package com.example;

import java.io.FileWriter;

import java.io.IOException;

public class ScoreLogger {

private final String filePath;

public ScoreLogger(String filePath) {

this.filePath = filePath;

}

public synchronized void logScore(Student student) throws IOException {

try (FileWriter writer = new FileWriter(filePath, true)) {

writer.write(student.toString());

}

}

}

package com.example;

public class Main {

public static void main(String[] args) {

String filePath = "scores.csv";

ScoreLogger logger = new ScoreLogger(filePath);

Student s1 = new Student("Alice", 101, 88);

Student s2 = new Student("Bob", 102, 92);

Student s3 = new Student("Charlie", 103, 79);

Thread t1 = new StudentThread(s1, logger);

Thread t2 = new StudentThread(s2, logger);

Thread t3 = new StudentThread(s3, logger);

t1.start();

t2.start();

t3.start();

try {

t1.join();

t2.join();

t3.join();

} catch (InterruptedException e) {

e.printStackTrace();

}

System.*out*.println(" All scores logged successfully in " + filePath);

}

}

package com.example;

public class Student {

private String name;

private int rollNumber;

private int marks;

public Student(String name, int rollNumber, int marks) {

this.name = name;

this.rollNumber = rollNumber;

this.marks = marks;

}

@Override

public String toString() {

return name + "," + rollNumber + "," + marks + "\n";

}

}

package com.example;

public class StudentThread extends Thread {

private final Student student;

private final ScoreLogger logger;

public StudentThread(Student student, ScoreLogger logger) {

this.student = student;

this.logger = logger;

}

@Override

public void run() {

try {

logger.logScore(student);

System.*out*.println(" Score submitted: " + student);

} catch (Exception e) {

e.printStackTrace();

}

}

}

package com.example;

import org.junit.\*;

import java.io.\*;

import static org.junit.Assert.\*;

public class ScoreLoggerTest {

private static final String *TEST\_FILE* = "extra\_scores.csv";

@Before

public void setUp() throws IOException {

new FileWriter(*TEST\_FILE*, false).close(); // clear file

}

@Test

public void testMultipleSequentialLogs() throws IOException {

ScoreLogger logger = new ScoreLogger(*TEST\_FILE*);

Student s1 = new Student("Alice", 101, 88);

Student s2 = new Student("Bob", 102, 92);

logger.logScore(s1);

logger.logScore(s2);

BufferedReader reader = new BufferedReader(new FileReader(*TEST\_FILE*));

String line1 = reader.readLine();

String line2 = reader.readLine();

reader.close();

*assertEquals*("Alice,101,88", line1);

*assertEquals*("Bob,102,92", line2);

}

@Test(expected = IOException.class)

public void testInvalidFilePathThrowsException() throws IOException {

ScoreLogger logger = new ScoreLogger("/invalid\_path/test.csv");

Student s = new Student("Eve", 104, 95);

logger.logScore(s); // should throw IOException

}

@After

public void tearDown() {

new File(*TEST\_FILE*).delete();

}

}