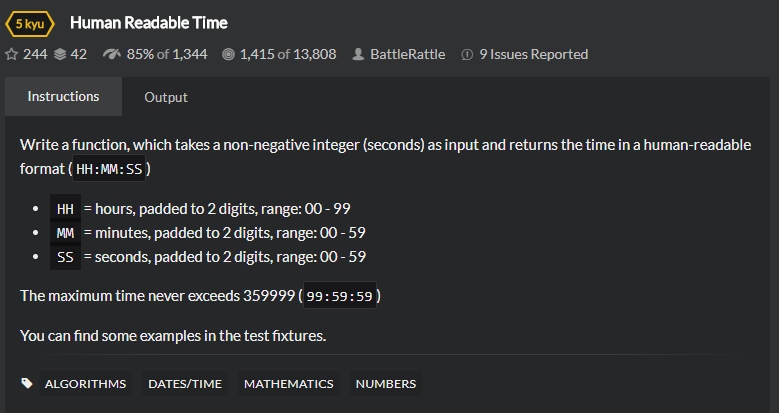
Day23 這時間你看得懂嗎? Codewars\_Human Readable Time

今天的題目難度是5kyu喔喔喔

今天的題目長這樣



今天的題目需求需要將秒數換算成小時分鐘跟秒數。

現在就來拆一下題目吧。

1. 換算小時的方法。
2. 換算分鐘的方法。
3. 換算剩餘秒數的方法。

一開始我們先來寫一個換算小時方法的測試資料吧!!

```

[TestMethod]

public void GetHour\_Input\_0\_Should\_Be\_0()

{

Assert.AreEqual(0, TimeFormat.GetHour(0));

}

```

而GetHour方法的Production Code 也就是老樣子會長成這個樣子

```

public static int GetHour(int num)

{

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 int GetHour(int num)

{

return 0;

}  
```

接下來跑個測試，PASS! Commit~

再來加上3600秒的測試。

```

[TestMethod]

public void GetHour\_Input\_3600\_Should\_Be\_1()

{

Assert.AreEqual(1, TimeFormat.GetHour(3600));

}

```

而GetHour方法的code就會長這個樣子

```

public static int GetHour(int num)

{

return num / 3600;

}

```

接下來要寫計算分的方法的測試。

```

[TestMethod]

public void GetMin\_Input\_0\_Should\_Be\_0()

{

Assert.AreEqual(0,TimeFormat.GetMin(0));

}

```

這時候Production Code會變成

```

public static int GetMin(int num)

{

return 0;

}

```

再來寫輸入3600的測試

```

[TestMethod]

public void GetMin\_Input\_3600\_Should\_Be\_0()

{

Assert.AreEqual(0, TimeFormat.GetMin(3600));

}

```

這時候Production Code會變成這個概念

```

public static int GetMin(int num)

{

return num / 60;

}

```

再來想到要寫正常的數字，所以就寫了輸入100秒的測試。

```

[TestMethod]

public void GetMin\_Input\_100\_Should\_Be\_1()

{

Assert.AreEqual(1, TimeFormat.GetMin(100));

}

```

GetMin方法的Code最後會長成這個樣子。

```

public static int GetMin(int num)

{

return num / 60 % 60;

}

```

最後要開始寫GetSec的方法測試啦。

第一個一樣是輸入0。

```

[TestMethod]

public void GetSec\_Input\_0\_Should\_Be\_0()

{

Assert.AreEqual(0, TimeFormat.GetSec(0));

}

```

GetSec的Code就會長這個樣子。

```

public static int GetSec(int num)

{

return 0;

}

```

再來寫一個輸入100的測試。

```

[TestMethod]

public void GetSec\_Input\_100\_Should\_Be\_40()

{

Assert.AreEqual(40, TimeFormat.GetSec(100));

}

```

GetSec方法的Code就會變成這個樣子

```

public static int GetSec(int num)

{

return num % 60;

}

```

其實這樣就足夠涵蓋範圍了。

不過我還是補了一個輸入3600的測試案例這我就不多貼了。

接下來就要開始寫主要Production Code的測試啦!

```

[TestMethod]

public void Input\_0\_Should\_Be\_00\_00\_00()

{

Assert.AreEqual("00:00:00", TimeFormat.GetReadableTime(0));

}

```

由於要加入中間的冒號跟判斷各個取得的時間是否小於零所以要加上幾個if並相加起來處理，所以ProductionCode 就變成這個樣子。

```

public static string GetReadableTime(int num)

{

var hour = GetHour(num).ToString();

var min = GetMin(num).ToString();

var sec = GetSec(num).ToString();

if (Convert.ToInt32(hour) < 10)

hour = "0" + hour;

if (Convert.ToInt32(min) < 10)

min = "0" + min;

if (GetSec(num) < 10)

sec = "0" + sec;

return hour + ":" + min + ":" + sec;

}

```

接下來補上一些test case，all pass之後Commit!

以下是所有的Production Code

```

public class TimeFormat

{

public static int GetHour(int num)

{

return num / 3600;

}

public static int GetMin(int num)

{

return num / 60 % 60;

}

public static int GetSec(int num)

{

return num % 60;

}

public static string GetReadableTime(int num)

{

var hour = GetHour(num).ToString();

var min = GetMin(num).ToString();

var sec = GetSec(num).ToString();

if (Convert.ToInt32(hour) < 10)

hour = "0" + hour;

if (Convert.ToInt32(min) < 10)

min = "0" + min;

if (GetSec(num) < 10)

sec = "0" + sec;

return hour + ":" + min + ":" + sec;

}

}

```

以下是今天所有的測試案例

```

[TestClass]

public class UnitTest1

{

[TestMethod]

public void GetHour\_Input\_0\_Should\_Be\_0()

{

Assert.AreEqual(0, TimeFormat.GetHour(0));

}

[TestMethod]

public void GetHour\_Input\_3600\_Should\_Be\_1()

{

Assert.AreEqual(1, TimeFormat.GetHour(3600));

}

[TestMethod]

public void GetMin\_Input\_0\_Should\_Be\_0()

{

Assert.AreEqual(0, TimeFormat.GetMin(0));

}

[TestMethod]

public void GetMin\_Input\_3600\_Should\_Be\_0()

{

Assert.AreEqual(0, TimeFormat.GetMin(3600));

}

[TestMethod]

public void GetMin\_Input\_100\_Should\_Be\_1()

{

Assert.AreEqual(1, TimeFormat.GetMin(100));

}

[TestMethod]

public void GetSec\_Input\_0\_Should\_Be\_0()

{

Assert.AreEqual(0, TimeFormat.GetSec(0));

}

[TestMethod]

public void GetSec\_Input\_100\_Should\_Be\_40()

{

Assert.AreEqual(40, TimeFormat.GetSec(100));

}

[TestMethod]

public void GetSec\_Input\_3600\_Should\_Be\_0()

{

Assert.AreEqual(0, TimeFormat.GetSec(3600));

}

[TestMethod]

public void Input\_0\_Should\_Be\_00\_00\_00()

{

Assert.AreEqual("00:00:00", TimeFormat.GetReadableTime(0));

}

[TestMethod]

public void Input\_5\_Should\_Be\_00\_00\_05()

{

Assert.AreEqual("00:00:05", TimeFormat.GetReadableTime(5));

}

[TestMethod]

public void Input\_60\_Should\_Be\_00\_01\_00()

{

Assert.AreEqual("00:01:00", TimeFormat.GetReadableTime(60));

}

[TestMethod]

public void Input\_86399\_Should\_Be\_23\_59\_59()

{

Assert.AreEqual("23:59:59", TimeFormat.GetReadableTime(86399));

}

[TestMethod]

public void Input\_359999\_Should\_Be\_99\_99\_99()

{

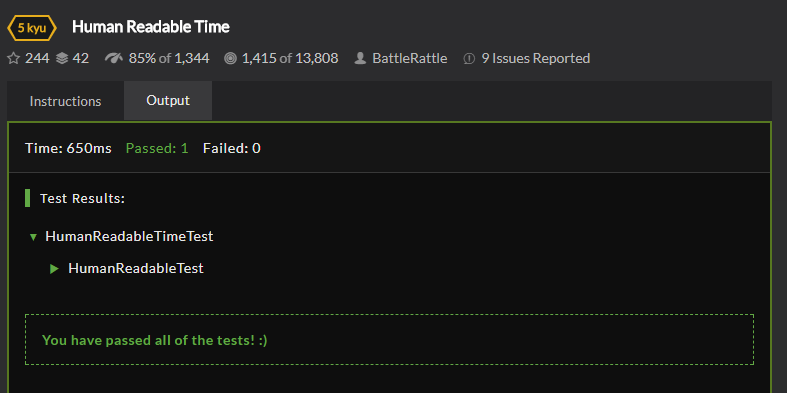
Assert.AreEqual("99:59:59", TimeFormat.GetReadableTime(359999));

}

}

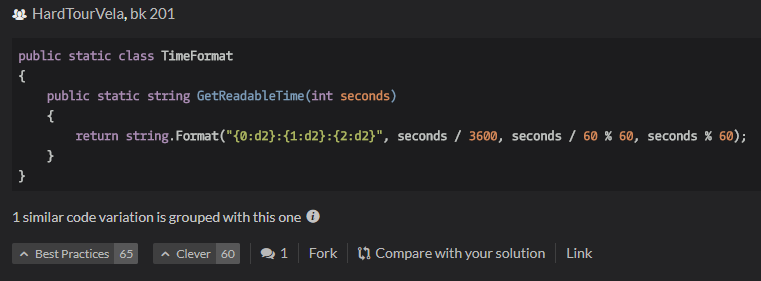
```

現在就可以很愜意的上傳到Codewars然後就Pass啦!



在Codewars上成功提交了~

來看一下其他人怎麼寫吧!



看起來我的寫法跟別人有點落差，別人用一行就解決了XD

不過我們的想法是一樣的。

只是我把東西都把它拆解開來了~

Git url :

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

Codewars Link:

<https://www.codewars.com/kata/human-readable-time/train/csharp>

下一題，明天見!