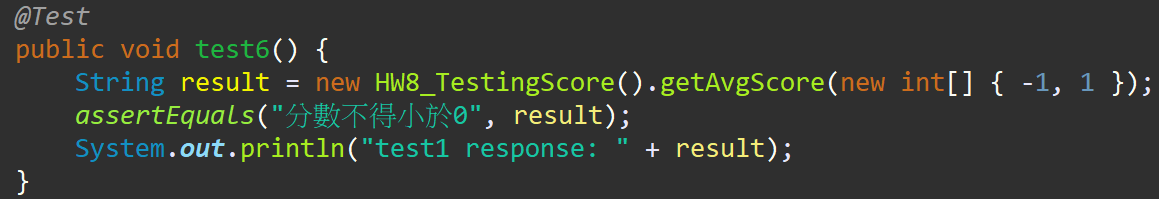
**Boundary value 邊界值測試**

**有效範圍0~100**

test case 6 (測試案例 6)

1. Input values(測試時候需要控制資料的值):

int[] scoreArr :{ -1, 1 }

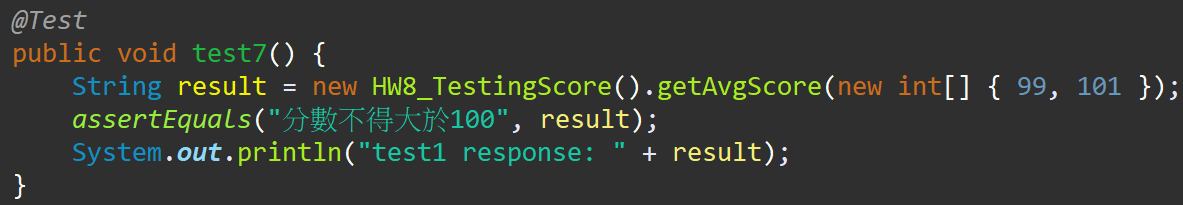
2) expected result(正確的結果):分數不得小於0

3) test program's result(程式執行後的結果):

分數不得小於0

4) criteria analysis(測試目標的分析)

測試值:-1,1



test case 7 (測試案例 7)

1. Input values(測試時候需要控制資料的值):

int[] scoreArr :{ 99, 101 }

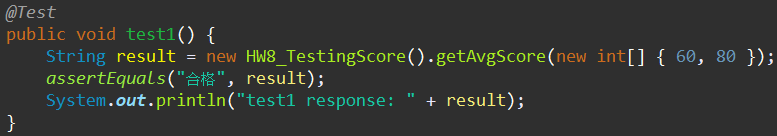
2) expected result(正確的結果): 分數不得大於100

3) test program's result(程式執行後的結果):

分數不得大於100

4) criteria analysis(測試目標的分析)

測試值:99,101

test case 1 (測試案例 1)

1. Input values(測試時候需要控制資料的值):

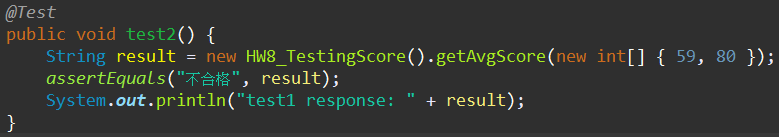
int[] scoreArr :{ 60, 80 }

2) expected result(正確的結果): 合格

3) test program's result(程式執行後的結果): 合格

4) criteria analysis(測試目標的分析)

測試值:60,80



test case 2 (測試案例 2)

1. Input values(測試時候需要控制資料的值):

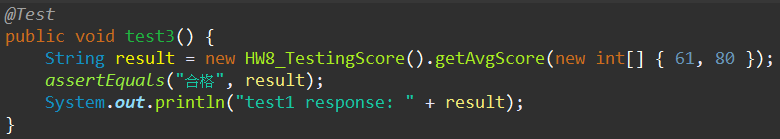
int[] scoreArr :{ 59, 80 }

2) expected result(正確的結果): 不合格

3) test program's result(程式執行後的結果): 不合格

4) criteria analysis(測試目標的分析)

測試值:59,80

test case 3 (測試案例 3)

1. Input values(測試時候需要控制資料的值):

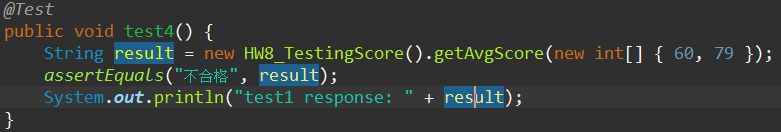
int[] scoreArr :{ 61, 80 }

2) expected result(正確的結果): 合格

3) test program's result(程式執行後的結果): 合格

4) criteria analysis(測試目標的分析)

測試值:61,80



test case 4 (測試案例 4)

1. Input values(測試時候需要控制資料的值):

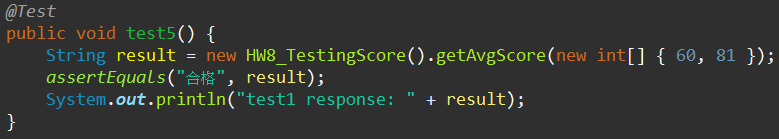
int[] scoreArr :{ 60, 79 }

2) expected result(正確的結果): 不合格

3) test program's result(程式執行後的結果): 不合格

4) criteria analysis(測試目標的分析)

測試值:60,79

test case 5 (測試案例 5)

1. Input values(測試時候需要控制資料的值):

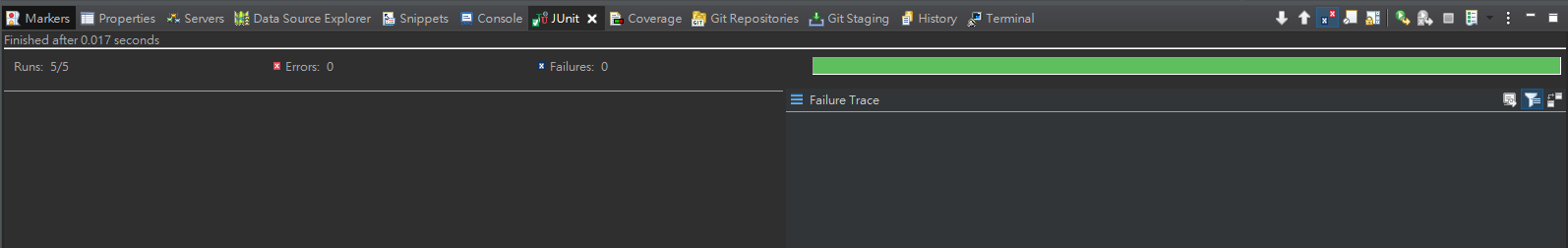
int[] scoreArr :{ 60, 81 }

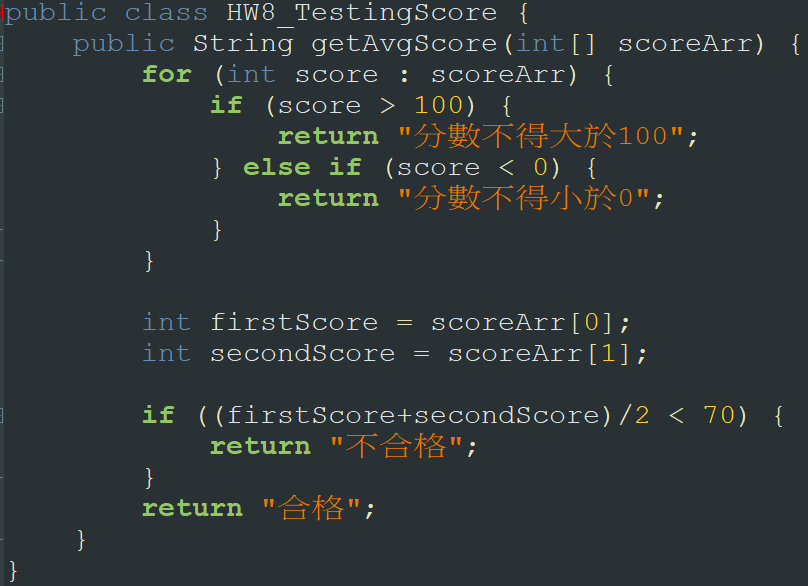
2) expected result(正確的結果): 合格

3) test program's result(程式執行後的結果): 合格

4) criteria analysis(測試目標的分析)

測試值:60,81

junit結果：

程式碼：