//

// main.cpp

// as8

//

// Created by Jeff on 10/8/16.

// Copyright © 2016 Jeff zhang. All rights reserved.

//

// Assignment 8 Esay Class test driver

// Assignment 8 Esay Class test driver

#include <iostream>

#include "Essay.h"

using namespace std;

int main()

{

GradedActivity g85(85), g65(65);

Essay report85(25, 20, 20, 20), report65(15, 10, 20, 20);

cout << "g85(85):"

<< "\nGraded Score: " << g85.getScore()

<< ", Grade: " << g85.getLetterGrade() << endl;

cout << "\nreport85(25, 20, 20, 20): \n" << report85.toString()

<< "\nReport Score: " << report85.getScore()

<< ", Grade: " << report85.getLetterGrade() << endl;

cout << "\ng65(65):"

<< "\nGraded Score: " << g65.getScore()

<< ", Grade: " << g65.getLetterGrade() << endl;

cout << "\nreport65(15, 10, 20, 20): \n" << report65.toString()

<< "\nReport Score: " << report65.getScore()

<< ", Grade: " << report65.getLetterGrade() << endl;

return 0;

}

//

// Essay.h

// as8

//

// Created by Jeff on 10/8/16.

// Copyright © 2016 Jeff zhang. All rights reserved.

//

#ifndef Essay\_h

#define Essay\_h

#include <sstream>

#include <string>

#include "GradedActivity.h"

using namespace std;

class Essay :public GradedActivity{

private:

int grammer ,spelling ,length ,content ;

public:

Essay(){

grammer = 0;

spelling =0;

length = 0;

content = 0;

}

Essay(int g,int s,int l,int c){

setGrammer(g);

setSpelling(s);

setLength(l);

setContent(c);

setScore(g,s,l,c);

}

void setGrammer(int g){

grammer = g;

}

int getGrammer(){

return grammer;

}

void setSpelling(int s){

spelling = s;

}

int getspelling(){

return spelling;

}

void setLength(int l){

length = l;

}

int getlenght(){

return length;

}

void setContent(int c){

content = c;

}

int getcontent(){

return content;

}

void setScore(int g,int s, int l, int c)

{ score = s + g + l +c; }

// Accessor functions

double getScore() const

{ return score; }

string toString();

string getLetterGrade();

friend ostream &operator << (ostream &, const Essay &);

};

string Essay::toString(){

stringstream ss;

ss << "\n Grammer Point: " << this->getGrammer() << " \n Spelling points:"

<<this->getspelling() << "\n Length points: " << this->getlenght()

<<" \n Content points: " << this->getcontent() <<endl;

return ss.str();

}

string Essay::getLetterGrade(){

stringstream ss;

if(score>80)

ss<<"A" << endl;

else if (score >65)

ss<<"B";

else

ss<<"c" << endl;

return ss.str();

}

#endif /\* Essay\_h \*/

//

// GradedActivity.h

// as8

//

// Created by Jeff on 10/8/16.

// Copyright © 2016 Jeff zhang. All rights reserved.

//

#ifndef GradedActivity\_h

#define GradedActivity\_h

// This program demonstrates the GradedActivity class.

#include <iostream>

#include "GradedActivity.h"

using namespace std;

class GradedActivity

{

protected:

double score; // To hold the numeric score

public:

// Default constructor

GradedActivity()

{ score = 0.0; }

// Constructor

GradedActivity(double s)

{ score = s; }

// Mutator function

void setScore(double s)

{ score = s; }

// Accessor functions

double getScore() const

{ return score; }

//char getLetterGrade() const;

virtual char getLetterGrade() const

{

char letterGrade; // To hold the letter grade

if (score > 89)

letterGrade = 'A';

else if (score > 79)

letterGrade = 'B';

else if (score > 69)

letterGrade = 'C';

else if (score > 59)

letterGrade = 'D';

else

letterGrade = 'f';

return letterGrade;

}

};

#endif /\* GradedActivity\_h \*/

