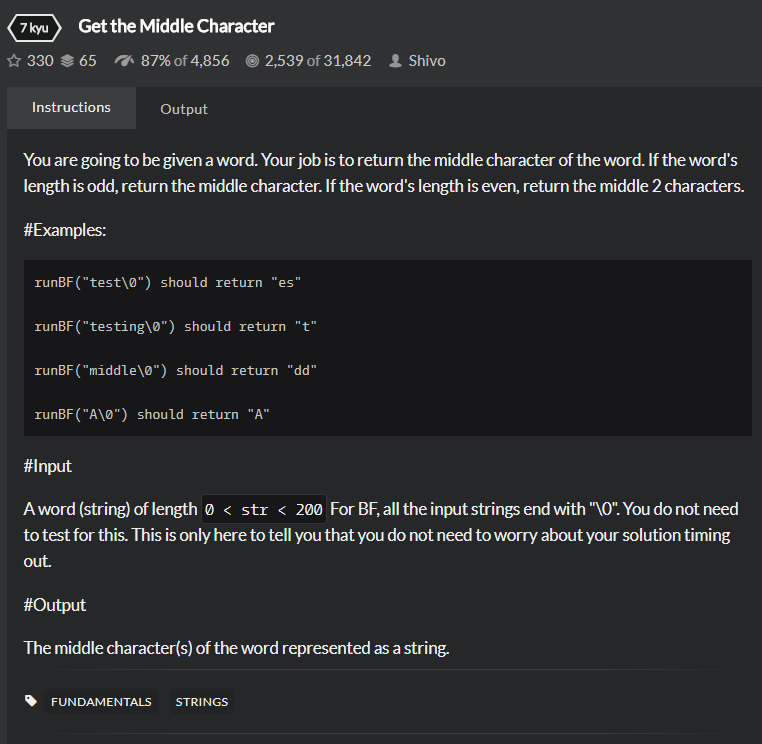
Day14. 嘗試切他中路!－Codewars\_Remove the minium

今天的題目長這樣



啊，抱歉，貼錯，我是說這個



今天的題目也Hen簡單，只需要輸入一組字串，找出中間的字是哪一個就可以惹。

題目表示:不需要考慮處理速度問題，所以字串長度只會介於0-200之間。

現在就來拆一下題目吧。

1. 輸入字串長度為奇數時，找出正中間的字元
2. 輸入字串長度為偶數時，找出正中間兩個字元

首先來吧! Test Code!!!

我們先從第一個需求切入，寫出第一個Test case!

```

[TestMethod]

public void Input\_stringEmpty\_Should\_Be\_stringEmpty()

{

Assert.AreEqual(string.Empty, Kata.GetMiddle(string.Empty));

}

```

而Production Code 也就是老樣子會長成這個樣子

```

public static string GetMiddle(string str)

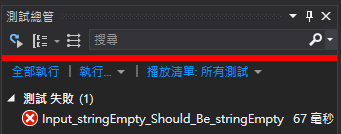
{

throw new System.**NotImplementedException**();

}

```

老樣子，跑個測試，沒過很正常，commit一下



接下來把Production Code改一下，用最簡單的方式解決他!

```

public static string GetMiddle(string str)

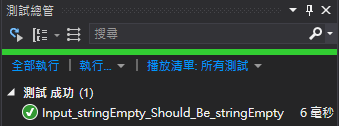
{

return string.Empty;

}

```

接下來跑個測試，PASS!



接下來寫個只輸入一個的測試吧!

```

[TestMethod]

public void Input\_A\_Should\_Be\_A()

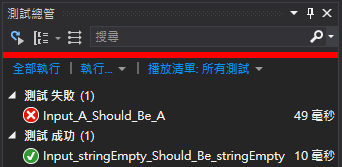
{

Assert.AreEqual("A",Kata.GetMiddle("A"));

}

```

接下來跑一下測試吧，紅燈! Commit!



接下來再把Production Code改一下

```

public static string GetMiddle(string str)

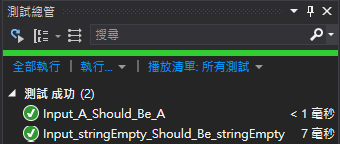
{

return str;

}

```

寫完就來跑個測試，PASS，Commit一下唄!



接下來想第3個測試，因為是先針對輸入奇數長度的字串，所以就先輸入長度為3的字串來當作第3個測試。

```

[TestMethod]

public void Input\_ABC\_Should\_Be\_B()

{

Assert.AreEqual("B",Kata.GetMiddle("ABC"));

}

```

寫好之後很正常的紅燈就來Commit一下吧!

再來就改一下Production Code唄

```

public static string GetMiddle(string str)

{

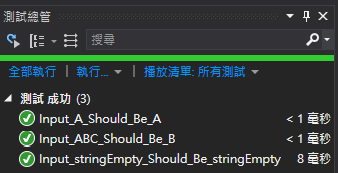
if (str.Length < 2) return str;

return str.Substring(str.Length / 2, 1);

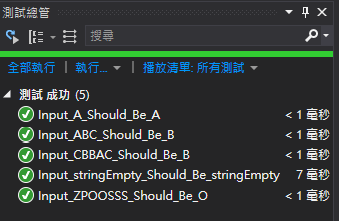
}

```

改完之後，跑個測試，Pass! Commit!



接下來補上幾個輸入為奇數長度的test case!



接下來就可以開始思考輸入為偶數長度的字串了，所以第一個偶數長度字串的測試案例是這個樣子。

```

[TestMethod]

public void Input\_AB\_Should\_Be\_AB()

{

Assert.AreEqual("AB",Kata.GetMiddle("AB"));

}

```

因為有奇數長度字串的前車之鑑，所以我們多一個判斷跟改一下subString的使用方法就可以通過輸入偶數長度字串的測試了。

```

public static string GetMiddle(string str)

{

if (str.Length < 2) return str;

if (str.Length % 2 == 1)

{

return str.Substring(str.Length / 2, 1);

}

else

{

return str.Substring(str.Length / 2 - 1, 2);

}

}

```

接下來補上幾個偶數字串長度的測試案例就可以開始Refactor我們的Code囉

```

public static string GetMiddle(string str)

{

if (str.Length < 2) return str;

return str.Length % 2 == 1 ?

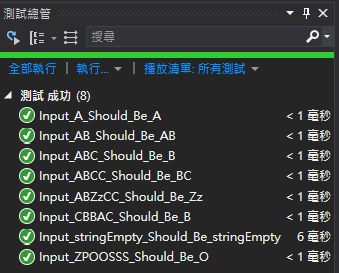
str.Substring(str.Length / 2, 1) :

str.Substring(str.Length / 2 - 1, 2);

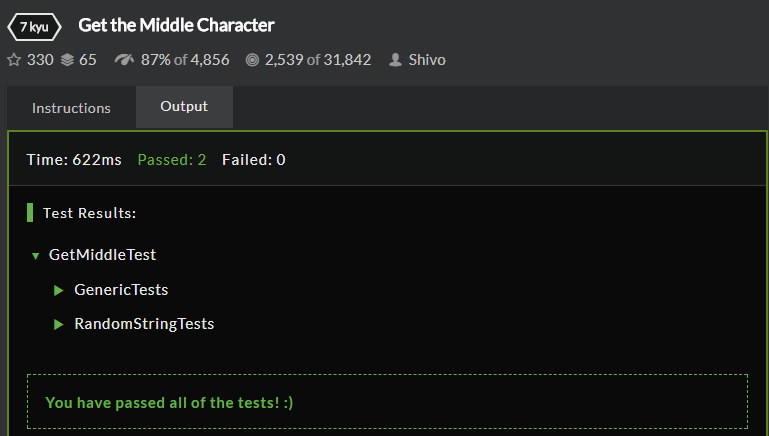
}

```

Refactor完畢之後再跑一下測試，pass，commit!

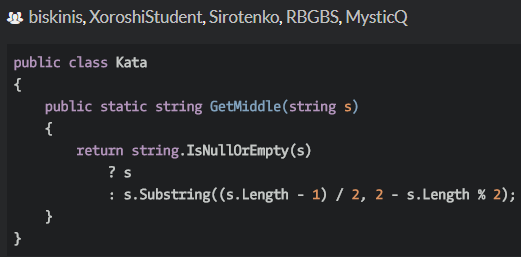


然後再Codewars上提交! Pass!!



通過! PASS!

老樣子我最喜歡的部分就是去看看別人寫這題寫得怎麼樣勒



看起來也跟我的差不多，不過他們的寫法比我的還要有進一步的思考，直接把長度考慮寫進Substring的方法中，真是高招OAO

Git url :

https://github.com/SQZ777/Codewars\_GetTheMiddleCharacter

下一題，明天見!