0;
}
```



```
First Matrix:
1 2 3
4 5 6

Second Matrix:
6 5 4
3 2 1

Resultant Matrix:
-5 -3 -1
1 3 5
```

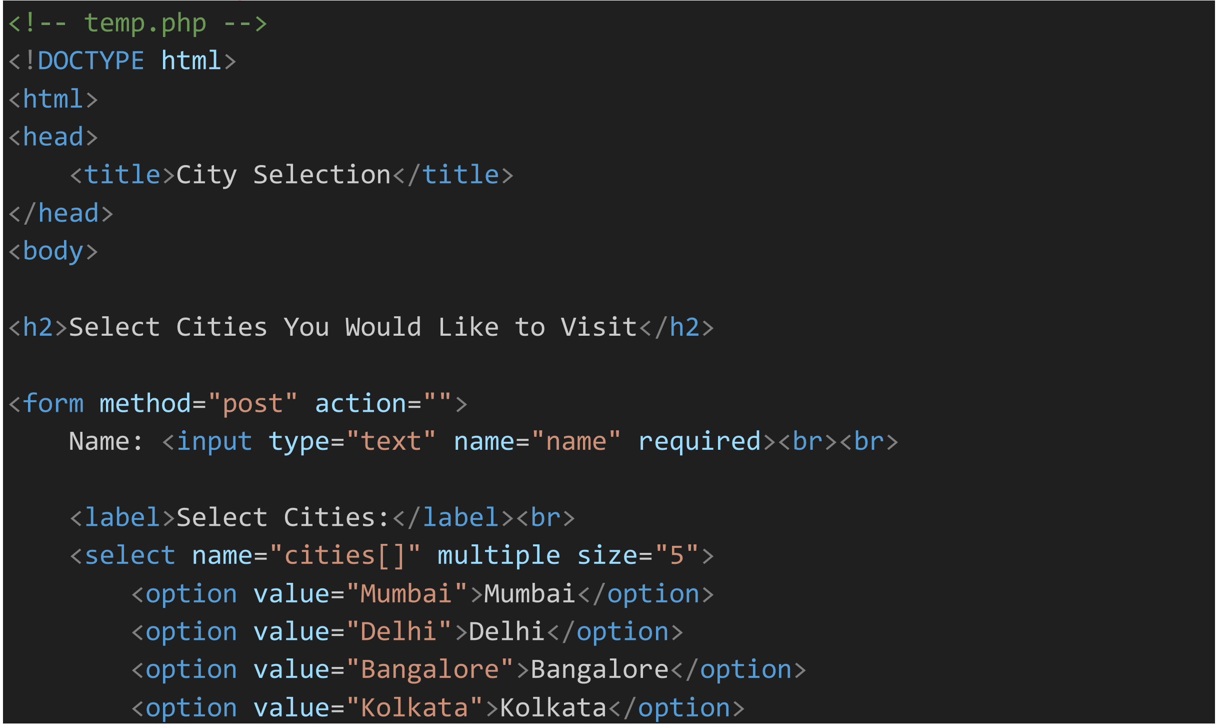
Output , Enter Matrix Elements

b) Design two base classes Student (S id, Name, C lass) and Competition (C id, C Name). Derive a class Stud Comp(Rank) from it. Write a menu driven program to perform following functions:
i. Accept information.
ii. Display information.
iii. Display Student Details in the ascending order of Rank of a specified competition.
(Use array of objects) [Marks 25]

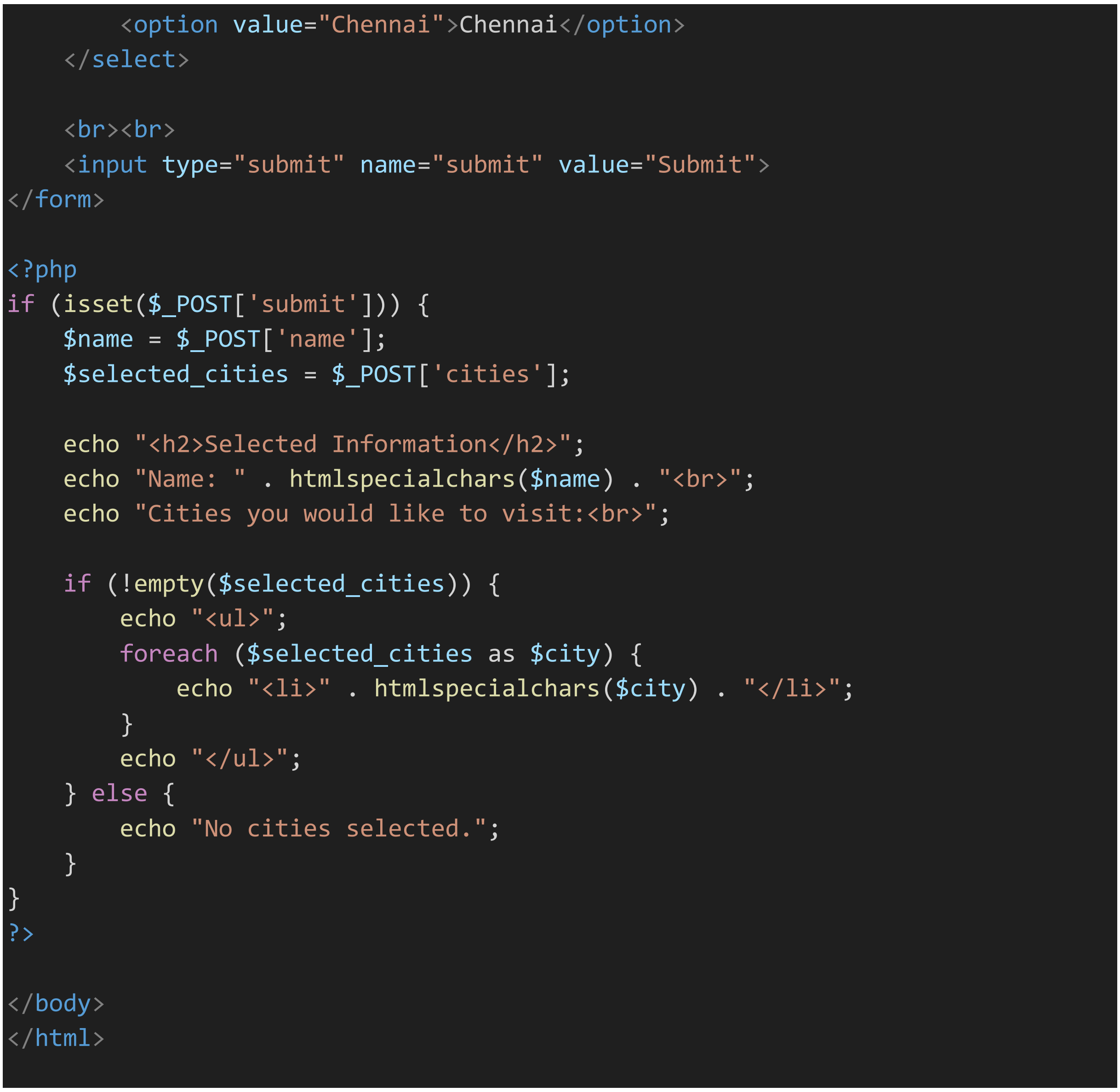
Already Done Above { Kisi Ek Slip me tha }

Q. 2)

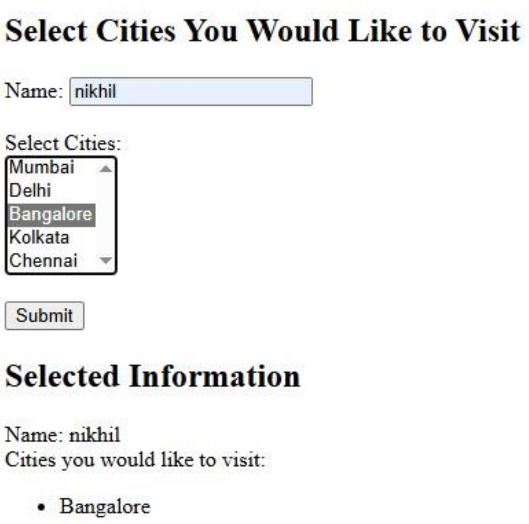
a) Write PHP program accept name, select your cities you would like to visit and display selected information on page. (Use multi-valued parameter),. [Marks l5]



```
<!-- temp.php -->
<!DOCTYPE html>
<html>
<head>
    <title>City Selection</title>
</head>
<body>

<h2>Select Cities You Would Like to Visit</h2>

<form method="post" action="">
    Name: <input type="text" name="name" required><br><br>

    <label>Select Cities:</label><br>
    <select name="cities[]" multiple size="5">
        <option value="Mumbai">Mumbai</option>
        <option value="Delhi">Delhi</option>
        <option value="Bangalore">Bangalore</option>
        <option value="Kolkata">Kolkata</option>
```



```
        <option value="Chennai">Chennai</option>
    </select>

    <br><br>
    <input type="submit" name="submit" value="Submit">
</form>

<?php
if (isset($_POST['submit'])) {
    $name = $_POST['name'];
    $selected_cities = $_POST['cities'];

    echo "<h2>Selected Information</h2>";
    echo "Name: " . htmlspecialchars($name) . "<br>";
    echo "Cities you would like to visit:<br>";

    if (!empty($selected_cities)) {
        echo "<ul>";
        foreach ($selected_cities as $city) {
            echo "<li>" . htmlspecialchars($city) . "</li>";
        }
        echo "</ul>";
    } else {
        echo "No cities selected.";
    }
}
?>

</body>
</html>
```



Select Cities You Would Like to Visit

Name: nikhil

Select Cities:

Selected Information

Name: nikhil
Cities you would like to visit:

- Bangalore

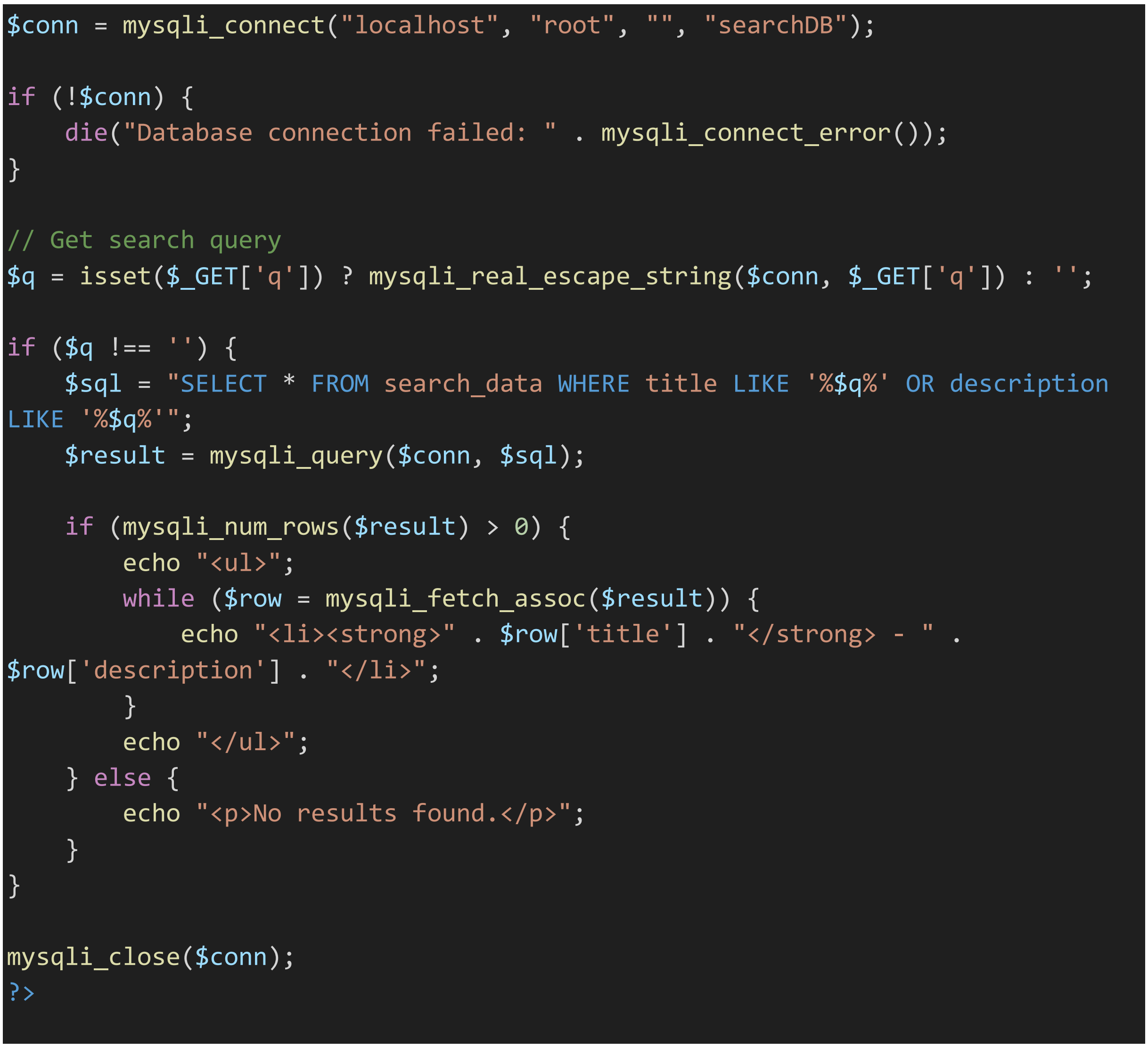
b) Write Ajax program to fetch suggestions when is user is typing in a textbox. (eg like google suggestions.Hint create array of suggestions and matching string will be displayed) [Marks 25]

HTML / AJAX:-

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Live Search Engine</title>
    <script>
        function searchQuery() {
            let query = document.getElementById("search").value;
            if (query.length === 0) {
                document.getElementById("results").innerHTML = "";
                return;
            }

            let xhr = new XMLHttpRequest();
            xhr.open("GET", "1.php?q=" + query, true);
            xhr.onreadystatechange = function () {
                if (xhr.readyState == 4 && xhr.status == 200) {
                    document.getElementById("results").innerHTML =
xhr.responseText;
                }
            };
            xhr.send();
        }
    </script>
</head>
<body>

    <h2>AJAX Search Engine</h2>
    <input type="text" id="search" onkeyup="searchQuery()" placeholder="Search
here..." autocomplete="off">
    <div id="results"></div>

</body>
</html>
```

PHP File:-

```
<?php
// Database connection
```



```php
$conn = mysqli_connect("localhost", "root", "", "searchDB");

if (!$conn) {
    die("Database connection failed: " . mysqli_connect_error());
}

// Get search query
$q = isset($_GET['q']) ? mysqli_real_escape_string($conn, $_GET['q']) : '';

if ($q !== '') {
    $sql = "SELECT * FROM search_data WHERE title LIKE '%$q%' OR description
LIKE '%$q%'";
    $result = mysqli_query($conn, $sql);

    if (mysqli_num_rows($result) > 0) {
        echo "<ul>";
        while ($row = mysqli_fetch_assoc($result)) {
            echo "<li><strong>" . $row['title'] . "</strong> - " .
$row['description'] . "</li>";
        }
        echo "</ul>";
    } else {
        echo "<p>No results found.</p>";
    }
}

mysqli_close($conn);
?>
```

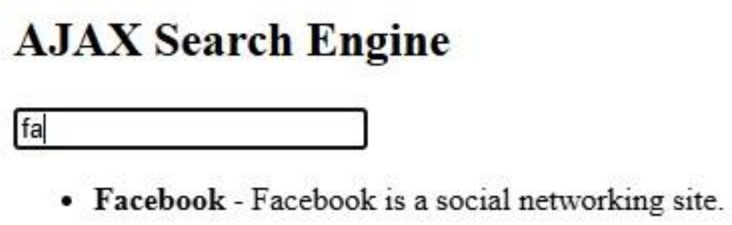
Sql Queries :-

CREATE DATABASE searchDB;

USE searchDB;

```sql
CREATE TABLE search_data (
    id INT AUTO_INCREMENT PRIMARY KEY,
    title VARCHAR(255) NOT NULL,
    description TEXT NOT NULL
);
```

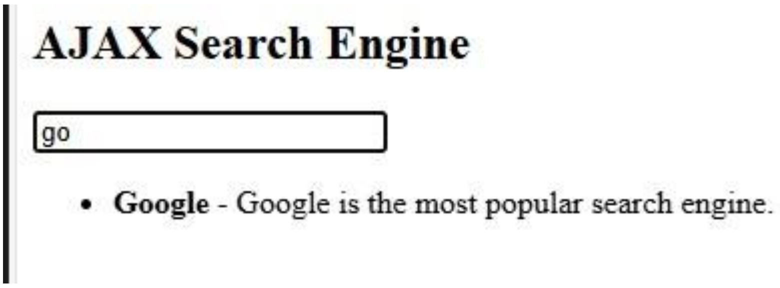
```
INSERT INTO search_data (title, description) VALUES
('Google', 'Google is the most popular search engine.'),
('YouTube', 'YouTube is a video-sharing platform.'),
('Facebook', 'Facebook is a social networking site.'),
('Amazon', 'Amazon is an e-commerce website.'),
('Netflix', 'Netflix is a streaming service.');
```



AJAX Search Engine

fa

- **Facebook** - Facebook is a social networking site.



AJAX Search Engine

go

- **Google** - Google is the most popular search engine.

Q. 3) Lab Book **[Marks 10]**

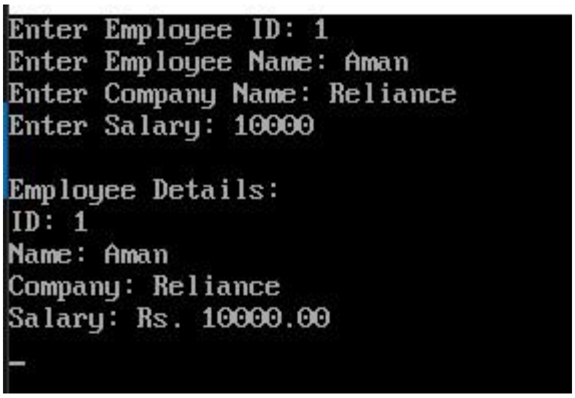
Q. 4) Viva **[Marks 10]**

Slip -15

Q. 1)

a) Create a C++ Class employee with data members with data members Emp_id, Emp_name, Company_name and salary. Write member functions to accept and display employee information. Design user defined manipulators to print salary.
[Marks 15]

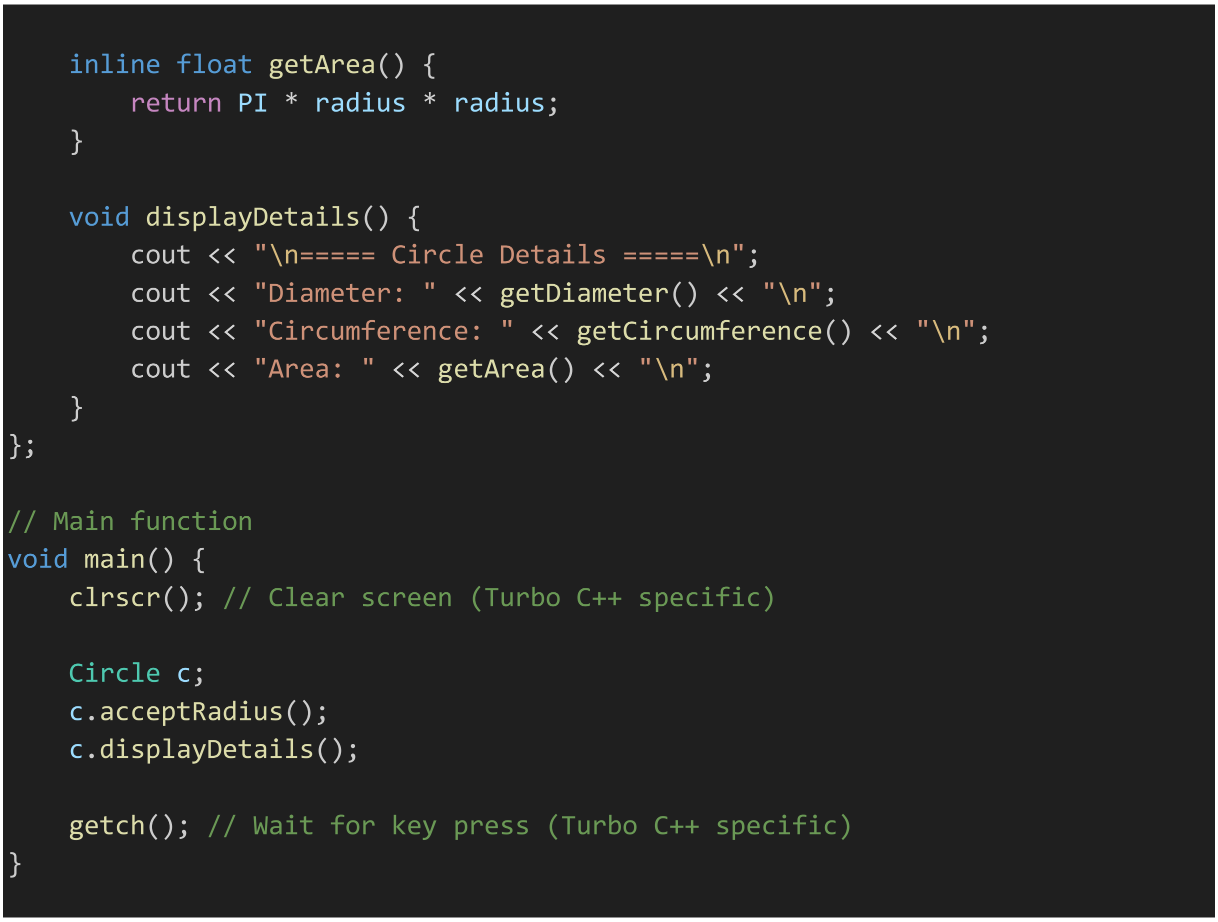
```
#include <iostream.h>
#include <conio.h>

class Employee {
private:
    int Emp_id;
    char Emp_name[50];
    char Company_name[50];
    float salary;

public:
    void acceptDetails() {
        cout << "Enter Employee ID: ";
        cin >> Emp_id;
        cout << "Enter Employee Name: ";
        cin >> Emp_name;
        cout << "Enter Company Name: ";
        cin >> Company_name;
        cout << "Enter Salary: ";
        cin >> salary;
    }

    void displayDetails() {
        cout << "\nEmployee Details:\n";
        cout << "ID: " << Emp_id << "\n";
        cout << "Name: " << Emp_name << "\n";
        cout << "Company: " << Company_name << "\n";
        cout << "Salary: " << formatSalary(salary) << "\n";
    }

    // User-defined manipulator for salary formatting
    static char* formatSalary(float sal) {
        static char formattedSalary[20];
        sprintf(formattedSalary, "Rs. %.2f", sal);
        return formattedSalary;
    }
};

// Main function
void main() {
    clrscr();  // Clear screen (Turbo C++ specific)

    Employee emp;
    emp.acceptDetails();
    emp.displayDetails();

    getch();  // Wait for key press (Turbo C++ specific)
}
```



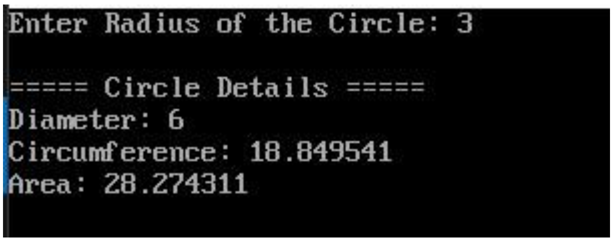
```
Enter Employee ID: 1
Enter Employee Name: Aman
Enter Company Name: Reliance
Enter Salary: 10000

Employee Details:
ID: 1
Name: Aman
Company: Reliance
Salary: Rs. 10000.00
_
```

b) Write a C++ program to accept radius of a circle. Calculate and display diameter, circumference as well as area of circle. (Use inline function) [Marks 25]

```
#include <iostream.h>
#include <conio.h>

const float PI = 3.14159;  // Define PI constant

class Circle {
private:
    float radius;

public:
    void acceptRadius() {
        cout << "Enter Radius of the Circle: ";
        cin >> radius;
    }

    inline float getDiameter() {
        return 2 * radius;
    }

    inline float getCircumference() {
        return 2 * PI * radius;
    }
```



```
    inline float getArea() {
        return PI * radius * radius;
    }

    void displayDetails() {
        cout << "\n===== Circle Details =====\n";
        cout << "Diameter: " << getDiameter() << "\n";
        cout << "Circumference: " << getCircumference() << "\n";
        cout << "Area: " << getArea() << "\n";
    }
};

// Main function
void main() {
    clrscr(); // Clear screen (Turbo C++ specific)

    Circle c;
    c.acceptRadius();
    c.displayDetails();

    getch(); // Wait for key press (Turbo C++ specific)
}
```



```
Enter Radius of the Circle: 3

===== Circle Details =====
Diameter: 6
Circumference: 18.849541
Area: 28.274311
```

Q. 2)

a) Write a PHP program to accept two strings from user and check whether entered strings are matching or not. (Use sticky form concept).
[Marks l5]

Already Done Above { Kisi Ek Slip me tha }

b) Write Ajax program to get player details from XML file when user select a player name. Create XML file for storing details of player (Country, player name, wickets, runs).
[Marks 25]

Already Done Above { Kisi Ek Slip me tha }

Q. 3) Lab Book **[Marks 10]**

Q. 4) Viva **[Marks 10]**

Slip -16

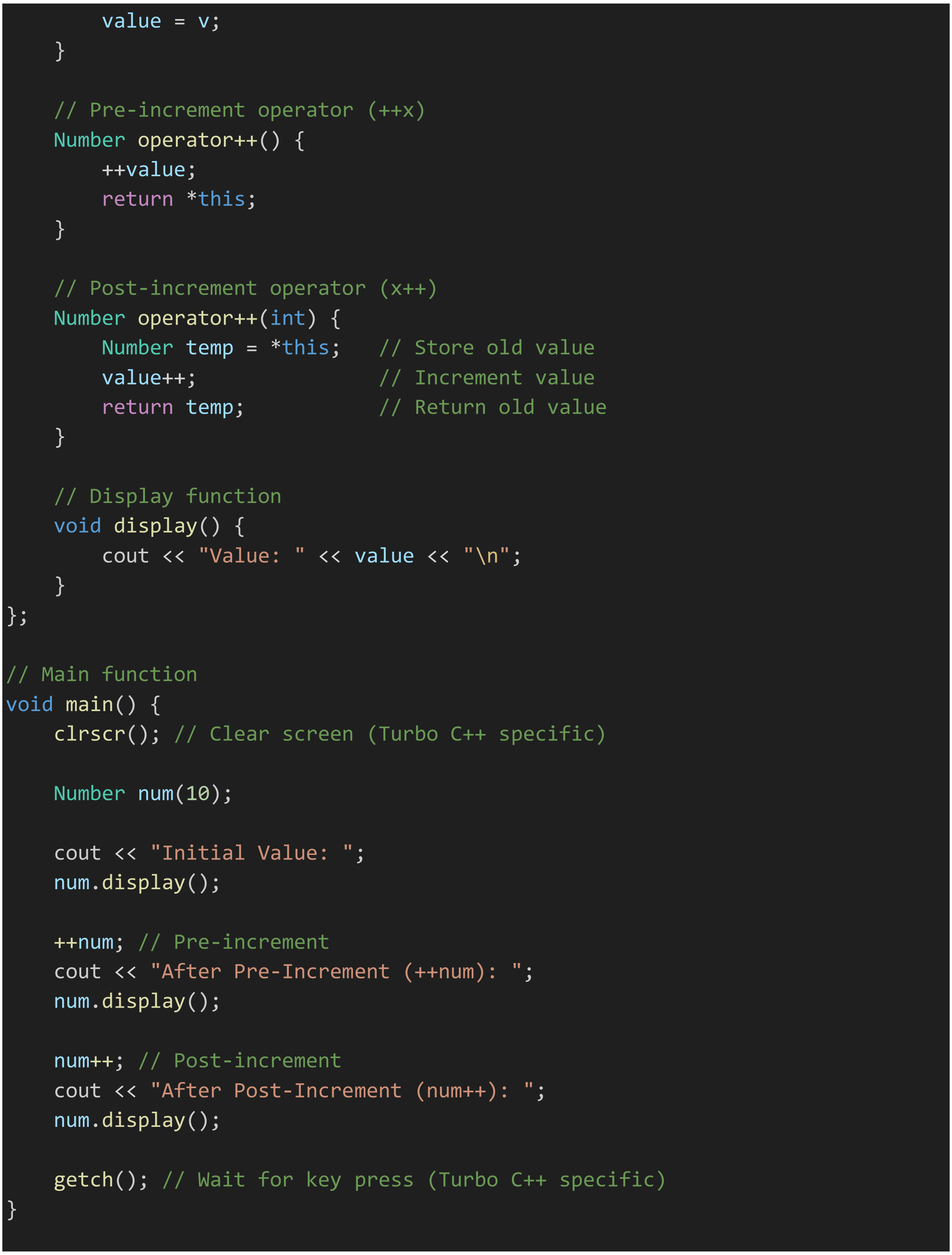
Q. 1)

a) Create a C++ Class student with data members rollno, name ,class , percentage. Accept two students information and display information of student having maximum percentage. (Use this pointer) **[Marks 15]**

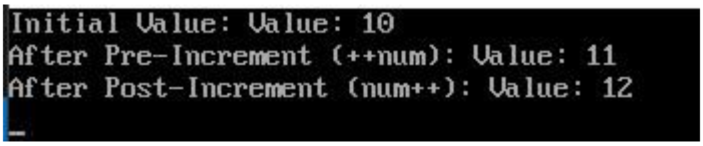
Already Done Above { Kisi Ek Slip me tha }

b) Write a C++ class number with integer data member. Write necessary member function to overload the operator unary pre and post increment '++'.
[Marks 25]

```
#include <iostream.h>
#include <conio.h>

class Number {
private:
    int value;

public:
    // Constructor to initialize value
    Number(int v = 0) {
```



```
        value = v;
    }

    // Pre-increment operator (++x)
    Number operator++() {
        ++value;
        return *this;
    }

    // Post-increment operator (x++)
    Number operator++(int) {
        Number temp = *this;    // Store old value
        value++;                // Increment value
        return temp;            // Return old value
    }

    // Display function
    void display() {
        cout << "Value: " << value << "\n";
    }
};

// Main function
void main() {
    clrscr(); // Clear screen (Turbo C++ specific)

    Number num(10);

    cout << "Initial Value: ";
    num.display();

    ++num; // Pre-increment
    cout << "After Pre-Increment (++num): ";
    num.display();

    num++; // Post-increment
    cout << "After Post-Increment (num++): ";
    num.display();

    getch(); // Wait for key press (Turbo C++ specific)
}
```



```
Initial Value: Value: 10
After Pre-Increment (++num): Value: 11
After Post-Increment (num++): Value: 12
_
```

Q. 2)

a) Write a PHP program to accept two strings from user and check whether entered strings are matching or not. (Use sticky form concept). **[Marks l5]**

Already Done Above { Kisi Ek Slip me tha }

b) Write Ajax program to get player details from XML file when user select a player name. Create XML file for storing details of player (Country, player name, wickets, runs). **[Marks 25]**

Already Done Above { Kisi Ek Slip me tha }

Q. 3) Lab Book **[Marks 10]**

Q. 4) Viva **[Marks 10]**