

Questions C++

Q1 Print Multiplication Table of a Number

Ans

```
#include <iostream>
```

```
using namespace std;
```

```
void printTable(int num) {
```

```
    for(int i = 1; i <= 10; ++i) {
```

```
        cout << num << " * " << i << " = " << num * i << endl;
```

```
    }
```

```
}
```

```
int main() {
```

```
    int number;
```

```
    cout << "Enter an integer: ";
```

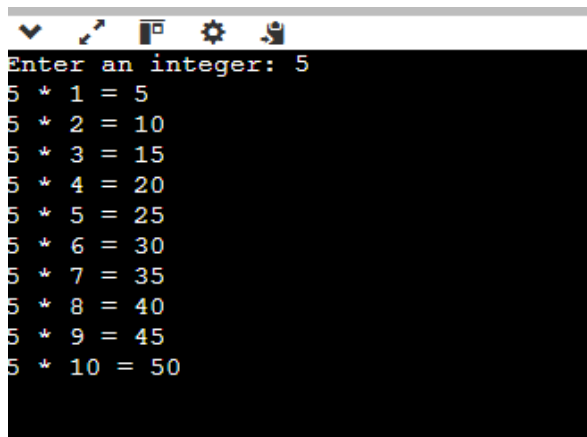
```
    cin >> number;
```

```
    printTable(number);
```

```
    return 0;
```

```
}
```

Output

A screenshot of a terminal window with a dark background. The prompt 'Enter an integer: 5' is at the top. Below it, ten lines of output show the multiplication of 5 by integers from 1 to 10, formatted as '5 * 1 = 5', '5 * 2 = 10', etc., up to '5 * 10 = 50'.

```
Enter an integer: 5
5 * 1 = 5
5 * 2 = 10
5 * 3 = 15
5 * 4 = 20
5 * 5 = 25
5 * 6 = 30
5 * 7 = 35
5 * 8 = 40
5 * 9 = 45
5 * 10 = 50
```

Q2. SUM OF ALL NATURAL NO

ANS #include <iostream>

using namespace std;

int sum(int n)

{

int s;

s=n*(n+1)/2;

cout<<s;

}

int main()

{

int n;

cout<<"enter the value"<<endl;

cin>>n;

cout<<"sum of n natural no"<<endl;

sum(n);

}

OUTPUT:

```
main.cpp: In function 'int sum(int)':
main.cpp:8:1: warning: no return statement
    8 | }
      | ^
enter the value
5
sum of n natural no
15

...Program finished with exit code 0
Press ENTER to exit console.
```

Q3 NUMBER IS PRIME OR NOT

ANS

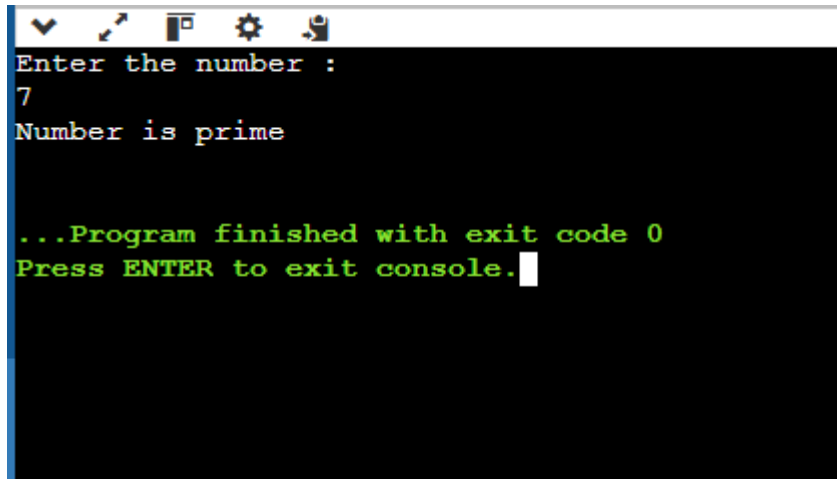
```
#include <iostream>
using namespace std;
int main()
{
    int n,s;
    cout<<"Enter the number :"<<endl;
    cin>>n;
    s=0;
    for(int i=2;i<n;i++)
    {
        if(n%i==0)
        {
            s++;
        }
    }
    if(s>0)
    {
        cout<<"Number is not prime"<<endl;
    }
    else
```

```

{
    cout<<"Number is prime"<<endl;
}
}

```

OUTPUT:



```

Enter the number :
7
Number is prime

...Program finished with exit code 0
Press ENTER to exit console.

```

Q4. COUNT THE TOTAL NO OF DIGIT IN GIVEN NO N

ANS

```
#include <iostream>
```

```
using namespace std;
```

```
int countDigits(int n) {
```

```
    int count = 0;
```

```
    while (n > 0) {
```

```
        n /= 10;
```

```
        count++;
```

```
    }
```

```
    return count;
```

```
}
```

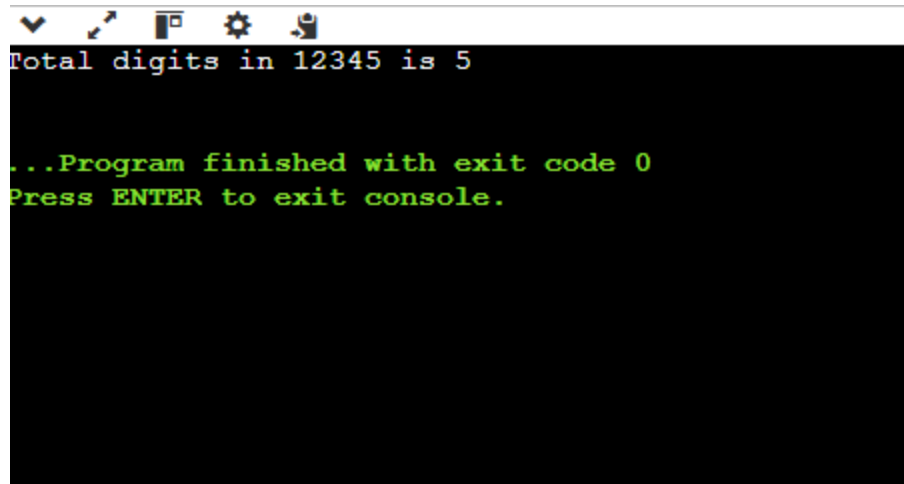
```
int main() {
```

```
    int n = 12345;
```

```
    cout << "Total digits in " << n << " is " << countDigits(n) << endl;
```

```
    return 0;
}
```

OUTPUT:

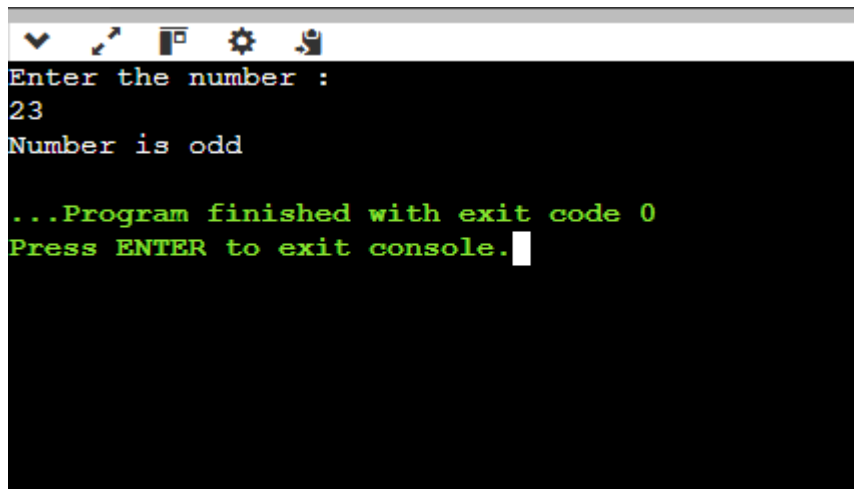
A screenshot of a terminal window with a black background and green text. The window has a standard macOS-style title bar at the top with icons for window control (red, yellow, green buttons), zoom, and settings. The text in the terminal reads: "Total digits in 12345 is 5" on the first line, followed by "...Program finished with exit code 0" on the second line, and "Press ENTER to exit console." on the third line. The rest of the terminal area is empty.

Q5 EVEN OR ODD

ANS

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    cout<<"Enter the number :"<<endl;
    cin>>n;
    if(n%2==0)
    {
        cout<<"Number is even";
    }
    else
    {
        cout<<"Number is odd";
    }
}
```

OUTPUT:



```
Enter the number :
23
Number is odd

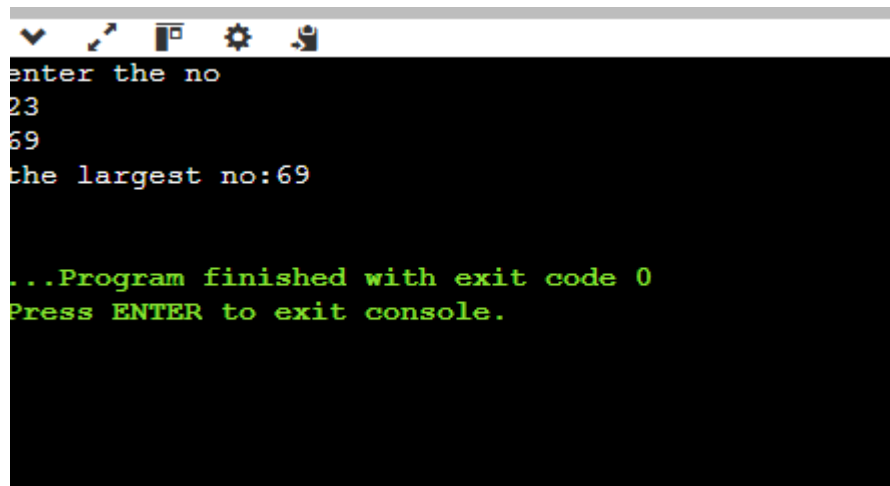
...Program finished with exit code 0
Press ENTER to exit console.
```

Q6 FIND THE LARGEST TWO NO

ANS

```
#include <iostream>
using namespace std;
int main()
{
    int a,b;
    cout<<"enter the no"<<endl;
    cin>>a>>b;
    if(a>b)
    {
        cout<<"the largest no:"<<a<<endl;
    }
    else
    {
        cout<<"the largest no:"<<b<<endl;
    }
}
```

OUTPUT



```
enter the no
23
69
the largest no:69

...Program finished with exit code 0
Press ENTER to exit console.
```

Q7 SUM OF ODD NO UPTO N

ANS

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int n, sum = 0;
```

```
    cout << "Enter the value of N: ";
```

```
    cin >> n;
```

```
    // Calculate the sum of odd numbers
```

```
    for (int i = 1; i <= n; i += 2) {
```

```
        sum += i;
```

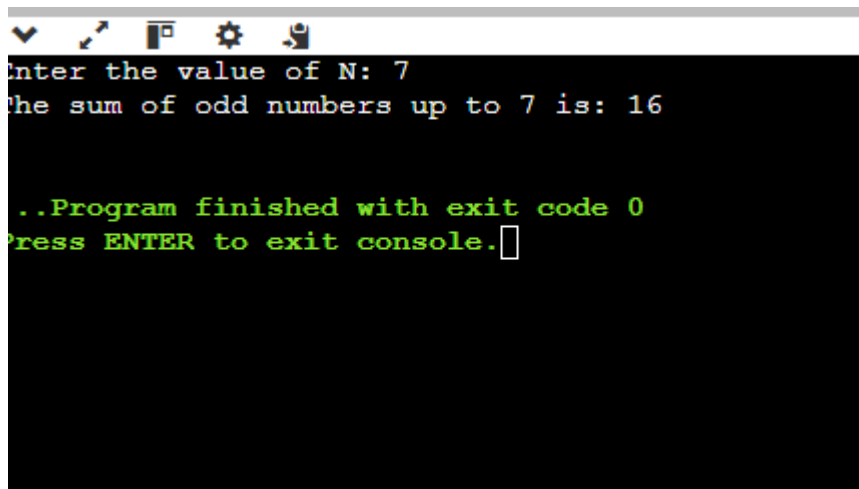
```
    }
```

```
    cout << "The sum of odd numbers up to " << n << " is: " << sum << endl;
```

```
    return 0;
```

```
}
```

Output:



```
Enter the value of N: 7
The sum of odd numbers up to 7 is: 16

..Program finished with exit code 0
Press ENTER to exit console.
```

Q8. Write a program to calculate the area of different shapes using function overloading. Implement overloaded functions to compute the area of a circle, a rectangle, and a triangle.

Ans

```
#include <iostream>

#include <cmath>

using namespace std;

double area(double radius) {
    return M_PI * radius * radius;
}

double area(double length, double width){
    return length * width;
}

double area1(double base, double height){
    return 0.5 * base * height;
}

int main() {
    double radius, length, width, base, height;

    cout << "Enter the radius of the circle: ";
    cin >> radius;
```



```

    cout << "Area of the circle: " << area(radius) << endl;

    cout<<"Enter the length:";

    cin>> length;

    cout<<"Enter the width:";

    cin>> width;

    cout<<"Area of rectangle:"<< area(length,width) << endl;

    cout<<"Enter the base:";

    cin>> base;

    cout<<"Enter the height:";

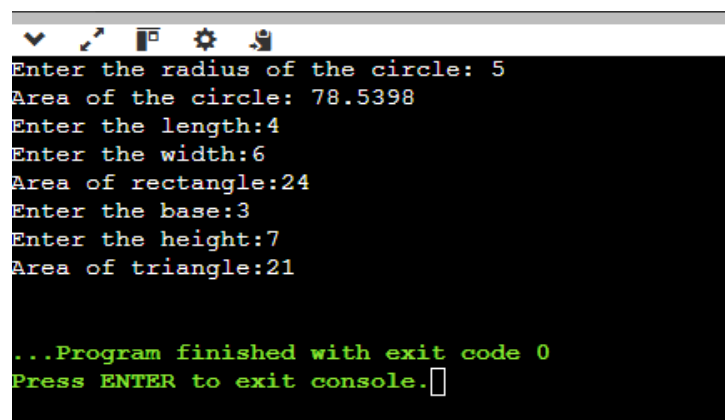
    cin>> height;

    cout<<"Area of triangle:"<< area(base,height) << endl;


    return 0;
}

```

Output



```

Enter the radius of the circle: 5
Area of the circle: 78.5398
Enter the length:4
Enter the width:6
Area of rectangle:24
Enter the base:3
Enter the height:7
Area of triangle:21

...Program finished with exit code 0
Press ENTER to exit console.

```

Q9. Write a program that demonstrates function overloading to calculate the salary of employees at different levels in a company hierarchy. Implement overloaded functions to compute salary for:

- Intern (basic stipend).
- Regular employee (base salary + bonuses).
- Manager (base salary + bonuses + performance incentives).

Ans

```
#include <iostream>
```

```
using namespace std;
```

```
int calculateSalary(int stipend) {  
    return stipend;  
}
```

```
int calculateSalary(int baseSalary, int bonuses) {  
    return baseSalary + bonuses;  
}
```

```
int calculateSalary(int baseSalary, int bonuses, int incentives) {  
    return baseSalary + bonuses + incentives;  
}
```

```
int main() {  
    int stipend, baseSalary, bonuses, incentives;  
  
    cout << "Enter stipend for intern: ";  
    cin >> stipend;  
    cout << "Intern Salary: " << calculateSalary(stipend) << endl;  
  
    cout << "Enter base salary and bonuses for a regular employee: ";  
    cin >> baseSalary >> bonuses;  
    cout << "Employee Salary: " << calculateSalary(baseSalary, bonuses) << endl;  
  
    cout << "Enter base salary, bonuses, and incentives for a manager: ";  
    cin >> baseSalary >> bonuses >> incentives;  
    cout << "Manager Salary: " << calculateSalary(baseSalary, bonuses, incentives) << endl;  
  
    return 0;  
}
```

Output

```
Enter stipend for intern: 10000
Intern Salary: 10000
Enter base salary and bonuses for a regular employee: 50000
20000
Employee Salary: 70000
Enter base salary, bonuses, and incentives for a manager: 80000
30000
20000
Manager Salary: 130000

...Program finished with exit code 0
Press ENTER to exit console.
```

Q10 Create a C++ program that uses polymorphism to calculate the area of various shapes. Define a base class Shape with a virtual method calculateArea(). Extend this base class into the following derived classes:

Rectangle: Calculates the area based on length and width.

Circle: Calculates the area based on the radius.

Triangle: Calculates the area using base and height.

The program should use dynamic polymorphism to handle these shapes and display the area of each.

Ans

```
#include <iostream>
```

```
#include <cmath>
```

```
using namespace std;
```

```
class Shape {
```

```
public:
```

```
    virtual void calculateArea() = 0;
```

```
};
```

```
class Rectangle : public Shape {
```

private:

float length, width;

public:

Rectangle(float l, float w) : length(l), width(w) {}

void calculateArea() override {

cout << "Shape: Rectangle" << endl;

cout << "Area: " << length * width << endl;

}

};

class Circle : public Shape {

private:

float radius;

public:

Circle(float r) : radius(r) {}

void calculateArea() override {

cout << "Shape: Circle" << endl;

cout << "Area: " << M_PI * radius * radius << endl;

}

};

class Triangle : public Shape {

private:

float base, height;

public:

Triangle(float b, float h) : base(b), height(h) {}

void calculateArea() override {

cout << "Shape: Triangle" << endl;

cout << "Area: " << 0.5 * base * height << endl;

}

};

```
int main() {  
    int shapeType;  
    cout << "Enter shape type (1 for Rectangle, 2 for Circle, 3 for Triangle): ";  
    cin >> shapeType;  
  
    Shape* shape = nullptr;  
  
    switch(shapeType) {  
        case 1: {  
            float length, width;  
            cout << "Enter length and width of the rectangle: ";  
            cin >> length >> width;  
            shape = new Rectangle(length, width);  
            break;  
        }  
        case 2: {  
            float radius;  
            cout << "Enter radius of the circle: ";  
            cin >> radius;  
            shape = new Circle(radius);  
            break;  
        }  
        case 3: {  
            float base, height;  
            cout << "Enter base and height of the triangle: ";  
            cin >> base >> height;  
            shape = new Triangle(base, height);  
            break;  
        }  
        default:
```

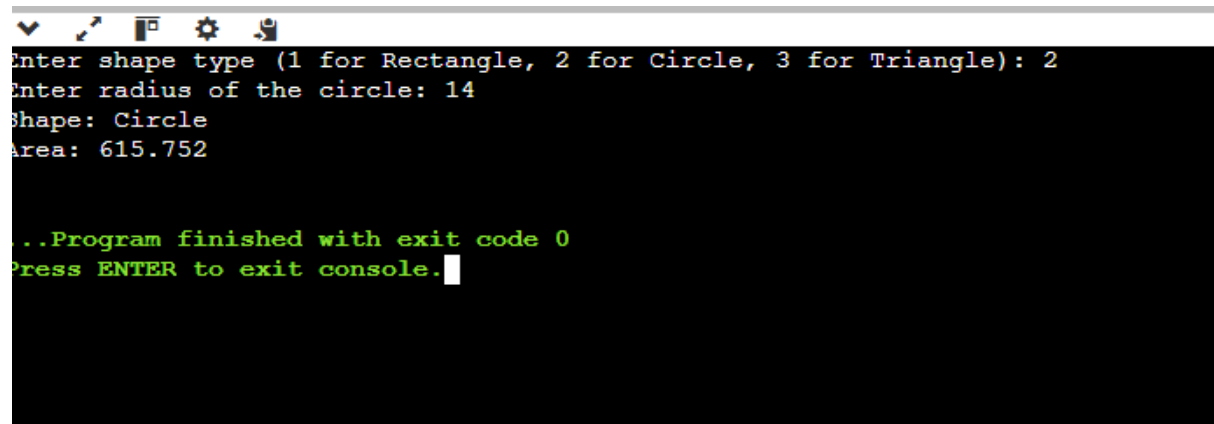
```
        cout << "Invalid shape type." << endl;

        return 1;
    }

    if(shape) {
        shape->calculateArea();
        delete shape;
    }

    return 0;
}
```

Output

A screenshot of a terminal window with a black background and white text. The window has a title bar with standard icons. The text inside shows the program's execution flow: it prompts for a shape type (1 for Rectangle, 2 for Circle, 3 for Triangle), receives input '2', prompts for the radius of the circle, receives input '14', outputs 'Shape: Circle', and outputs 'Area: 615.752'. At the end, it shows '...Program finished with exit code 0' and 'Press ENTER to exit console.' with a cursor.

```
Enter shape type (1 for Rectangle, 2 for Circle, 3 for Triangle): 2
Enter radius of the circle: 14
Shape: Circle
Area: 615.752

...Program finished with exit code 0
Press ENTER to exit console.
```