Coding (Or Script) - 6

CIS 266-003 C++ OOP

Instructor: Jim Yung

LAB 6

STUDENT: KEJIAN WU

```
//  CIS 266-003 C++ OOP   Instructor: Jim Yung
//  STUDENT:  KEJIAN WU
//  Lab6

//  Description of Program:
//  Write a small program to calculate the grade for students in the class.

//  Learned from this exercise:  Objected Oriented Programming
//                               (Data abstraction, Encapsulation)
//                               Flow control using "exceptions"
//                               Static variables, Static functions

//  Date:  10/26/98


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

// Declarations
class Student {
private:
    char name[30];
    float Quiz1;
    float Quiz2;
    float Final;
    float Lab;
    float Average;
    float InputScore(void);

    static char *teacherName;
    static char *Course_Des;
public:
    void getStudentInfo();
    void printStudentInfo();
    float getScore(char * ScoreType );
    char* getName(void)  { return name; };

    static int numOfStudent;
    static void getTeacherInfo();
    static void printTeacherInfo();
};


class ClassStat {
    Student* pHigh;
    Student* pLow;
    float    average;
    int      numStud;
public:
    //constructor
    ClassStat()  { pHigh=0; pLow=0; numStud=0; }

    void collectStat( Student* pStud );

    void printStat();
};


int Student::numOfStudent=0;
```

```
char *Student::teacherName = new char[30];
char *Student::Course_Des= new char[50];

// main program
void main()
{
    Student stud[30];  //define an array of 30 students.
    ClassStat clsSt;

    Student::getTeacherInfo();
    Student::printTeacherInfo();

    for(int i=0; i<Student::numOfStudent; i++)
    {
        stud[i].getStudentInfo();
    }

    for(int j=0; j<Student::numOfStudent; j++)
    {
        stud[j].printStudentInfo();
    }

    for( int k=0; k<Student::numOfStudent; k++ )
    {
        clsSt.collectStat( &stud[k] );
    }
    clsSt.printStat();
}

// Definitions
void  Student::getTeacherInfo()  // To get teacher information
{

    cout << "Course Description: ";
    cin.getline(Student::Course_Des,34);

    cout << "Instructor: ";
    cin.getline(Student::teacherName,29);

    cout << "Number of Students: ";
    cin >> Student::numOfStudent;


    cin.ignore();
}

void Student::printTeacherInfo()  // To print teachet information
{
    cout << '\n' ;
    cout << "Course Description: " << Student::Course_Des << '\n' << endl ;
    cout << "Instructor:  " << Student::teacherName << '\n' << endl ;
    cout << "Number of Students: " << Student::numOfStudent << '\n' << endl ;
}

float Student::getScore(char * ScoreType )
{
    if ( ScoreType == NULL )
        return ( Average );
    else if ( strcmp( ScoreType, "Quiz1") ==0 )
        return ( Quiz1 );
```

```
    else if (  strcmp( ScoreType, "Quiz2") ==0 )
        return ( Quiz2 );
    else if (  strcmp( ScoreType, "Final") ==0 )
        return ( Final );
    else if (  strcmp( ScoreType, "Lab") ==0 )
        return ( Lab );
    else
        return ( Average );
}

void Student::getStudentInfo() // To get student information
{
    cout << "Student Name: ";
    cin.getline(name,29);

    cout << "Quiz1: ";
        Quiz1 = InputScore();

    cout << "Quiz2: ";
        Quiz2 = InputScore();

    cout << "Final: ";
        Final = InputScore();

    cout << "Lab  : ";
        Lab = InputScore();

    Average = ( Quiz1 + Quiz2 + Final + Lab ) / 4 ;

    cin.ignore();  // to flush input buffer
}

void Student::printStudentInfo()  // To print student information
{
    cout << "Student Name: " << name << endl ;
    cout << "Quiz1              " << Quiz1 << endl ;
    cout << "Quiz2              " << Quiz2 << endl ;
    cout << "Final              " << Final << endl ;
    cout << "LAb                " << Lab << endl ;
    cout << "----------------- " << endl ;
    cout << "Average            " << Average << endl ;
    cout << '\n';
}


float Student::InputScore(void)
{
    float i;

    int InputError = 0;

    do {
        cin >> i;
        try
        {
            if ( (i < 0 ) || ( i > 100 )  )  {
                throw i;
            }
            InputError= 0;
        }
```

```
        catch ( float i ) {
            cout << "Please try again( 0 <= score && score <= 100)" << endl ;
            InputError= 1;
        }
    } while ( InputError );

    return i;
}

void ClassStat::collectStat( Student* pStud )
{
    if ( pHigh != 0 )  {
        if ( pStud->getScore(NULL) > pHigh->getScore(NULL) )
            pHigh= pStud;
    } else
        pHigh= pStud;

    if ( pLow != 0 )  {
        if ( pStud->getScore(NULL) < pLow->getScore(NULL) )
            pLow= pStud;
    } else
        pLow= pStud;

    average= (average*numStud + pStud->getScore(NULL))/(numStud+1);
    numStud++;
}

void ClassStat::printStat()
{
    cout << "High    : " << pHigh->getName() << " : " << pHigh->getScore(NULL) << endl;
    cout << "Low     : " << pLow->getName() << " : " << pLow->getScore(NULL) << endl;
    cout << "Average: " << average << endl;
}
```


Lab6
Auto
Number of Students: 1
Course Description: C++ OOP
Instructor: Dr. Chen
Number of Students: 1
Student Name: ken
Quiz1: 90
Quiz2: 80
Final: 100
Lab : 100
Student Name: ken
Quiz1 90
Quiz2 80
Final 100
LAb 100
Average 92.5
High : ken : 92.5
Low : ken : 92.5
Average: 92.5
Press any key to continue