Assignment 03 Namespaces

Assignment: Namespace

Write a program to

1. Create a namespace “SY” which has class SYMARKS (Computer Total, MathsTotal, ElectronicsTotal).

2. Create another namespace “TY” which has a class TYMarks (Theory, Practical).

3. Create object of student class (Outside SY & TY package) having roll number, name, SYMakrs and TYMarks. Add the marksof SY and TY Computer subjects and calculate grade ("A" for >=70, "B" for >=60, "C" for >=50, “Pass Class” for >=40 else “Fail”) and display the result of the student in proper format.

Main.cpp

```
#include "student.h"
#include "sy.h"
#include "ty.h"
using namespace student;
using namespace sy;
using namespace ty;
int main()
{
    sy::Marks syMarks(45, 65, 43);
    ty::Marks tyMarks(75, 83);
    Student *s1 = new Student(12, "Bhagvat", syMarks, tyMarks);
    s1->calculateGrade();
    s1->display();
    return 0;
}
```

Student.h

```
#include <bits/stdc++.h>
// Create object of student class (Outside SY & TY package) having roll number, name,
SYMakrs and TYMarks. Add the marksof SY and TY Computer subjects and calculate grade ("A"
for >=70, "B" for >=60, "C" for >=50, “Pass Class” for >=40 else “Fail”) and display the
result of the student in proper format.
#include <string>
#include "sy.h"
#include "ty.h"
using namespace sy;
using namespace ty;
using namespace std;

namespace student
{

    class Student
    {
    private:
        int roll_no;
        string name;
        sy::Marks SyMarks;
        ty::Marks TyMarks;
        string grade;

    public:
        Student();
        Student(int, string, sy::Marks, ty::Marks);

        void calculateGrade();

        void display();
    };

}
```

Student.cpp

```
#include "student.h"

namespace student
{

    Student::Student()
    {
        this->name = "No Name";
        this->roll_no = 0;
        this->SyMarks;
        this->TyMarks;
        this->grade = "undefined";
    }
    Student::Student(int rollNo, string Name, sy::Marks syMarks, ty::Marks tyMarks)
        : roll_no(rollNo), name(Name), SyMarks(syMarks), TyMarks(tyMarks)
    {
    }

    // Student::Student(int rollNo, string Name, sy::Marks syMarks, ty::Marks tyMarks)
    // {
    //     this->roll_no = rollNo;
    //     this->name = Name;
    //     this->SyMarks = SyMarks;
    //     this->TyMarks = TyMarks;
    // }

    // void Student::display()
    // {
    //     cout << "Roll No : " << this->roll_no << endl;
    //     cout << "Name   : " << this->name << endl;
    //     cout << "SY Marks :: " << endl;
    //     cout << "CompTotal:" << this->SyMarks.getComputerTotal() << endl;
    //     cout << "ElectronicsTotal:" << this->SyMarks.getElectronicsTotal() << endl;
    //     cout << "MathsTotal:" << this->SyMarks.getMathsTotal() << endl;
    // }
    void Student::display()
    {
        cout << "Roll No : " << roll_no << endl;
        cout << "Name : " << name << endl;
        cout << "SY Marks: " << SyMarks.getComputerTotal() << ", "
             << SyMarks.getMathsTotal() << ", " << SyMarks.getElectronicsTotal() << endl;
        cout << "TY Marks: " << TyMarks.getTheory() << ", " << TyMarks.getPractical() <<
endl;
        cout << "Grade: " << grade << endl;
    }

    void Student::calculateGrade()
    {
        int total = SyMarks.getComputerTotal() + TyMarks.getPractical();
        total /= 2;
        if (total >= 70)
        {
            grade = "A";
        }
        else if (total >= 60)
        {
            grade = "B";
        }
        else if (total >= 50)
        {
            grade = "C";
        }
        else if (total >= 40)
        {
            grade = "Pass Class";
        }
        else
        {
            grade = "Fail";
        }
    }

}
```

SY.h

```cpp
#include <bits/stdc++.h>
#include <string>
using namespace std;
#pragma once
namespace sy
{

    class Marks
    {
    private:
        int ComputerTotal, MathsTotal, ElectronicsTotal;

    public:
        Marks();
        Marks(int, int, int);
        int getComputerTotal();
        int getMathsTotal();
        int getElectronicsTotal();
        void display();
    };

}
```

SY.cpp

```cpp
#include "sy.h"

namespace sy
{

    Marks::Marks()
    {
        this->ComputerTotal = 0;
        this->ElectronicsTotal = 0;
        this->MathsTotal = 0;
    }
    Marks::Marks(int CompTTL, int EleTTL, int MathTTL)
    {
        this->ComputerTotal = CompTTL;
        this->ElectronicsTotal = EleTTL;
        this->MathsTotal = MathTTL;
    }
    int Marks::getComputerTotal()
    {
        return this->ComputerTotal;
    }
    int Marks::getElectronicsTotal()
    {
        return this->ElectronicsTotal;
    }
    int Marks::getMathsTotal()
    {
        return this->MathsTotal;
    }
    void Marks::display()
    {
        cout << ":: SY TOTAL MARKS ::\n";
        cout << "ComputerTotal = " << this->ComputerTotal << endl;
        cout << "ElectronicsTotal = " << this->ElectronicsTotal << endl;
        cout << "MathsTotal = " << this->MathsTotal << endl;
    }

}
```

TY.h

```
#include <bits/stdc++.h>
#include <string>
using namespace std;
#pragma once
namespace ty
{

    class Marks
    {
    private:
        int Theory, Practical;

    public:
        Marks();
        Marks(int, int);
        int getTheory();
        int getPractical();
        // void display();
    };

}
```

Ty.cpp

```
#include "ty.h"

namespace ty
{

    Marks::Marks()
    {
        this->Theory = 0;
        this->Practical = 0;
    }
    Marks::Marks(int Theory, int Practical)
    {
        this->Theory = Theory;
        this->Practical = Practical;
    }
    int Marks::getTheory()
    {
        return this->Theory;
    }
    int Marks::getPractical()
    {
        return this->Practical;
    }

    // void Marks::display()
    // {
    //     cout << ":: SY TOTAL MARKS ::\n";
    //     cout << "Theory = " << this->Theory << endl;
    //     cout << "Practical = " << this->Practical << endl;
    //     cout << "MathsTotal = " << this->MathsTotal << endl;
    // }

}
```

OUTPUT:

```
PS D:\Fullstack-Java-FirstBit-Solutions\DSA\Assignments\NameSpacs\SyTyGr./main

Roll No : 12

Name : Bhagvat

SY Marks: 45, 43, 65

TY Marks: 75, 83

Grade: B

PS D:\Fullstack-Java-FirstBit-Solutions\DSA\Assignments\NameSpacs\SyTyGrades>
```