

```
1 //Ben Scherer
2 // 6/22/2017
3 // Averaging Quiz Scores
4 // Writes scores for individual students to file, then reads in file to calculate average score
5
6 //Headers to include
7 #include <iostream> //cout
8 #include <iomanip> // used to manipulate cout
9 #include <string> //needed for string variable
10 #include <math.h> //used for basic arithmetic
11 #include <limits> //user for numeric_limits
12 #include <fstream> //used for file access
13
14 using namespace std;
15
16 //Functions
17 void pauseProgram(); //Pauses program until user hits enter key
18 bool populateQuizData(string fileName,int numQuiz); //Gets user input and writes to file. Returns bool
19 int getInput(string questionToAsk, string errorMsg); //Basic input validation, returns int
20 int getInput(string questionToAsk, string errorMsg, int lowRange, int highRange); //Input validation for int that must fall within a numeric range. Returns int
21 double calculateClassAverage(int numQuiz, string fileName); //Reads data from file and calculates average score. Returns double
22 char getSentinel(); //Input validation for sentinel. simple y/n
23
24
25 //Main function of program
26 int main() {
27     //Variables
28     string fileName; //Filename to be used for output/input
29     int numQuiz; //Number of scores to input
30     double classAvg; //Average of all scores
31
32     //Request file name to use
33     cout << "Please enter file name to use:";
34     cin >> fileName;
35
36     //Request number of quizzes to input for each student
37     numQuiz = getInput("Number of scores to enter per student", "ERROR: Please Enter a valid number"); //Get user input and generate quiz results file
38
39     if (!populateQuizData(fileName, numQuiz)) { return 1; } // if unable to open file for write, exit program
40
41     //Calculate average and output results
42     cout << "Calculating average score for class.....\n";
43     classAvg = calculateClassAverage(numQuiz, fileName); //Calculate average
44     cout << "Average Score for class: " << setprecision(2) << fixed << classAvg
```

```
<< endl; //Output, 2 decimal
45
46     pauseProgram(); //Pause program so that results can be viewed
47     return 0;
48 }
49
50 //Simple function. Pauses program execution and waits for input
51 void pauseProgram() {
52     //pauses program
53     cout << "Press enter key to exit program\n";
54     cin.ignore(numeric_limits<streamsize>::max(), '\n');
55     cin.get();
56 }
57
58
59 // Populates file with quiz data from userInput
60 bool populateQuizData(string fileName,int numQuiz) {
61     ofstream outFile(fileName);
62     if (!outFile) {
63         cout << "ERROR: Unable to open file for write\n";
64         return false;
65     }
66
67     //User input loop. Loops until sentinel is reached. inner for loop used for ↗
        score input
68     do {
69         outFile << getInput("Enter Student ID number: ","ERROR: Please Enter a ↗
            valid student id");
70         for (int i = 1; i <= numQuiz; i++) {
71             outFile << "\t" << getInput("Enter score for quiz #" + to_string(i) + ↗
                ": ", "ERROR: Please enter a valid score",0,100);
72         }
73         outFile << endl;
74     } while (getSentinel() == 'y');
75
76     outFile.close();
77     return true;
78 }
79
80 //simple input validation. returns int
81 int getInput(string questionToAsk,string errorMsg) {
82     int score;
83     cout << questionToAsk ;
84     while (!(cin >> score)) {
85         cout << errorMsg << endl;
86         cin.clear();
87         cin.ignore(numeric_limits<streamsize>::max(), '\n');
88     }
89     return score;
90 }
91
92 //Validates input based on a range. returns int
```

```
93 int getInput(string questionToAsk, string errorMsg, int lowRange, int highRange) ↗
94 {
95     int score;
96     cout << questionToAsk ;
97     while (!(cin >> score) || score < lowRange || score > highRange) { //Loop ↗
98         until integer in the specified range is entered
99         cout << errorMsg << endl;
100         cin.clear();
101         cin.ignore(numeric_limits<streamsize>::max(), '\n');
102     }
103     return score;
104 }
105 //validates sentinel input, then returns char value
106 char getSentinel() {
107     char varToReturn;
108     bool isValidInput = false;
109
110     // loop until a valid y or n char is entered
111     do {
112         cout << "Enter 'y' to add more student scores. Enter 'n' if you are ↗
113             finished" << endl;
114         if (!(cin >> varToReturn) || (tolower(varToReturn) != 'y' && tolower ↗
115             (varToReturn) != 'n')) {
116             cout << "\tERROR: Enter 'y' to add more student scores. Enter 'n' if ↗
117                 you are finished\n";
118             cin.clear();
119             cin.ignore(numeric_limits<streamsize>::max(), '\n');
120         }
121         else {
122             isValidInput = true;
123         }
124     } while (!isValidInput);
125     return tolower(varToReturn);
126 }
127 //Calculate class average from scores file, then return double value
128 double calculateClassAverage(int numQuiz, string fileName) {
129     //Attempt to open file for read. Throw error and exit program if open fails
130     ifstream inFile(fileName);
131     if (!inFile) {
132         cout << "ERROR: Unable to open file for write\n";
133         exit(1);
134     }
135
136     int classTotal = 0; //Sum of all quiz scores
137     int score; // temporary place holder for scores as read from file
138     int studentid; //student id field, used for loop
139     int quizCounter = 0; //Total number of scores read from file
```

```
140
141     //Read test data. Assumes first field is student id, then <numQuiz> scores
142     while (inFile >> studentid) { //Loop until end of file
143
144         for (int i = 1; i <= numQuiz; i++) { //read in test scores for student and
            sum
145             inFile >> score;
146             classTotal += score;
147             quizCounter++;
148         }
149
150     }
151     return (double(classTotal) / quizCounter); //Return average of all quiz
        scores
152 }
153
154
```