D0745378 薛竣祐 M6.2

Swim.java

public class Swim {  
 private static final int *NORMAL\_PRICE* = 200;  
 private static final int *WEEKEND\_PRICE* = 250;  
 double getPrice (boolean is\_member,boolean is\_group,int time,int week,int age) throws Exception {  
 if(time>22 || time<5) throw new Exception("Not Open");  
 if(age<3 || age>75) throw new Exception("Not Open");  
 double discount = 1;  
 if (week == 6 || week == 7) {  
 discount = is\_member ? 0.5 : 1;  
 return *WEEKEND\_PRICE* \* discount;  
 }  
 if (time < 7) discount = 0.8;  
 if (age <=12 || age>=60) discount = 0.8;  
 if (is\_group) discount = 0.7;  
 if (is\_member) discount = 0.5;  
  
 return *NORMAL\_PRICE* \* discount;  
 }  
}

SwimTest.java (弱涵蓋)

public class SwimTest {  
 Swim swim;  
 @BeforeEach  
 void set(){  
 swim = new Swim();  
 }  
 @Test  
 void priceTestWeak1() throws Exception {  
 *assertEquals*(100,swim.getPrice(true,false,12,4,25));  
 *assertEquals*(200,swim.getPrice(false,false,12,4,25));  
 }  
 @Test  
 void priceTestWeak2() throws Exception {  
 *assertEquals*(140,swim.getPrice(false,true,12,4,25));  
 *assertEquals*(200,swim.getPrice(false,false,12,4,25));  
 }  
 @Test  
 void priceTestWeak3() throws Exception {  
 *assertEquals*(160,swim.getPrice(false,false,6,4,25));  
 *assertEquals*(200,swim.getPrice(false,false,12,4,25));  
 }  
 @Test  
 void priceTestWeak4() throws Exception {  
 *assertEquals*(250,swim.getPrice(false,false,12,6,25));  
 *assertEquals*(200,swim.getPrice(false,false,12,4,25));  
 }  
 @Test  
 void priceTestWeak5() throws Exception {  
 *assertEquals*(160,swim.getPrice(false,false,12,4,10));  
 *assertEquals*(200,swim.getPrice(false,false,12,4,25));  
 }  
}

涵蓋度測試：41%

一張含有 桌 的圖片

自動產生的描述

SwimTest.java(100%)

public class SwimTest {  
 Swim swim;  
 @BeforeEach  
 void set(){  
 swim = new Swim();  
 }  
 @Test  
 void priceTestWeak1() throws Exception {  
 *assertEquals*(100,swim.getPrice(true,false,12,4,25));  
 *assertEquals*(200,swim.getPrice(false,false,12,4,25));  
 }  
 @Test  
 void priceTestWeak2() throws Exception {  
 *assertEquals*(140,swim.getPrice(false,true,12,4,25));  
 *assertEquals*(200,swim.getPrice(false,false,12,4,25));  
 }  
 @Test  
 void priceTestWeak3() throws Exception {  
 *assertEquals*(160,swim.getPrice(false,false,6,4,25));  
 *assertEquals*(200,swim.getPrice(false,false,12,4,25));  
 }  
 @Test  
 void priceTestWeak4() throws Exception {  
 *assertEquals*(250,swim.getPrice(false,false,12,6,25));  
 *assertEquals*(200,swim.getPrice(false,false,12,4,25));  
 }  
 @Test  
 void priceTestWeak5() throws Exception {  
 *assertEquals*(160,swim.getPrice(false,false,12,4,10));  
 *assertEquals*(200,swim.getPrice(false,false,12,4,25));  
 }  
 @Test  
 void priceTestStrong1() {  
 *assertThrows*(Exception.class, ()-> {  
 swim.getPrice(false, false, 4, 4, 25);  
 });  
 *assertThrows*(Exception.class, ()-> {  
 swim.getPrice(false, false, 23, 4, 25);  
 });  
  
 }  
 @Test  
 void priceTestStrong2() {  
 *assertThrows*(Exception.class, ()-> {  
 swim.getPrice(false, false, 12, 4, 2);  
 });  
 *assertThrows*(Exception.class, ()-> {  
 swim.getPrice(false, false, 12, 4, 76);  
 });  
  
 }  
 @Test  
 void priceTestStrong3() throws Exception {  
 *assertEquals*(125,swim.getPrice(true,false,12,6,25));  
 *assertEquals*(250,swim.getPrice(false,true,12,6,25));  
 *assertEquals*(250,swim.getPrice(false,false,6,7,25));  
 *assertEquals*(250,swim.getPrice(false,false,12,7,10));  
 }  
 @Test  
 void priceTestStrong4() throws Exception {  
 *assertEquals*(160,swim.getPrice(false,false,12,4,60));  
 }  
}

涵蓋度：100%

一張含有 桌 的圖片

自動產生的描述

心得

弱涵蓋測試可以快速的把大部分的分支都走過一遍，能夠快速的簡單測試程式的錯誤可能性，但沒辦法讓可能前後分支會互相影響的狀況考量到。

透過涵蓋度測試得到的結果是41%，許多的條件都沒有走到，透過IDE的涵蓋度測試工具逐漸改善成100%，這樣更能確保程式的錯誤機率降低，但仍然有些問題，如果把Swim.java中的折數的if順序打亂會導致折數不是取小的優先，涵蓋度測試並沒辦法找出這些錯誤，所以涵蓋度100%也只能表達出有測到這些地方而已，並不能保證有測到這個地方就沒錯。