**SHAURYA SRINET  
RA2111032010006**

**CSE IOT**

**WEEK 7 EXERCISE**

**Q1. Write a C++ program to read and print student information using two classes and simple inheritance.**

#include <iostream>

using namespace std;

// Base class

class std\_basic\_info {

private:

    char name[30];

    int age;

    char gender;

public:

    void getBasicInfo(void);

    void putBasicInfo(void);

};

// function definitions

void std\_basic\_info::getBasicInfo(void)

{

    cout << "Enter student's basic information:" << endl;

    cout << "Name?: ";

    cin >> name;

    cout << "Age?: ";

    cin >> age;

    cout << "Gender?: ";

    cin >> gender;

}

void std\_basic\_info::putBasicInfo(void)

{

    cout << "Name: " << name << ",Age: " << age << ",Gender: " << gender << endl;

}

// Derived class

class std\_result\_info : public std\_basic\_info {

private:

    int totalM;

    float perc;

    char grade;

public:

    void getResultInfo(void);

    void putResultInfo(void);

};

// Function definitions

void std\_result\_info::getResultInfo(void)

{

    cout << "Enter student's result information:" << endl;

    cout << "Total Marks?: ";

    cin >> totalM;

    perc = (float)((totalM \* 100) / 500);

    cout << "Grade?: ";

    cin >> grade;

}

void std\_result\_info::putResultInfo(void)

{

    cout << "Total Marks: " << totalM << ",Percentage: " << perc << ",Grade: " << grade << endl;

}

int main()

{

    // Create object of derived class

    std\_result\_info std;

    // Read student basic and result information

    std.getBasicInfo();

    std.getResultInfo();

    //print student basic and result information

    std.putBasicInfo();

    std.putResultInfo();

    return 0;

}

**Q2. Write a C++ program to read and print employee information using multiple inheritance.**

#include <iostream>

#include <stdio.h>

using namespace std;

// Base Class - basicInfo

class basicInfo {

protected:

    char name[30];

    int empId;

    char gender;

public:

    void getBasicInfo(void)

    {

        cout << "Enter Name: ";

        cin.getline(name, 30);

        cout << "Enter Emp. Id: ";

        cin >> empId;

        cout << "Enter Gender: ";

        cin >> gender;

    }

};

// Base Class - deptInfo

class deptInfo {

protected:

    char deptName[30];

    char assignedWork[30];

    int time2complete;

public:

    void getDeptInfo(void)

    {

        cout << "Enter Department Name: ";

        cin.ignore(1);

        cin.getline(deptName, 30);

        cout << "Enter assigned work: ";

        fflush(stdin);

        cin.getline(assignedWork, 30);

        cout << "Enter time in hours to complete work: ";

        cin >> time2complete;

    }

};

// final class (Derived Class)- employee

class employee : private basicInfo, private deptInfo {

public:

    void getEmployeeInfo(void)

    {

        cout << "Enter employee's basic info: " << endl;

        //call getBasicInfo() of class basicInfo

        getBasicInfo(); //calling of public member function

        cout << "Enter employee's department info: " << endl;

        //call getDeptInfo() of class deptInfo

        getDeptInfo(); //calling of public member function

    }

    void printEmployeeInfo(void)

    {

        cout << "Employee's Information is: " << endl;

        cout << "Basic Information...:" << endl;

        cout << "Name: " << name << endl; //accessing protected data

        cout << "Employee ID: " << empId << endl; //accessing protected data

        cout << "Gender: " << gender << endl

             << endl; //accessing protected data

        cout << "Department Information...:" << endl;

        cout << "Department Name: " << deptName << endl; //accessing protected data

        cout << "Assigned Work: " << assignedWork << endl; //accessing protected data

        cout << "Time to complete work: " << time2complete << endl; //accessing protected data

    }

};

int main()

{

    //create object of class employee

    employee emp;

    emp.getEmployeeInfo();

    emp.printEmployeeInfo();

    return 0;

}

**Q3. Write a C++ program to demonstrate an example of hierarchical inheritance to get square and cube of a number.**

#include <iostream>

using namespace std;

class Number {

private:

    int num;

public:

    void getNumber(void)

    {

        cout << "Enter an integer number: ";

        cin >> num;

    }

    //to return num

    int returnNumber(void)

    {

        return num;

    }

};

//Base Class 1, to calculate square of a number

class Square : public Number {

public:

    int getSquare(void)

    {

        int num, sqr;

        num = returnNumber(); //get number from class Number

        sqr = num \* num;

        return sqr;

    }

};

//Base Class 2, to calculate cube of a number

class Cube : public Number {

private:

public:

    int getCube(void)

    {

        int num, cube;

        num = returnNumber(); //get number from class Number

        cube = num \* num \* num;

        return cube;

    }

};

int main()

{

    Square objS;

    Cube objC;

    int sqr, cube;

    objS.getNumber();

    sqr = objS.getSquare();

    cout << "Square of " << objS.returnNumber() << " is: " << sqr << endl;

    objC.getNumber();

    cube = objC.getCube();

    cout << "Cube   of " << objS.returnNumber() << " is: " << cube << endl;

    return 0;

}

**Q4. Write a C++ program to read and print employee information with department and PF information using hierarchicalOI inheritance.**

#include <iostream>

#include <stdio.h>

using namespace std;

//Base Class - basicInfo

class basicInfo {

protected:

    char name[30];

    int empId;

    char gender;

public:

    void getBasicInfo(void)

    {

        cout << "Enter Name: ";

        cin.ignore(1);

        cin.getline(name, 30);

        cout << "Enter Emp. Id: ";

        cin >> empId;

        cout << "Enter Gender: ";

        cin >> gender;

    }

};

//Base Class - deptInfo

class deptInfo : private basicInfo {

protected:

    char deptName[30];

    char assignedWork[30];

    int time2complete;

public:

    void getDeptInfo(void)

    {

        getBasicInfo(); //to get basic info of an employee

        cout << "Enter Department Name: ";

        cin.ignore(1);

        cin.getline(deptName, 30);

        cout << "Enter assigned work: ";

        fflush(stdin);

        cin.getline(assignedWork, 30);

        cout << "Enter time in hours to complete work: ";

        cin >> time2complete;

    }

    void printDeptInfo(void)

    {

        cout << "Employee's Information is: " << endl;

        cout << "Basic Information...:" << endl;

        cout << "Name: " << name << endl; //accessing protected data

        cout << "Employee ID: " << empId << endl; //accessing protected data

        cout << "Gender: " << gender << endl

             << endl; //accessing protected data

        cout << "Department Information...:" << endl;

        cout << "Department Name: " << deptName << endl; //accessing protected data

        cout << "Assigned Work: " << assignedWork << endl; //accessing protected data

        cout << "Time to complete work: " << time2complete << endl; //accessing protected data

    }

};

//another Base Class : loadInfo

class loanInfo : private basicInfo {

protected:

    char loanDetails[30];

    int loanAmount;

public:

    void getLoanInfo(void)

    {

        getBasicInfo(); //to get basic info of an employee

        cout << "Enter Loan Details: ";

        cin.ignore(1);

        cin.getline(loanDetails, 30);

        cout << "Enter loan amount: ";

        cin >> loanAmount;

    }

    void printLoanInfo(void)

    {

        cout << "Employee's Information is: " << endl;

        cout << "Basic Information...:" << endl;

        cout << "Name: " << name << endl; //accessing protected data

        cout << "Employee ID: " << empId << endl; //accessing protected data

        cout << "Gender: " << gender << endl

             << endl; //accessing protected data

        cout << "Loan Information...:" << endl;

        cout << "Loan Details: " << loanDetails << endl; //accessing protected data

        cout << "Loan Amount : " << loanAmount << endl; //accessing protected data

    }

};

int main()

{

    //read and print department information

    deptInfo objD;

    objD.getDeptInfo();

    objD.printDeptInfo();

    cout << endl

         << endl;

    //read and print loan information

    loanInfo objL;

    objL.getLoanInfo();

    objL.printLoanInfo();

    return 0;

}

**Q7. Write a C++ Program to find area and volume using multiple inheritance**.

#include<iostream>

using namespace std;

class Circle      // First base class

{

        protected:

                float radius ;

        public:

                void Enter\_r(void)

                {

                        cout << "\n\t Enter the radius: ";  cin >> radius ;

                }

                void Display\_ca(void)

                {

                        cout << "\t The area = " << (22/7 \* radius\*radius) ;

                }

};

class Rectangle    // Second base class

{

                protected:

                        float length, breadth ;

                public:

                        void Enter\_lb(void)

                        {

                                cout << "\t Enter the length : "; cin >> length ;

                                cout << “\t Enter the breadth : ” ; cin >> breadth ;

                        }

                        void Display\_ar(void)

                        {

                                cout << "\t The area = " << (length \* breadth);

                        }

};

class Cylinder : public Circle, public Rectangle // Derived class, inherited

{                                               // from classes Circle & Rectangle

                public:

                        void volume\_cy(void)

                        {

                                cout << "\t The volume of the cylinder is: "

                                        << (22/7\* radius\*radius\*length) ;

                        }

};

int main()

{

                Circle c ;

                cout << "\n Getting the radius of the circle\n" ;

                c.Enter\_r( );

                c.Display\_ca( );

                Rectangle r ;

                cout << "\n\n Getting the length and breadth of the rectangle\n\n";

                r.Enter\_lb( );

                r.Display\_ar( );

                Cylinder cy ; // Object cy of the class cylinder which can access all the

                 // public members of the class circle as well as of the class rectangle

                cout << "\n\n Getting the height and radius of the cylinder\n";

                cy.Enter\_r( );

                cy.Enter\_lb( );

                cy.volume\_cy( );

return 0;

}