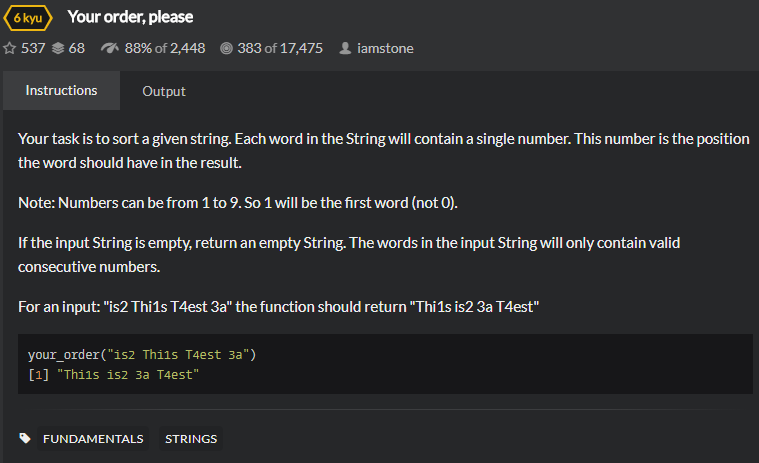
Day21. 你的餐點，謝謝(?) Codewars\_Your Order,Please

今天的題目難度是6kyu喔喔喔喔

廢話我就不多說惹，今天的題目長這樣



今天的題目要去拆解輸入的字串，而每一個被拆解的字串會含有一個數字，其數字只會在1~9之間。

現在就來拆一下題目吧。

1. 將字串拆解->split方法
2. 找出相對應數字字串
3. 將排序後的字串串起來->Join方法

一開始我們先來寫一個在字串陣列中找出相對應數字字串的方法吧!

那就是從輸入空的開始吧!!

```

[TestMethod]

public void Find\_Input\_1andEmpty\_Should\_Be\_Empty()

{

Assert.AreEqual(string.Empty, Kata.Find(string.Empty, 1));

}

```

而Production Code 也就是老樣子會長成這個樣子

```

public static string Find(string s, int n)

{

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

}  
```

老樣子，跑個測試，沒過很正常，紅燈，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 Find(string s, int n)

{

return string.Empty;

}  
```

接下來跑個測試，PASS! Commit~

突然發現是要傳入陣列才對，所以改好，跑過測試Pass再Commit一次。

接下來寫一個要找出含有數字1的陣列元素的測試。

```

[TestMethod]

public void Find\_Input\_1and1\_Should\_Be\_Empty()

{

Assert.AreEqual("1", Kata.Find(new string[] { "1" }, 1));

}

```

而Production Code就會長這個樣子

```

public static string Find(string[] strs, int n)

{

foreach (var s in strs)

{

if (s.Contains(n.ToString()))

{

return s;

}

}

return string.Empty;

}

```

重構之後Production Code可以變成這個樣子

```

public static string Find(string[] strs, int n)

{

return strs.FirstOrDefault(x => x.Contains(n.ToString())) ?? string.Empty;

}

```

??這個方法是之前自己有整理過關於?的用法的部落格，超讚的啦XD

可以趁機打廣告了

<https://dotblogs.com.tw/im_sqz777/2017/08/17/222734>

所以現在可以補上一些Find方法的測試案例

```

[TestMethod]

public void Find\_Input\_2andIm2Hi1DZ3()

[TestMethod]

public void Find\_Input\_1andIm2Hi1DZ3()

[TestMethod]

public void Find\_Input\_3andIm2Hi1DZ3()

```

都通過了之後就來寫符合第1跟第3的Production Code吧!

```

[TestMethod]

public void Input\_Im2Hi1DZ3\_Should\_Be\_Hi1Im2DZ3()

{

Assert.AreEqual("Hi1 Im2 DZ3", Kata.Order("Im2 Hi1 DZ3"));

}

```

而Production Code會長成這個樣子。

首先把str拆解。

然後迴圈輪流去尋找相對應的數字如果沒有就回傳Empty

最後再用Join把陣列中的值串接起來就大功告成啦!

```

public static string Order(string str)

{

var splitted = str.Split();

var result = new List<string>();

for (int i = 1; i <= splitted.Length; i++)

{

result.Add(Find(splitted, i) ?? string.Empty);

}

return string.Join(" ", result);

}

```

跑個測試，Pass!

接下來就可以提交Codewars啦!

以下是今天所有的測試案例

```

[TestClass]

public class UnitTest1

{

[TestMethod]

public void Find\_Input\_1andEmpty\_Should\_Be\_Empty()

{

Assert.AreEqual(string.Empty, Kata.Find(new string[] { string.Empty }, 1));

}

[TestMethod]

public void Find\_Input\_1and1\_Should\_Be\_Empty()

{

Assert.AreEqual("1", Kata.Find(new string[] { "1" }, 1));

}

[TestMethod]

public void Find\_Input\_2andIm2Hi1DZ3()

{

Assert.AreEqual("Im2", Kata.Find(new string[] { "Im2", "Hi1", "DZ3" }, 2));

}

[TestMethod]

public void Find\_Input\_1andIm2Hi1DZ3()

{

Assert.AreEqual("Hi1", Kata.Find(new string[] { "Im2", "Hi1", "DZ3" }, 1));

}

[TestMethod]

public void Find\_Input\_3andIm2Hi1DZ3()

{

Assert.AreEqual("DZ3", Kata.Find(new string[] { "Im2", "Hi1", "DZ3" }, 3));

}

[TestMethod]

public void Input\_Im2Hi1DZ3\_Should\_Be\_Hi1Im2DZ3()

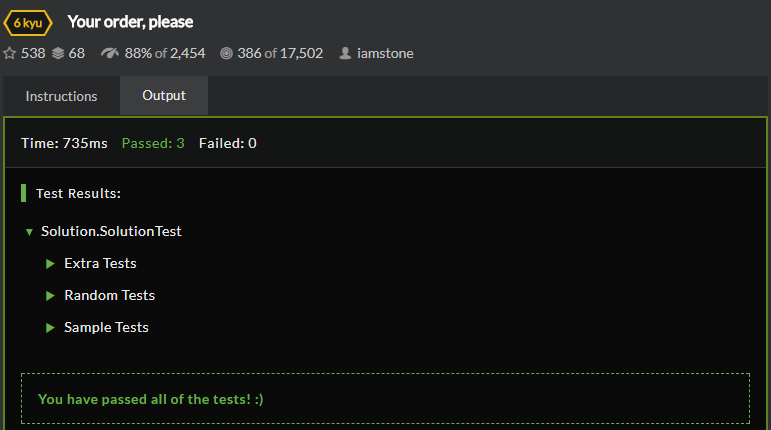
{

Assert.AreEqual("Hi1 Im2 DZ3", Kata.Order("Im2 Hi1 DZ3"));

}

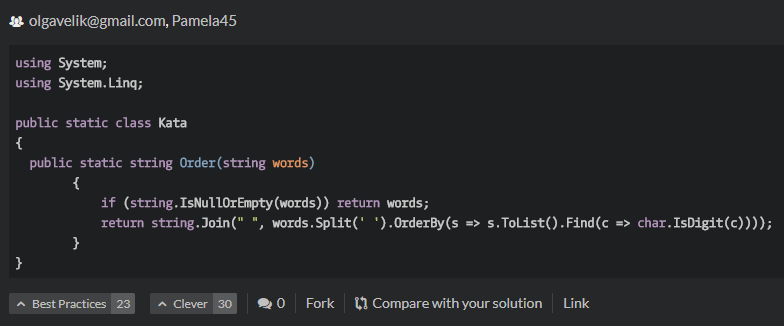
}

```



在Codewars上成功提交了~

來看一下其他人怎麼寫吧!



原來有char.IsDigit這種東西啊

看起來應該是這個char是數字應該就會存起來了。

看起來Hen猛啊 O\_O

Git url :

<https://github.com/SQZ777/Codewars_YourOrderPlease>

Codewars Link:

<https://www.codewars.com/kata/your-order-please/train/csharp>

下一題，明天見!