SWE 206
Lab12: Unit Testing and Junit
Section 53
Semester 232

Abdulmajeed AlOthman
Abdullah Al Abbas – 202156190

Table for test cases

Test Case	loanPeriod	Subject	Expected Fine
Case 1	**10**	ICS	**0.00**
Case 2	**10**	PHYSICS	**0.00**
Case 3	**15**	ICS	**35.00**
Case 4	**15**	PHYSICS	**15.00**
Case 5	**15**	MATH	**35.00**
Case 6	**100**	HISTORY	**500.00**
Case 7	**14**	ICS	**0.00**
Case 8	**14**	PHYSICS	**0.00**
Case 9	**0**	ICS	**0.00**
Case 10	**1**	PHYSICS	**0.00**
Case 11	**14**	MATH	**0.00**
Case 12	**15**	MATH	**35.00**
Case 13	**14**	ICS	**0.00**
Case 14	**14**	PHYSICS	**0.00**
Case 15	**28**	CS	**0.00**
Case 16	**28**	PHYSICS	**0.00**
Case 17	**28**	MATH	**70.00**
Case 18	**26**	CS	**100.00**
Case 19	**26**	PHYSICS	**0.00**
Case 20	**26**	MATH	**70.00**

```java
public class Library {
    private static final double FINE_PER_DAY = 5.00;
    private static final double MAX_FINE = 500.00;
    private static final double CS_FINE_INCREASE = 0.20;
    private static final double PHYSICS_FINE_DECREASE = 0.20;
    private static final int allowedLoanPeriod = 14;

    public static double calculateFine(int loanPeriod, String subject) {
        double fine = 0.00;

        if (loanPeriod > allowedLoanPeriod) {
            fine = (loanPeriod - allowedLoanPeriod) * FINE_PER_DAY;
            if (subject.equals("CS")) {
                fine += fine * CS_FINE_INCREASE;
            } else if (subject.equals("Physics")) {
                fine -= fine * PHYSICS_FINE_DECREASE;
            }
        }

        return Math.min(fine, MAX_FINE);
    }
}

```

```java
import static org.junit.Assert.assertEquals;
import org.junit.Test;

public class LibraryTest {
    @Test
    public void testCalculateFine() {
        // Cases for loanPeriod <= allowedLoanPeriod
        assertEquals(0.00, Library.calculateFine(0, "CS"), 0.001); // Case 9
        assertEquals(0.00, Library.calculateFine(1, "Physics"), 0.001); // Case 10
        assertEquals(0.00, Library.calculateFine(14, "Math"), 0.001); // Case 11
        assertEquals(35.00, Library.calculateFine(15, "Math"), 0.001); // Case 12

        // Cases for loanPeriod > allowedLoanPeriod
        assertEquals(0.00, Library.calculateFine(14, "CS"), 0.001); // Case 13
        assertEquals(0.00, Library.calculateFine(14, "Physics"), 0.001); // Case 14
        assertEquals(35.00, Library.calculateFine(15, "CS"), 0.001); // Case 15
        assertEquals(15.00, Library.calculateFine(15, "Physics"), 0.001); // Case 16
        assertEquals(35.00, Library.calculateFine(15, "Math"), 0.001); // Case 17
        assertEquals(500.00, Library.calculateFine(100, "History"), 0.001); // Case 18
        assertEquals(0.00, Library.calculateFine(28, "CS"), 0.001); // Case 19
        assertEquals(0.00, Library.calculateFine(28, "Physics"), 0.001); // Case 20
    }
}

```