Assignment 02

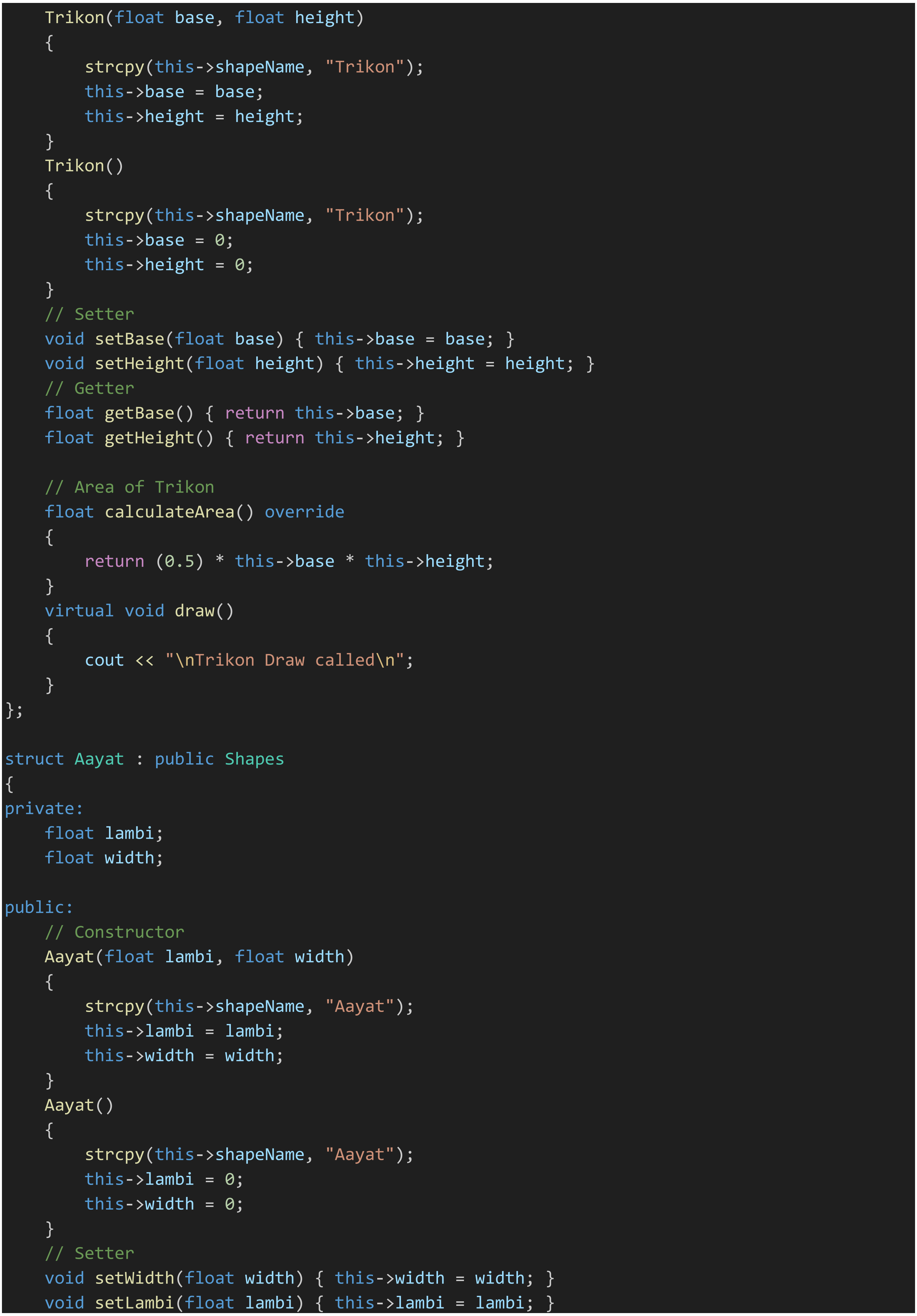
```
//Q1) Write a functions to add 2 int value, 2 float value, 1 int and
1 float value and vice versa. similary write functions for all other
arithmetic operations.

#include <iostream>
using namespace std;
struct Calculator
{
    // Add
    int add(int a, int b) { return a + b; }
    float add(float a, float b) { return a + b; }
    float add(int a, float b) { return a + b; }
    float add(float a, int b) { return a + b; }
    // Suctract
    int sub(int a, int b) { return a - b; }
    float sub(float a, float b) { return a - b; }
    float sub(int a, float b) { return a - b; }
    float sub(float a, int b) { return a - b; }
    // Divide
    int div(int a, int b) { return a / b; }
    float div(float a, float b) { return a / b; }
    float div(int a, float b) { return a / b; }
    float div(float a, int b) { return a / b; }
    // Mul
    int mul(int a, int b) { return a * b; }
    float mul(float a, float b) { return a * b; }
    float mul(int a, float b) { return a * b; }
    float mul(float a, int b) { return a * b; }
};

int main()
{
    int a, b;
    float c, d;
    Calculator ad;
    cout << "Enter Two Integers : ";
    cout << "\nNumber1 :";
    cin >> a;
    cout << "\nNumber2 :";
    cin >> b;
    cout << "\nAddition : " << ad.add(a, b);
    cout << "\nSub : " << ad.sub(a, b);
    cout << "\nMul : " << ad.mul(a, b);
    cout << "\nDiv : " << ad.div(a, b);

    cout << "\n\nEnter Two Floats : ";
    cout << "\nNumber1 :";
    cin >> c;
    cout << "\nNumber2 :";
    cin >> d;
    cout << "\nAddition : " << ad.add(c, d);
    cout << "\nSub : " << ad.sub(c, d);
    cout << "\nMul : " << ad.mul(c, d);
    cout << "\nDiv : " << ad.div(c, d);

    cout << "\n\nEnter  one integer One Float : ";
    cout << "\nNumber1 :";
    cin >> b;
    cout << "\nNumber2 :";
    cin >> d;
    cout << "\nAddition : " << ad.add(b, d);
    cout << "\nSub : " << ad.sub(b, d);
    cout << "\nMul : " << ad.mul(b, d);
    cout << "\nDiv : " << ad.div(b, d);

    cout << "\n\nEnter One Float one integer : ";
    cout << "\nNumber1 :";
    cin >> c;
    cout << "\nNumber2 :";
    cin >> a;
    cout << "\nAddition : " << ad.add(c, a);
    cout << "\nSub : " << ad.sub(c, a);
    cout << "\nMul : " << ad.mul(c, a);
    cout << "\nDiv : " << ad.div(c, a);
}
```

Output:

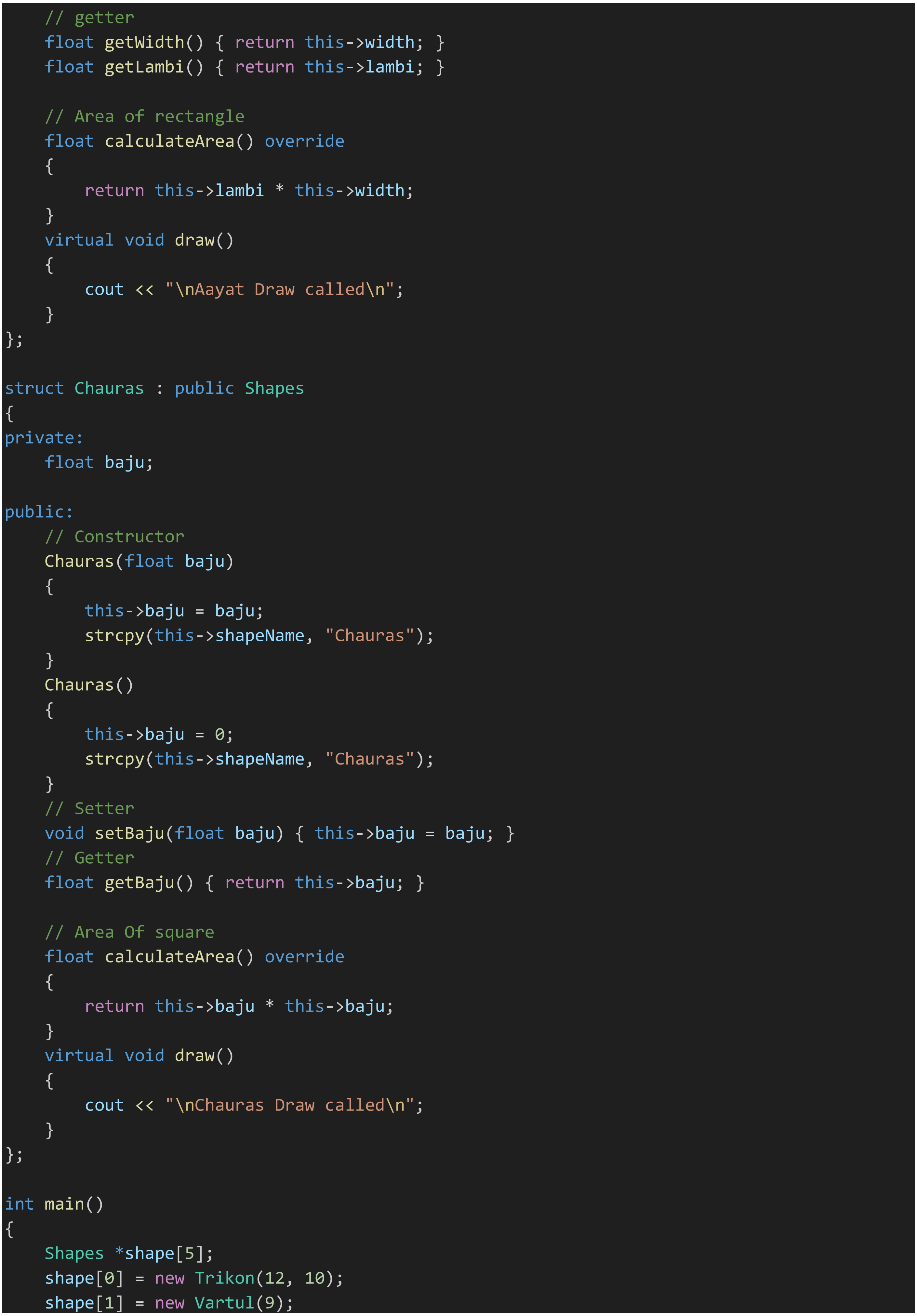
```
PS D:\Fullstack-Java-FirstBit-Solutions>  & 'c:\Users\bhagv\.vscode\.....\TDM-GCC-64\bin\gdb.exe' '--interpreter=mi'

Enter Two Integers :

Number1 :123


Number2 :10


Addition : 133

Sub : 113

Mul : 1230

Div : 12


Enter Two Floats :

Number1 :120.80


Number2 :12.8


Addition : 133.6
```

```
Sub : 108

Mul : 1546.24

Div : 9.4375


Enter  one integer One Float :

Number1 :120


Number2 :11.999


Addition : 131.999

Sub : 108.001

Mul : 1439.88

Div : 10.0008


Enter One Float one integer :

Number1 :119.9999999


Number2 :10


Addition : 130

Sub : 110

Mul : 1200

Div : 12

PS D:\Fullstack-Java-FirstBit-Solutions>
```

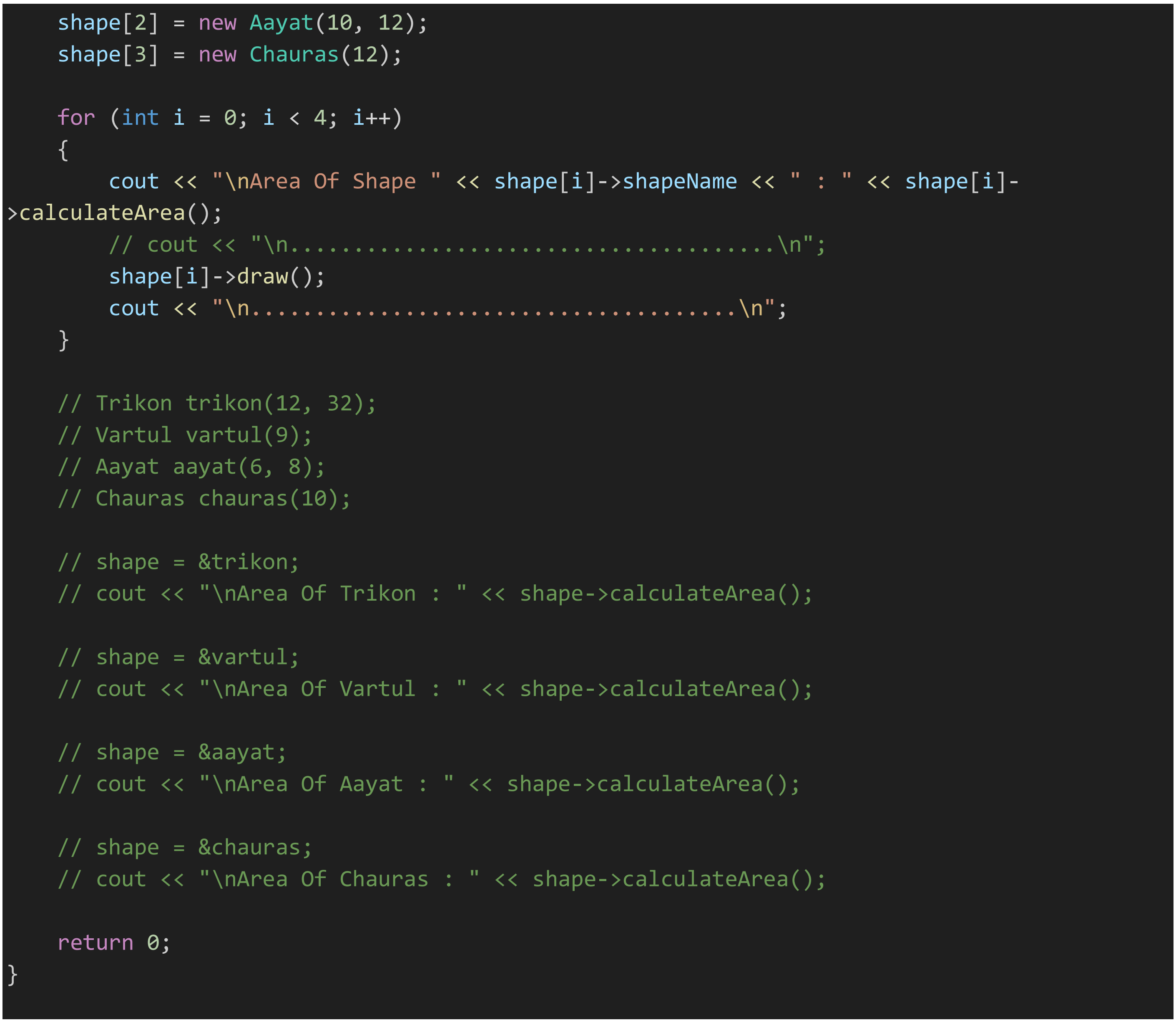
```
// 2. Write a function to calculate area of shape, for calculating area of triangle, rectangle, circle differently.

#include <iostream>
#include <string.h>
using namespace std;

struct Shapes
{
    char shapeName[20];
    virtual float calculateArea()
    {
        cout << "\nShapes CalculateArea called\n";
        return 0;
    }
    // void draw()
    virtual void draw()
    {
        cout << "\nShape Draw called\n";
    }
};

struct Vartul : public Shapes
{
private:
    float radious;

public:
    // Constructor
    Vartul(float red)
    {
        this->radious = red;
        strcpy(this->shapeName, "Vartul");
    }
    Vartul()
    {
        strcpy(this->shapeName, "Vartul");
        this->radious = 0;
    }

    // Setter
    void setRadious(float radious) { this->radious = radious; }
    // Getter
    float getRadious() { return this->radious; }
    // Area of Circle
    float calculateArea() override
    {
        return 3.14 * (this->radious * this->radious);
    }
    virtual void draw()
    {
        cout << "\nVartul Draw called\n";
    }
};

struct Trikon : public Shapes
{
private:
    float base;
    float height;

public:
    // Constructor
```



```cpp
    Trikon(float base, float height)
    {
        strcpy(this->shapeName, "Trikon");
        this->base = base;
        this->height = height;
    }
    Trikon()
    {
        strcpy(this->shapeName, "Trikon");
        this->base = 0;
        this->height = 0;
    }
    // Setter
    void setBase(float base) { this->base = base; }
    void setHeight(float height) { this->height = height; }
    // Getter
    float getBase() { return this->base; }
    float getHeight() { return this->height; }

    // Area of Trikon
    float calculateArea() override
    {
        return (0.5) * this->base * this->height;
    }
    virtual void draw()
    {
        cout << "\nTrikon Draw called\n";
    }
};

struct Aayat : public Shapes
{
private:
    float lambi;
    float width;

public:
    // Constructor
    Aayat(float lambi, float width)
    {
        strcpy(this->shapeName, "Aayat");
        this->lambi = lambi;
        this->width = width;
    }
    Aayat()
    {
        strcpy(this->shapeName, "Aayat");
        this->lambi = 0;
        this->width = 0;
    }
    // Setter
    void setWidth(float width) { this->width = width; }
    void setLambi(float lambi) { this->lambi = lambi; }
```



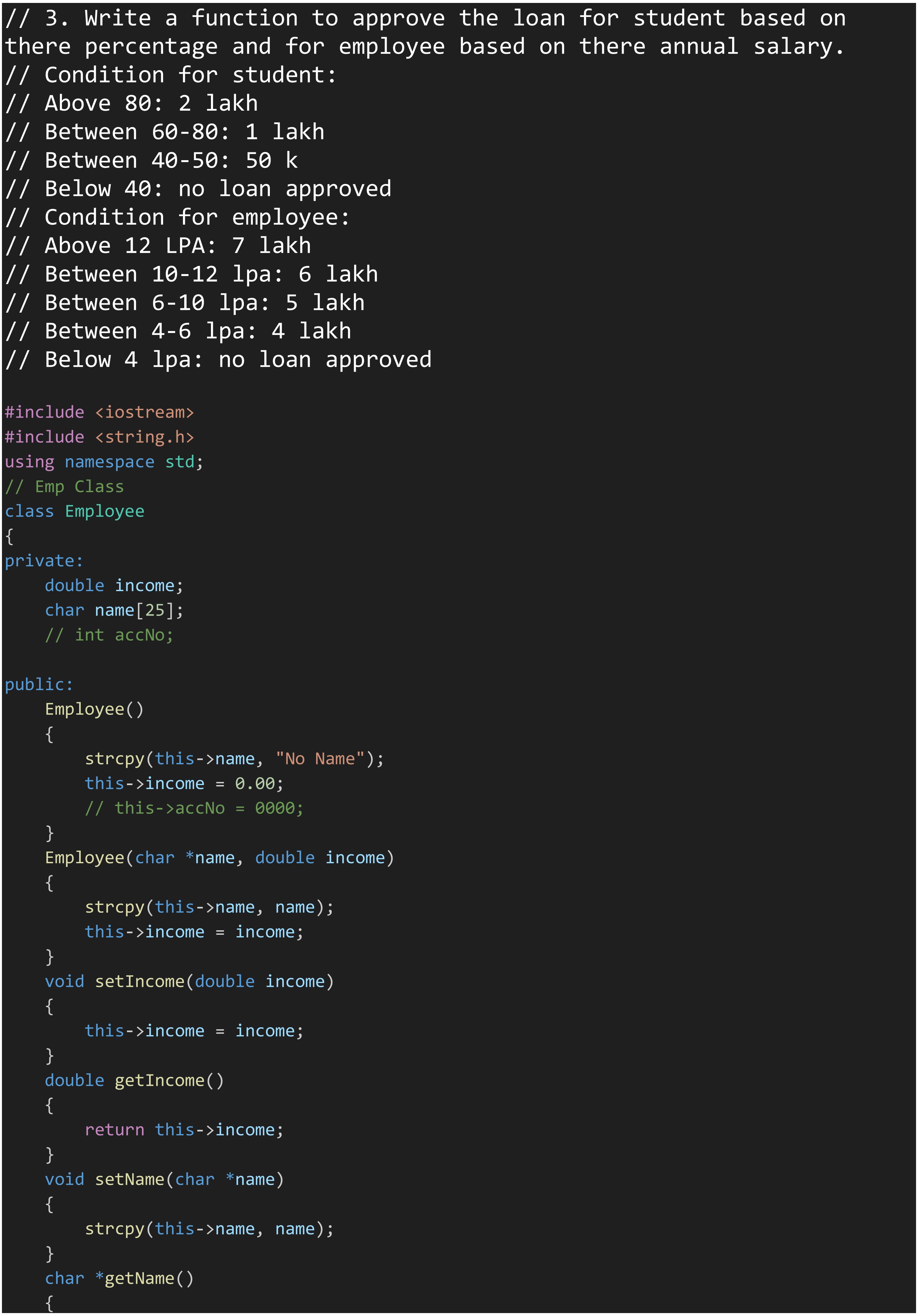
```cpp
    // getter
    float getWidth() { return this->width; }
    float getLambi() { return this->lambi; }

    // Area of rectangle
    float calculateArea() override
    {
        return this->lambi * this->width;
    }
    virtual void draw()
    {
        cout << "\nAayat Draw called\n";
    }
};

struct Chauras : public Shapes
{
private:
    float baju;

public:
    // Constructor
    Chauras(float baju)
    {
        this->baju = baju;
        strcpy(this->shapeName, "Chauras");
    }
    Chauras()
    {
        this->baju = 0;
        strcpy(this->shapeName, "Chauras");
    }
    // Setter
    void setBaju(float baju) { this->baju = baju; }
    // Getter
    float getBaju() { return this->baju; }

    // Area Of square
    float calculateArea() override
    {
        return this->baju * this->baju;
    }
    virtual void draw()
    {
        cout << "\nChauras Draw called\n";
    }
};

int main()
{
    Shapes *shape[5];
    shape[0] = new Trikon(12, 10);
    shape[1] = new Vartul(9);
```



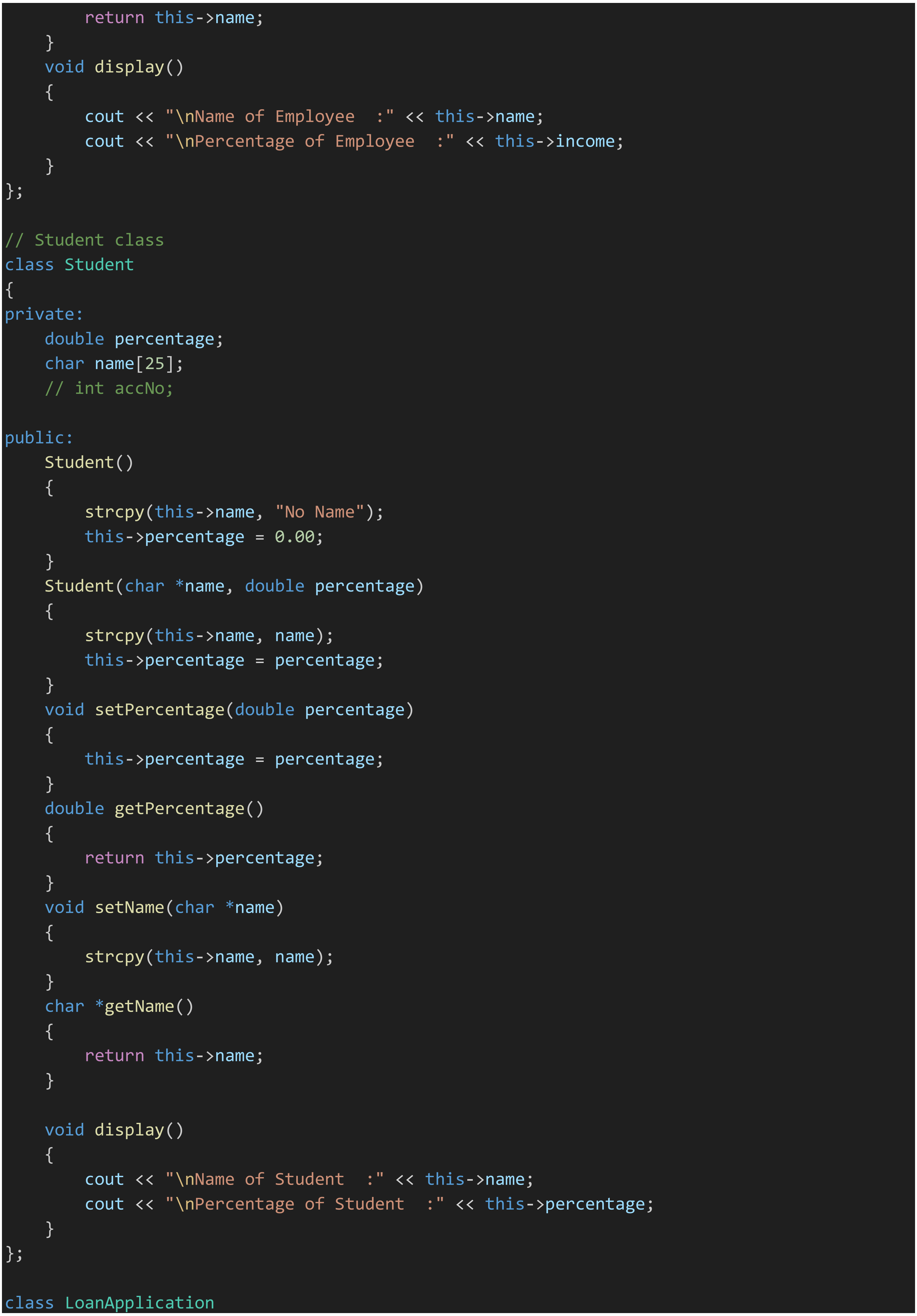
```
    shape[2] = new Aayat(10, 12);
    shape[3] = new Chauras(12);

    for (int i = 0; i < 4; i++)
    {
        cout << "\nArea Of Shape " << shape[i]->shapeName << " : " << shape[i]-
>calculateArea();
        // cout << "\n......................................\n";
        shape[i]->draw();
        cout << "\n......................................\n";
    }

    // Trikon trikon(12, 32);
    // Vartul vartul(9);
    // Aayat aayat(6, 8);
    // Chauras chauras(10);

    // shape = &trikon;
    // cout << "\nArea Of Trikon : " << shape->calculateArea();

    // shape = &vartul;
    // cout << "\nArea Of Vartul : " << shape->calculateArea();

    // shape = &aayat;
    // cout << "\nArea Of Aayat : " << shape->calculateArea();

    // shape = &chauras;
    // cout << "\nArea Of Chauras : " << shape->calculateArea();

    return 0;
}
```

Output:

```
PS D:\Fullstack-Java-FirstBit-Solutions\Basic-C-and-CPP\CPP\Assignments\Assignment02\output> &
.\'q2ShapesArea.exe'
Area Of Shape Trikon : 60
Trikon Draw called
.....................................
Area Of Shape Vartul : 254.34
Vartul Draw called
.....................................
Area Of Shape Aayat : 120
Aayat Draw called
.....................................

Area Of Shape Chauras : 144
Chauras Draw called
.....................................
PS D:\Fullstack-Java-FirstBit-Solutions\Basic-C-and-CPP\CPP\Assignments\Assignment02\output>
```

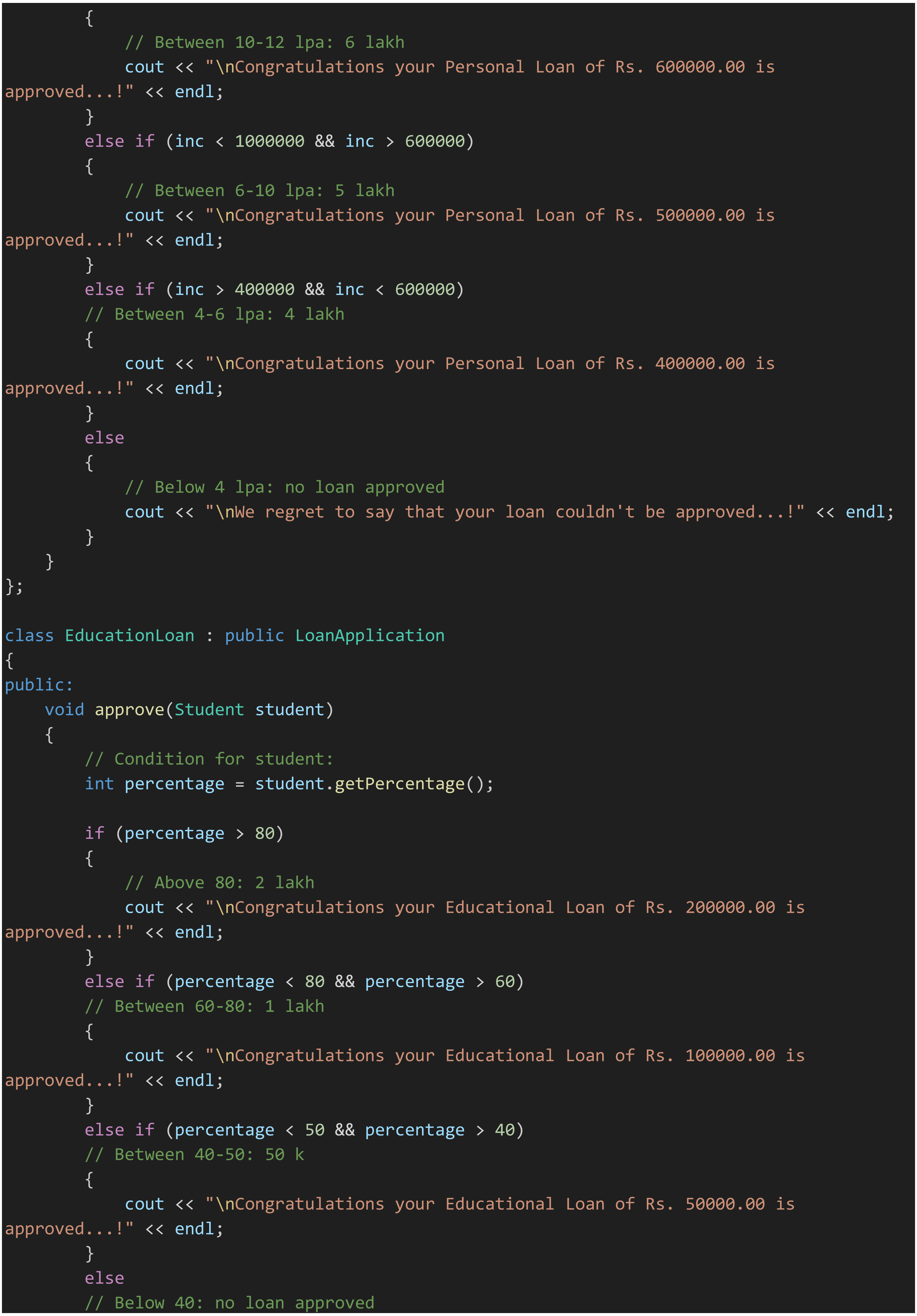


```
// 3. Write a function to approve the loan for student based on
there percentage and for employee based on there annual salary.
// Condition for student:
// Above 80: 2 lakh
// Between 60-80: 1 lakh
// Between 40-50: 50 k
// Below 40: no loan approved
// Condition for employee:
// Above 12 LPA: 7 lakh
// Between 10-12 lpa: 6 lakh
// Between 6-10 lpa: 5 lakh
// Between 4-6 lpa: 4 lakh
// Below 4 lpa: no loan approved

#include <iostream>
#include <string.h>
using namespace std;
// Emp Class
class Employee
{
private:
    double income;
    char name[25];
    // int accNo;

public:
    Employee()
    {
        strcpy(this->name, "No Name");
        this->income = 0.00;
        // this->accNo = 0000;
    }
    Employee(char *name, double income)
    {
        strcpy(this->name, name);
        this->income = income;
    }
    void setIncome(double income)
    {
        this->income = income;
    }
    double getIncome()
    {
        return this->income;
    }
    void setName(char *name)
    {
        strcpy(this->name, name);
    }
    char *getName()
    {
```



```
        return this->name;
    }
    void display()
    {
        cout << "\nName of Employee  :" << this->name;
        cout << "\nPercentage of Employee  :" << this->income;
    }
};

// Student class
class Student
{
private:
    double percentage;
    char name[25];
    // int accNo;

public:
    Student()
    {
        strcpy(this->name, "No Name");
        this->percentage = 0.00;
    }
    Student(char *name, double percentage)
    {
        strcpy(this->name, name);
        this->percentage = percentage;
    }
    void setPercentage(double percentage)
    {
        this->percentage = percentage;
    }
    double getPercentage()
    {
        return this->percentage;
    }
    void setName(char *name)
    {
        strcpy(this->name, name);
    }
    char *getName()
    {
        return this->name;
    }

    void display()
    {
        cout << "\nName of Student  :" << this->name;
        cout << "\nPercentage of Student  :" << this->percentage;
    }
};

class LoanApplication
```

```
{
public:
    virtual void approve()
    // void approve()
    {
        cout << "\nLoanApplication Approval";
    };
};

class PersonalLoan : public LoanApplication
{
public:
    void approve(Student student)
    {
        // Condition for student:
        int percentage = student.getPercentage();

        if (percentage > 80)
        {
            // Above 80: 2 lakh
            cout << "\nCongratulations your Personal Loan of Rs. 200000.00 is
approved...!" << endl;
        }
        else if (percentage < 80 && percentage > 60)
        // Between 60-80: 1 lakh
        {
            cout << "\nCongratulations your Personal Loan of Rs. 100000.00 is
approved...!" << endl;
        }
        else if (percentage < 50 && percentage > 40)
        // Between 40-50: 50 k
        {
            cout << "\nCongratulations your Personal Loan of Rs. 50000.00 is approved...!"
<< endl;
        }
        else
        // Below 40: no loan approved
        {
            cout << "\nWe regret to say that your loan couldn't be approved...!" << endl;
        }
    }
    void approve(Employee employee)
    {
        // Condition for employee:
        int inc = employee.getIncome();

        if (inc > 1200000)
        {
            // Above 12 LPA: 7 lakh
            cout << "\nCongratulations your Personal Loan of Rs. 700000.00 is
approved...!" << endl;
        }
        else if (inc < 1200000 && inc > 1000000)
```



```
        {
            // Between 10-12 lpa: 6 lakh
            cout << "\nCongratulations your Personal Loan of Rs. 600000.00 is 
approved...!" << endl;
        }
        else if (inc < 1000000 && inc > 600000)
        {
            // Between 6-10 lpa: 5 lakh
            cout << "\nCongratulations your Personal Loan of Rs. 500000.00 is 
approved...!" << endl;
        }
        else if (inc > 400000 && inc < 600000)
        // Between 4-6 lpa: 4 lakh
        {
            cout << "\nCongratulations your Personal Loan of Rs. 400000.00 is 
approved...!" << endl;
        }
        else
        {
            // Below 4 lpa: no loan approved
            cout << "\nWe regret to say that your loan couldn't be approved...!" << endl;
        }
    }
};

class EducationLoan : public LoanApplication
{
public:
    void approve(Student student)
    {
        // Condition for student:
        int percentage = student.getPercentage();

        if (percentage > 80)
        {
            // Above 80: 2 lakh
            cout << "\nCongratulations your Educational Loan of Rs. 200000.00 is 
approved...!" << endl;
        }
        else if (percentage < 80 && percentage > 60)
        // Between 60-80: 1 lakh
        {
            cout << "\nCongratulations your Educational Loan of Rs. 100000.00 is 
approved...!" << endl;
        }
        else if (percentage < 50 && percentage > 40)
        // Between 40-50: 50 k
        {
            cout << "\nCongratulations your Educational Loan of Rs. 50000.00 is 
approved...!" << endl;
        }
        else
        // Below 40: no loan approved
        {
            cout << "\nWe regret to say that your Educational loan couldn't be 
approved...!" << endl;
        }
    }
};

class HomeLoan : public LoanApplication
{
public:
    // void approve(Employee employee) { cout << "\nPersonalLoan Application Approval."; }
    void approve(Employee employee)
    {
        // Condition for employee:
        int inc = employee.getIncome();

        if (inc > 1200000)
        {
            // Above 12 LPA: 7 lakh
            cout << "\nCongratulations your Home Loan of Rs. 700000.00 is approved...!" << 
endl;
        }
        else if (inc < 1200000 && inc > 1000000)
        {
            // Between 10-12 lpa: 6 lakh
            cout << "\nCongratulations your Home Loan of Rs. 600000.00 is approved...!" << 
endl;
        }
        else if (inc < 1000000 && inc > 600000)
        {
            // Between 6-10 lpa: 5 lakh
            cout << "\nCongratulations your Home Loan of Rs. 500000.00 is approved...!" << 
endl;
        }
        else if (inc > 400000 && inc < 600000)
        // Between 4-6 lpa: 4 lakh
        {
            cout << "\nCongratulations your Home Loan of Rs. 400000.00 is approved...!" << 
endl;
        }
        else
        {
            // Below 4 lpa: no loan approved
            cout << "\nWe regret to say that your Home loan couldn't be approved...!" << 
endl;
        }
    }
};

class AutoLoan : public LoanApplication
{
public:
    void approve() { cout << "\nAutoLoan Application Approval."; }
    void approve(Employee employee)
    {
        // Condition for employee:
        int inc = employee.getIncome();

        if (inc > 1200000)
        {
            // Above 12 LPA: 7 lakh
            cout << "\nCongratulations your Auto Loan of Rs. 700000.00 is approved...!" <<
endl;
        }
        else if (inc < 1200000 && inc > 1000000)
        {
            // Between 10-12 lpa: 6 lakh
            cout << "\nCongratulations your Auto Loan of Rs. 600000.00 is approved...!" <<
endl;
        }
        else if (inc < 1000000 && inc > 600000)
        {
            // Between 6-10 lpa: 5 lakh
            cout << "\nCongratulations your Auto Loan of Rs. 500000.00 is approved...!" <<
endl;
        }
        else if (inc > 400000 && inc < 600000)
        // Between 4-6 lpa: 4 lakh
        {
            cout << "\nCongratulations your Auto Loan of Rs. 400000.00 is approved...!" <<
endl;
        }
        else
        {
            // Below 4 lpa: no loan approved
            cout << "\nWe regret to say that your Auto loan couldn't be approved...!" <<
endl;
        }
    }
};

int main()
{
    // LoanApplication *LoanApplications[4];
    EducationLoan ed;
    Student s1("Happy", 84.99);
    s1.display();
    ed.approve(s1);
    // LoanApplications[0]->approve(s1);
    Student s2("Good", 75.69);
    s2.display();
    ed.approve(s2);
    // LoanApplications[0]->approve(s2);
    Student s3("OKesh", 47.89);
    s3.display();
    ed.approve(s3);
    // LoanApplications[0]->approve(s3);
    Student s4("sad", 35.50);
    s4.display();
    ed.approve(s4);
    // LoanApplications[0]->approve(s4);

    PersonalLoan pl;
    HomeLoan hL;
    AutoLoan al;
    Employee e1("Happy", 1800000);
    e1.display();
    pl.approve(e1);
    hL.approve(e1);
    al.approve(e1);

    Employee e2("Good", 1150000);
    e2.display();
    pl.approve(e2);
    hL.approve(e2);
    al.approve(e2);

    Employee e3("OKesh", 656000);
    e3.display();
    pl.approve(e3);
    hL.approve(e3);
    al.approve(e3);

    Employee e4("sad", 550000);
    e4.display();
    pl.approve(e4);
    hL.approve(e4);
    al.approve(e4);

    Employee e5("Verysad", 325000);
    e5.display();
    pl.approve(e5);
    hL.approve(e5);
    al.approve(e5);

    return 0;
}
```

Output:

```
PS D:\Fullstack-Java-FirstBit-Solutions\Basic-C-and-CPP\CPP\Assignments\Assignment02\output> &
.\'q3studLoanApproval.exe'

Name of Student  :Happy
Percentage of Student  :84.99
Congratulations your Educational Loan of Rs. 200000.00 is approved...!

Name of Student  :Good
Percentage of Student  :75.69
Congratulations your Educational Loan of Rs. 100000.00 is approved...!
```

```
Name of Student  :OKesh
Percentage of Student  :47.89
Congratulations your Educational Loan of Rs. 50000.00 is approved...!

Name of Student  :sad
Percentage of Student  :35.5
We regret to say that your Educational loan couldn't be approved...!

Name of Employee  :Happy
Percentage of Employee  :1.8e+06
Congratulations your Personal Loan of Rs. 700000.00 is approved...!

Congratulations your Home Loan of Rs. 700000.00 is approved...!

Congratulations your Auto Loan of Rs. 700000.00 is approved...!

Name of Employee  :Good
Percentage of Employee  :1.15e+06
Congratulations your Personal Loan of Rs. 600000.00 is approved...!

Congratulations your Home Loan of Rs. 600000.00 is approved...!

Congratulations your Auto Loan of Rs. 600000.00 is approved...!

Name of Employee  :OKesh
Percentage of Employee  :656000
Congratulations your Personal Loan of Rs. 500000.00 is approved...!

Congratulations your Home Loan of Rs. 500000.00 is approved...!

Congratulations your Auto Loan of Rs. 500000.00 is approved...!

Name of Employee  :sad
Percentage of Employee  :550000
Congratulations your Personal Loan of Rs. 400000.00 is approved...!

Congratulations your Home Loan of Rs. 400000.00 is approved...!

Congratulations your Auto Loan of Rs. 400000.00 is approved...!

Name of Employee  :Verysad
Percentage of Employee  :325000
We regret to say that your loan couldn't be approved...!

We regret to say that your Home loan couldn't be approved...!

We regret to say that your Auto loan couldn't be approved...!
PS D:\Fullstack-Java-FirstBit-Solutions\Basic-C-and-CPP\CPP\Assignments\Assignment02\output>
```