

測試執行結果

Test Results	50ms	/Users/alvin/Library/Java/JavaVirtualMachines/adopt-openjdk-15.0.2/Contents/Home/bin/java ...
GradesTest	50ms	
tooMuch()	32ms	Process finished with exit code 0
good()	5ms	
empty()	11ms	
minus()	2ms	

Grades.java

```
3 import java.util.Arrays;
4
5 public class Grades {
6     @
7     double[] averageScore(int[] math, int[] chinese) throws Exception {
8         if(math.length == 0 || chinese.length == 0){
9             throw new Exception("Empty Array");
10        }
11        double sum=0;
12        double[] ans = new double[2];
13        for (int i : math) {
14            if (i >100 || i<0){
15                throw new Exception("Out Range Grade");
16            }
17            sum += i;
18        }
19        ans[0] = sum/math.length;
20        sum=0;
21        for(int i:chinese){
22            if (i >100 || i<0){
23                throw new Exception("Out Range Grade");
24            }
25            sum += i;
26        }
27        ans[1] = sum/ chinese.length;
28        return ans;
29    }
30    public static void main(String[] args) throws Exception {
31        Grades g = new Grades();
32        System.out.println(Arrays.toString(g.averageScore(new int[]{1, 2, 3}, new int[]{3, 4, 5})));
33    }
34 }
```

GradesTest.java

```
1 import org.junit.jupiter.api.*;
2
3 import static org.junit.jupiter.api.Assertions.*;
4
5 class GradesTest {
6     Grades g;
7
8     @BeforeEach
9     void set() {
10         g = new Grades();
11     }
12
13     @Test
14     void minus() throws Exception {
15         assertThrows(Exception.class, () -> g.averageScore(
16             new int[]{60, 50, -30},
17             new int[]{30, 10, 50}
18         ));
19         assertThrows(Exception.class, () -> g.averageScore(
20             new int[]{60, 50, 30},
21             new int[]{30, 10, -50}
22         ));
23     }
24
25     @Test
26     void good() throws Exception {
27         assertEquals(
28             new double[]{60, 70},
29             g.averageScore(
30                 new int[]{50, 60, 70},
31                 new int[]{60, 70, 80}
32             )
33         );
34         assertEquals(
35             new double[]{0, 70},
36             g.averageScore(
37                 new int[]{0, 0, 0},
38                 new int[]{60, 70, 80}
39             )
40         );
41     }
42
43     @Test
44     void empty() throws Exception {
45         assertThrows(Exception.class, () -> g.averageScore(
46             new int[] {},
47             new int[]{1, 2, 3}
48         ));
49         assertThrows(Exception.class, () -> g.averageScore(
50             new int[]{1, 2, 3},
51             new int[] {}
52         ));
53         assertThrows(Exception.class, () -> g.averageScore(
54             new int[] {},
55             new int[] {}
56         ));
57     }
58
59     @Test
60     void tooMuch() throws Exception {
61         assertThrows(Exception.class, () -> g.averageScore(
62             new int[]{60, 50, 101},
63             new int[]{30, 10, 50}
64         ));
65         assertThrows(Exception.class, () -> g.averageScore(
66             new int[]{60, 50, 0},
67             new int[]{30, 10, 101}
68         ));
69     }
70 }
```