

บทที่ 1 แนะนำ ASP.NET Web Pages - บทนำ

เอกสารนี้จะแนะนำให้คุณรู้จักกับ ASP.NET Web Pages เบื้องต้นเพื่อนำความรู้ที่ได้ไปใช้ในการพัฒนา Web Application และฝึกฝีมือหาความรู้เพิ่มเติมขั้นสูงในลำดับต่อไป

ระดับความเชี่ยวชาญ: มือใหม่สำหรับ ASP.NET

ความรู้พื้นฐานที่ต้องมี: HTML, basic cascading style sheets (CSS)

เนื้อหาที่จะได้เรียนรู้ในบทนี้

- ASP.NET Web Pages คืออะไรและทำอะไรได้บ้าง
- แนะนำ Visual Studio 2013 สำหรับการพัฒนา ASP.NET Web Pages
- การติดตั้งซอฟต์แวร์ต่างๆที่จำเป็น
- การสร้าง Web Application ด้วย Visual Studio 2013

เทคโนโลยีที่ใช้ในบทนี้

- Visual Studio 2013 Community Edition
- ไฟล์แบบ .cshtml

ซอฟต์แวร์ที่ใช้ในบทนี้

- ASP.NET Web Pages (Razor) 3.0
- Visual Studio 2013 Community Edition

ความรู้พื้นฐานที่คุณต้องมี

เรามั่นใจว่าคุณต้องรู้จักเทคโนโลยีต่อไปนี้

- HTML – ไม่จำเป็นต้องรู้ลึกมาก และเราจะไม่อธิบายลงรายละเอียดให้ลึกลงไป และในเอกสารนี้ก็จะใช้แต่คำสั่ง HTML เบื้องต้นเท่านั้น
- Cascading style sheets (CSS) – เราอยากอธิบายเหมือนกับ HTML
- ความรู้พื้นฐานเรื่องระบบฐานข้อมูล – หากเคยใช้โปรแกรม Excel เพื่อกรอกข้อมูล คำนวณข้อมูล และค้นหาข้อมูล ก็ถือว่ามีความรู้พอสำหรับการพัฒนาตามเอกสารนี้แล้ว

เรามั่นใจว่าคุณมีความสนใจในการเขียนโปรแกรม ASP.NET Web Pages ขึ้นเริ่มต้นด้วยภาษาโปรแกรมที่เรียกว่า C# (อ่านว่าซีชาร์ป) คุณไม่จำเป็นต้องมีพื้นฐานของภาษานี้มาก่อน ขอแค่คุณสนใจ หากเคยเขียนภาษาอื่นๆมาก่อนเช่น JavaScript นั้นก็ถือว่ายอดเยี่ยมเลย

คำเตือน – หากคุณมีประสบการณ์การเขียนโปรแกรมมากมาย คุณจะรู้สึกว่าการอ่านเอกสารนี้ช้า ไม่ทันใจคุณ เพราะเราต้องการให้เป็นเช่นนั้น เพื่อโปรแกรมเมอร์มือใหม่สามารถเริ่มต้นได้รวดเร็ว คุณอาจจะข้ามบทนี้ และบทอื่นๆสองสามบทแรกไปหากคุณต้องการ

คุณต้องมีอะไรบ้าง

- เครื่องคอมพิวเตอร์ที่ทำงานโดยใช้ระบบปฏิบัติการ Windows 8, Windows 7, Windows Server 2008 หรือ Windows Server 2012
- มีการเชื่อมต่ออินเทอร์เน็ต
- คุณมีสิทธิ์ระดับผู้ดูแลระบบเพื่อให้สามารถติดตั้งซอฟต์แวร์ต่างๆได้

ASP.NET Web Pages คืออะไร?

ASP.NET Web Pages คือชุดซอฟต์แวร์และเครื่องมือต่างๆเพื่อช่วยในการสร้าง Web Application ที่สามารถปรับเปลี่ยนได้ตามการเขียนคำสั่งและเงื่อนไขตามต้องการ ซึ่งแตกต่างจากการสร้าง Web โดยใช้เพียง HTML อย่างเดียวที่จะได้ Web แบบปรับเปลี่ยนไม่ได้คงที่ถาวร

Web ที่ปรับเปลี่ยนได้ด้วยการเขียนคำสั่งช่วยให้คุณสร้าง Web Application ที่มีการทำงานได้ทุกอย่างตามที่ต้องการ คุณสามารถสร้างแบบฟอร์มเพื่อถามข้อมูลต่างๆและเปลี่ยนแปลงการแสดงผลของแบบฟอร์มด้วยการเขียนคำสั่ง สามารถนำไปจัดเก็บลงระบบฐานข้อมูลและนำมาแสดงเป็นรายการข้อมูลในภายหลัง คุณสามารถส่งอีเมลจาก Website ของคุณ คุณสามารถติดต่อกับบริการต่างๆบน Web เช่น บริการแผนที่ และสร้างเว็บไซต์ที่แสดงข้อมูลจากบริการต่างๆเหล่านั้น

Visual Studio 2013 Community Edition คืออะไร?

Visual Studio 2013 คือซอฟต์แวร์ที่รวมเครื่องมือทั้งหมดที่จำเป็นสำหรับการพัฒนาซอฟต์แวร์ต่างๆรวมทั้ง Web Application มีทั้งเครื่องมือการเขียนคำสั่ง การทดสอบ การจัดการระบบฐานข้อมูล และเครื่องมือในการติดตั้ง Web Application บนระบบเครือข่าย

Visual Studio 2013 Community Edition คือรุ่นหนึ่งของ Visual Studio 2013 ซึ่งเป็นรุ่นที่สามารถใช้งานได้โดยไม่เสียค่าใช้จ่ายโดยมีเงื่อนไขดังนี้

1. ใช้ในการเรียนการสอน การทำผลงานวิจัย และโครงการซอฟต์แวร์แบบ Open Source
2. หากนอกเหนือจากข้อ 1 คุณก็สามารถใช้งานได้โดยมีข้อจำกัดดังนี้
 - ต้องไม่ใช้งานในหน่วยงานเอกชนขนาดใหญ่ ที่มีจำนวนเครื่องคอมพิวเตอร์เกิน 250 เครื่อง หรือมีผลประกอบการเกิน 1 US Dollar
 - มีจำนวนโปรแกรมเมอร์ไม่เกิน 5 คน

เกี่ยวกับเอกสารการสอนฉบับนี้

เอกสารการสอนนี้จะแนะนำวิธีการใช้งาน ASP.NET Web Pages โดยมีทั้งหมด 9 บท ซึ่งเป็นส่วนหนึ่งของบทเรียนสำหรับผู้เริ่มต้นเรียนรู้ ASP.NET Web Pages เพื่อให้สามารถสร้าง Website ที่ดูเป็นมืออาชีพได้โดยง่าย

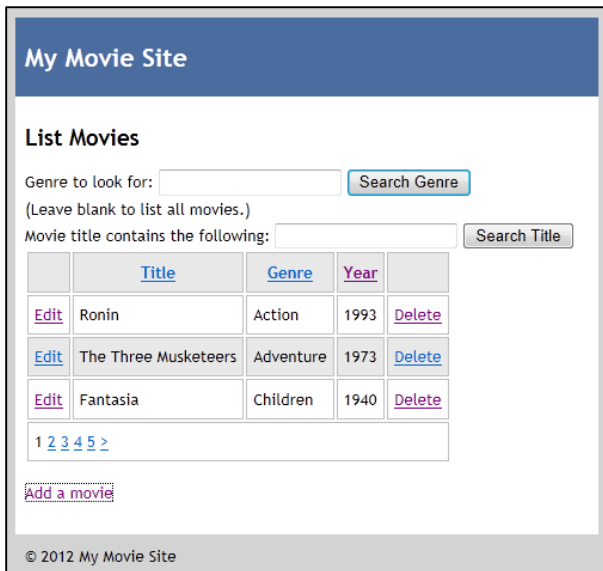
เนื้อหาทั้ง 9 บทประกอบไปด้วย

- แนะนำภาพรวมทุกอย่างและการติดตั้งซอฟต์แวร์ (คือบทที่คุณกำลังอ่านอยู่)
- แนะนำพื้นฐานการเขียนโปรแกรม ASP.NET Web Pages
- การสร้างฐานข้อมูล
- การสร้างและประมวลผลข้อมูลจากแบบฟอร์ม
- การเพิ่ม แก้ไขและลบข้อมูลจากฐานข้อมูล

สิ่งที่คุณจะสร้าง

เอกสารการสอนนี้จะเป็นการเรียนรู้ที่จะสร้าง Website ที่คุณสามารถแสดงรายการภาพยนตร์ที่คุณชื่นชอบ คุณสามารถกรอกข้อมูลภาพยนตร์ แก้ไขและแสดงรายการภาพยนตร์ โดยมีภาพตัวอย่าง Website ที่สร้างแล้วเสร็จดังนี้

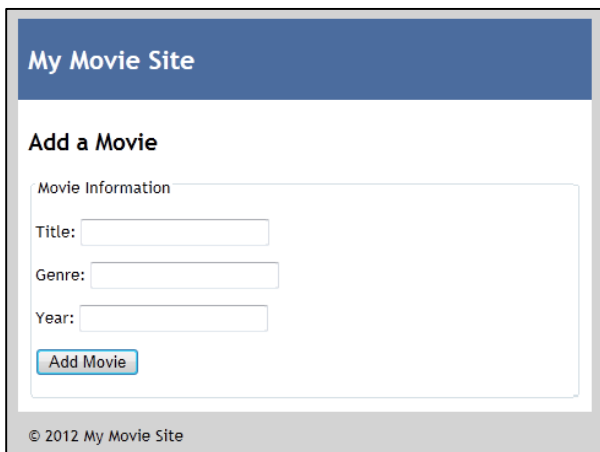
ภาพหน้าสำหรับแสดงและค้นหารายการภาพยนตร์



The screenshot shows a web page titled "My Movie Site" with a "List Movies" section. It features search filters for "Genre" and "Movie title", a table of movies, and a footer with the copyright notice "© 2012 My Movie Site".

	Title	Genre	Year	
Edit	Ronin	Action	1993	Delete
Edit	The Three Musketeers	Adventure	1973	Delete
Edit	Fantasia	Children	1940	Delete

ภาพหน้าแบบฟอร์มกรอกข้อมูลภาพยนตร์ที่ต้องการเพิ่ม

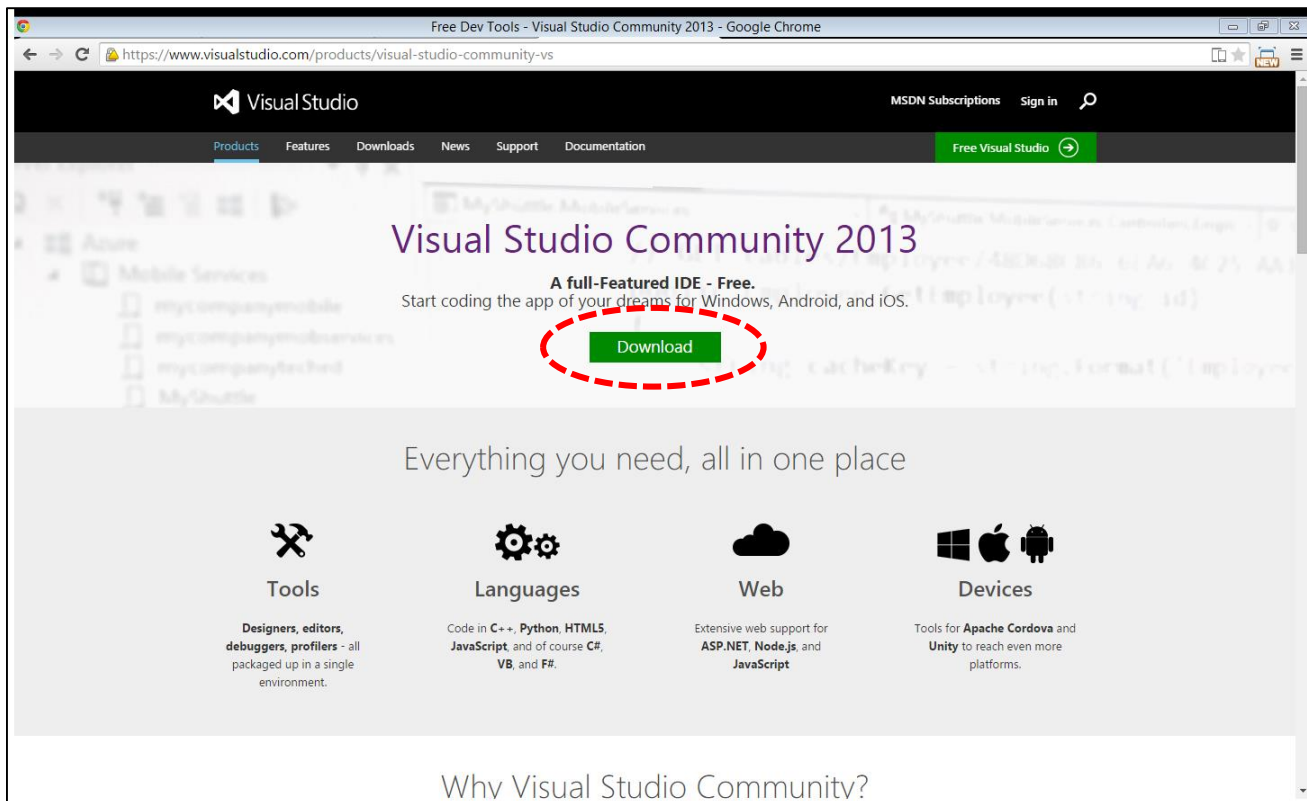


The screenshot shows a web page titled "My Movie Site" with an "Add a Movie" section. It contains a form with input fields for "Title", "Genre", and "Year", and an "Add Movie" button. The footer includes the copyright notice "© 2012 My Movie Site".

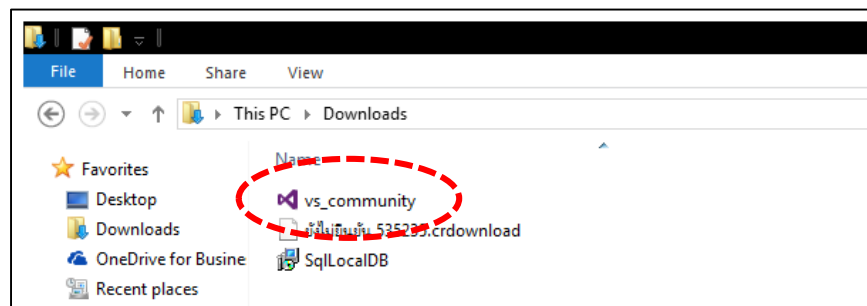
เริ่มการติดตั้งซอฟต์แวร์ที่ต้องใช้

ซอฟต์แวร์ตัวแรกที่คุณต้องทำการติดตั้งคือ Visual Studio 2013 Community Edition โดยคุณจะต้องทำการ Download จาก Website ของ Microsoft และทำการติดตั้งลงในเครื่องของคุณ และคุณต้องมี Microsoft Account ด้วย (Hotmail หรือ Outlook.com) เพื่อใช้ในการ Activate ซอฟต์แวร์จากทดลองใช้ เป็น Community Edition สำหรับขั้นตอนการติดตั้งมีดังนี้

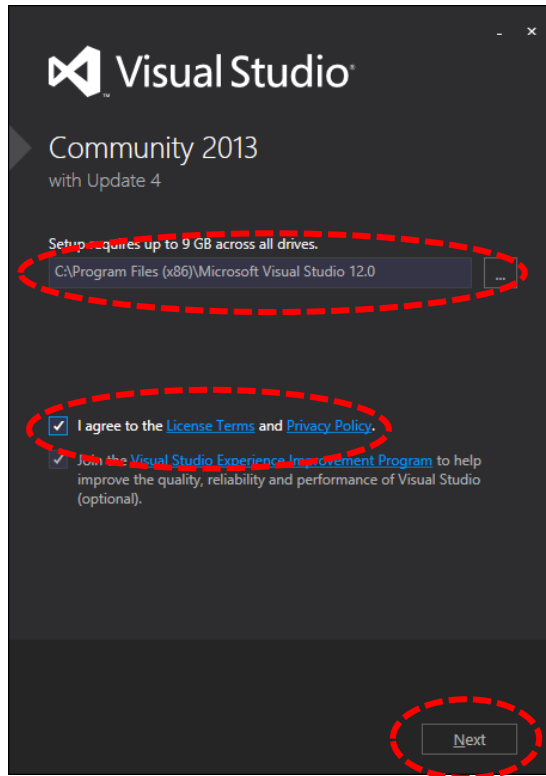
ทำการเปิดโปรแกรม Google Chrome แล้วไปที่ <https://www.visualstudio.com/products/visual-studio-community-vs> แล้วคลิกที่ปุ่ม Download



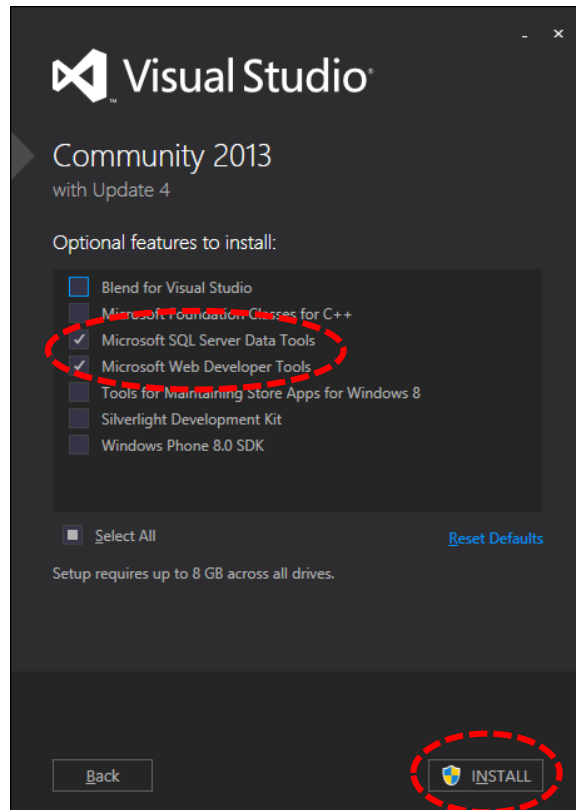
หลังจาก Download เสร็จจะได้ไฟล์ชื่อ vs_community.exe ในแฟ้ม Download ทำการ Double click เพื่อติดตั้ง



เลือกแฟ้มที่ต้องการติดตั้ง แล้วเลือกหัวข้อ I agree to the License Terms and Privacy Policy. แล้วคลิกปุ่ม Next

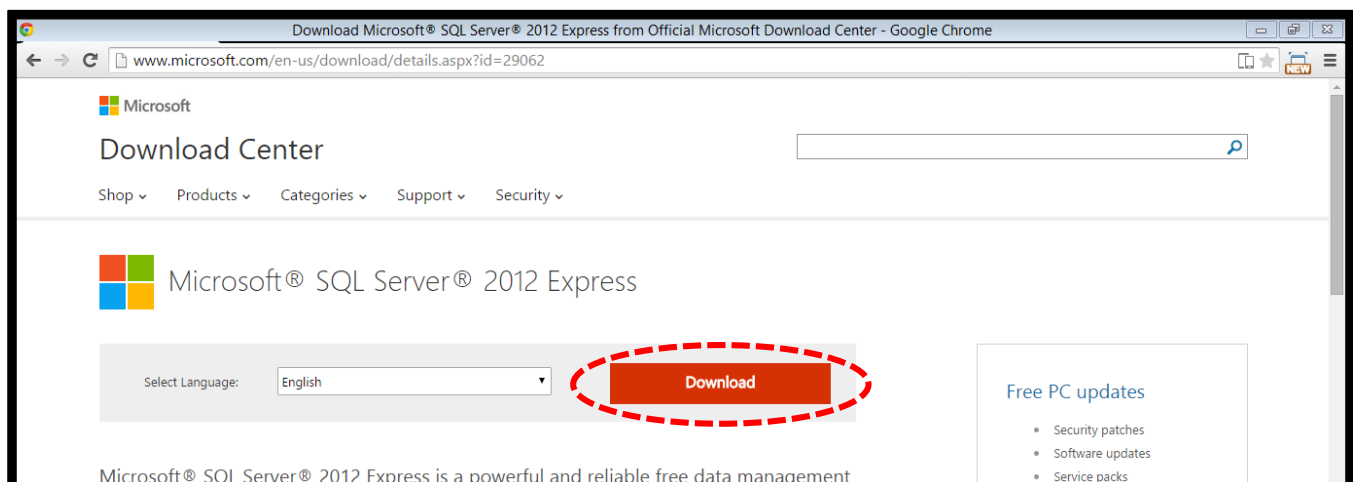


ทำการเลือกรายการซอฟต์แวร์และส่วนประกอบที่จะติดตั้งโดยสิ่งที่จำเป็นต้องเลือกสำหรับการเขียนโปรแกรมของเราก็คือ Microsoft SQL Server Data Tools และ Microsoft Web Developer Tools โดยคุณสามารถเลือกรายการอื่นๆได้ตามที่ต้องการ หลังจากเลือกเสร็จแล้วให้คลิกที่ปุ่ม INSTALL เพื่อเริ่มการติดตั้ง

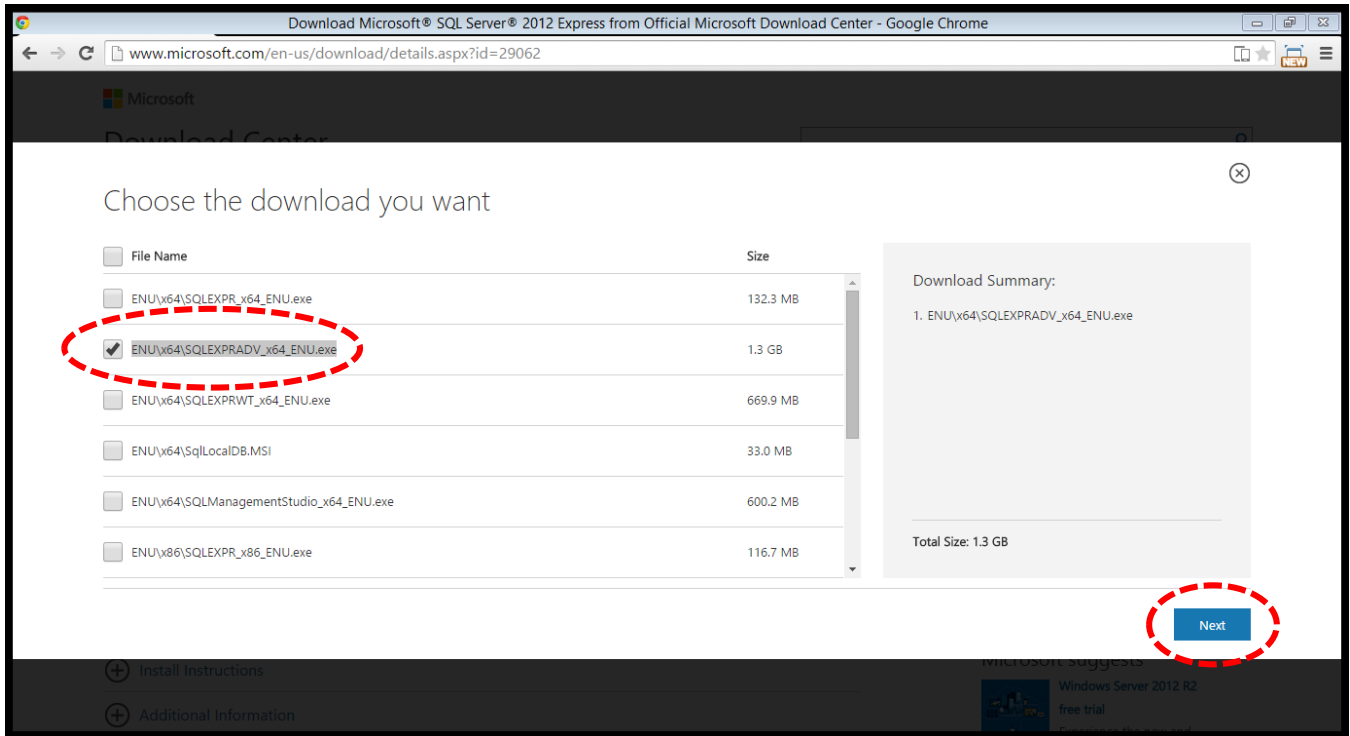


ขั้นตอนถัดไปเป็นการติดตั้งระบบจัดการฐานข้อมูลโดยเราสามารถใช้ระบบฐานข้อมูลได้มากมายหลายยี่ห้อในการทำ Web Application ด้วย ASP.NET Web Pages โดยเราจะใช้ SQL Server 2012 ซึ่งเป็นระบบฐานข้อมูลที่ผนวกเข้ากับ ASP.NET Web Pages ได้ดีที่สุดในขณะนี้ โดย SQL Server 2012 มีทั้งในแบบมีค่าใช้จ่ายและแบบใช้งานได้ฟรี โดยเราจะเลือกใช้ SQL Server 2012 Express Edition ซึ่งเป็นรุ่นสำหรับใช้งานได้ฟรี

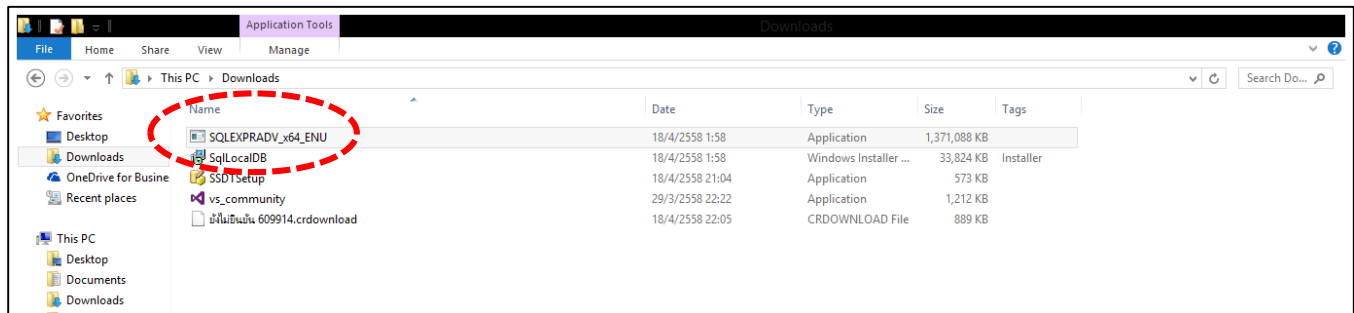
เริ่มทำการติดตั้ง SQL Server 2012 Express โดยการไปที่ <http://www.microsoft.com/en-us/download/details.aspx?id=29062> แล้วคลิกที่ปุ่ม Download



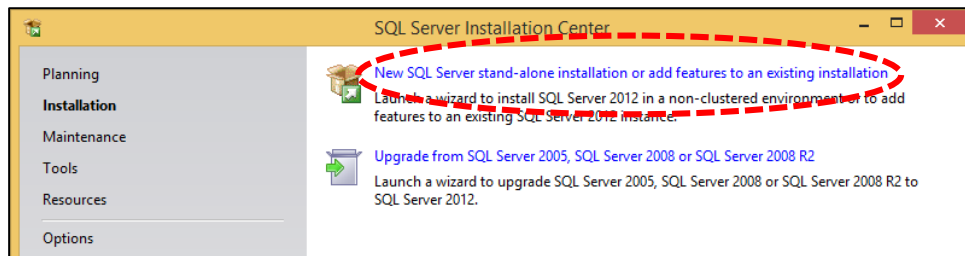
คลิกเลือกที่รายการ ENU\x64\SQLEXPADV_x64_ENU.exe แล้วคลิกปุ่ม Next เพื่อเริ่ม Download



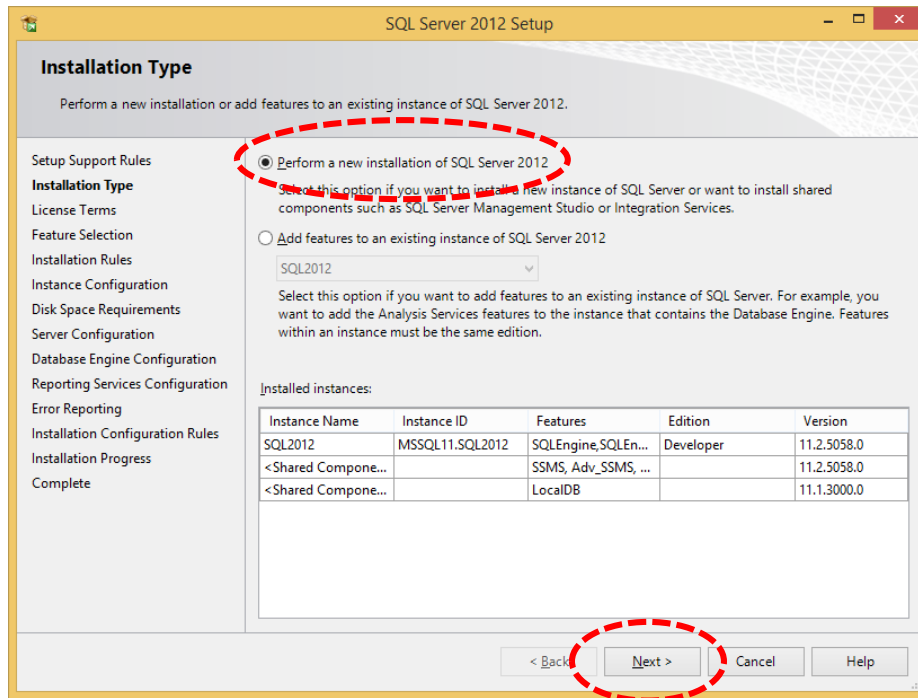
เมื่อการ Download เสร็จสิ้นจะได้ไฟล์ SQLSERVERPRADV_x64_ENU.exe ในแฟ้ม Downloads ให้ทำการ Double click เพื่อเริ่มการติดตั้ง



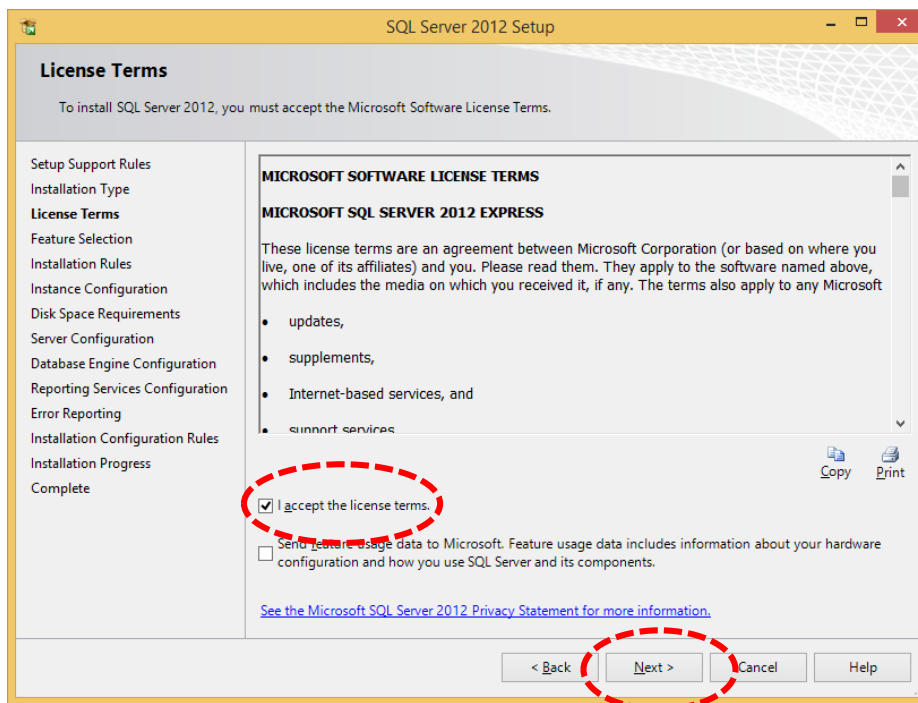
คลิกที่ New SQL Server stand-alone installation or add features to an existing installation



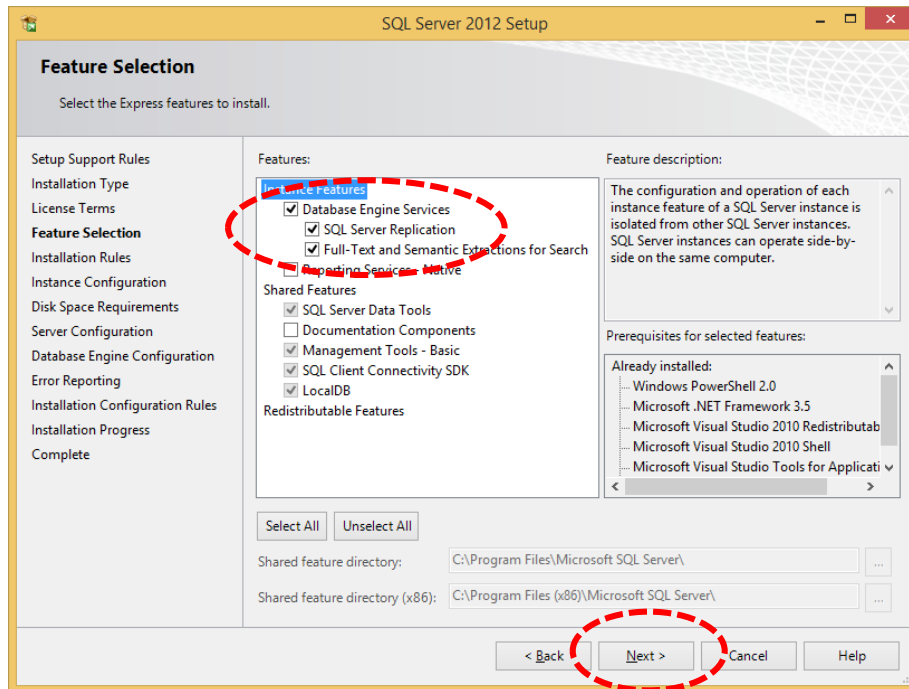
คลิกที่ Next แล้วเลือก Perform a new installation of SQL Server 2012



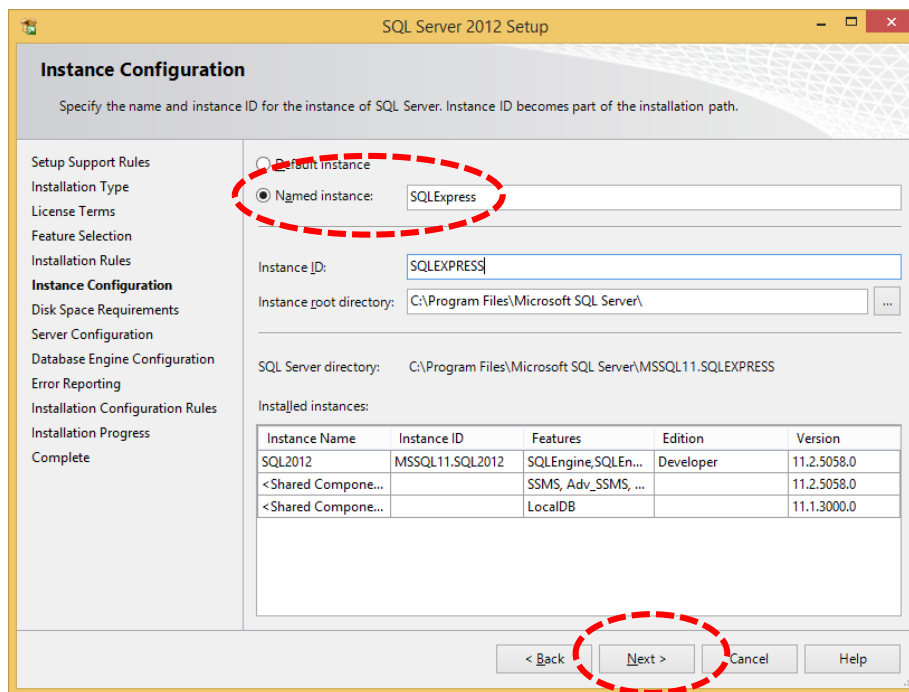
คลิกเลือกที่ I accept the license term แล้วคลิกปุ่ม Next



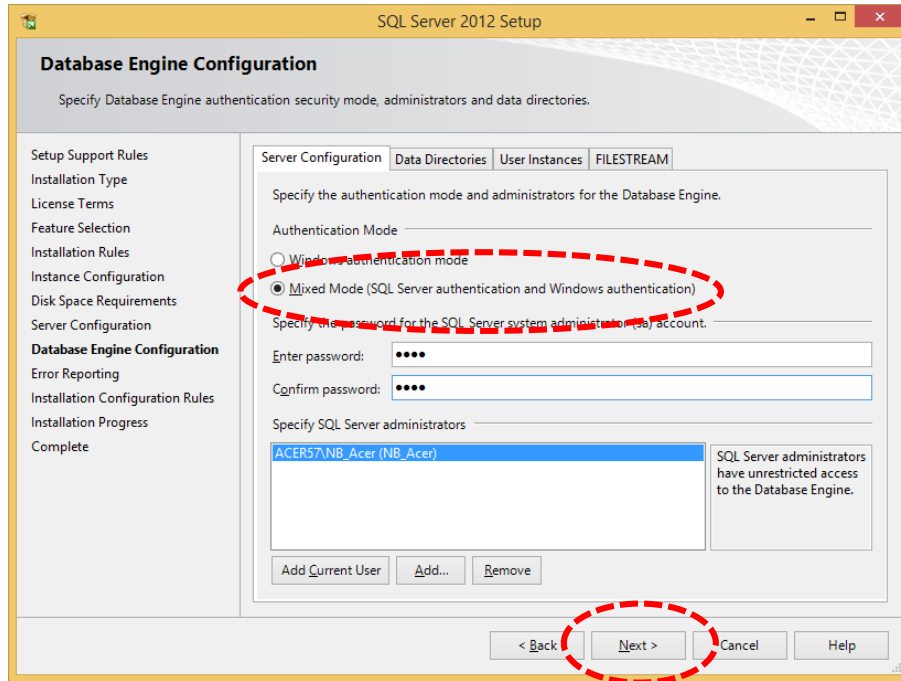
ในหน้าจอถัดไปเป็นการเลือกส่วนประกอบที่จะติดตั้ง โดยต้องเลือกรายการ Database Engine Services เป็นอย่างน้อยแล้วคลิกที่ปุ่ม Next



ในหน้าจอถัดไปให้เลือกที่ Named instance โดยใช้ชื่อ Instance ว่า SQLEXPRESS แล้วคลิกที่ปุ่ม Next



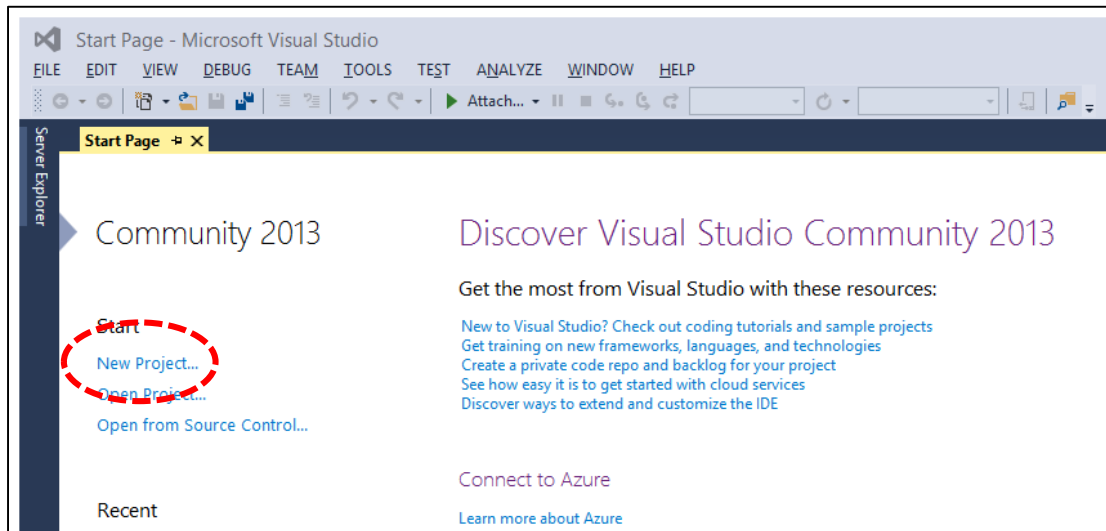
คลิกที่ปุ่ม Next ที่หน้าจอ Database Engine Configuration เลือก Authentication Mode เป็น Mixed Mode แล้วรอรหัสผ่านสำหรับ sa (System administrator) แล้วคลิกปุ่ม Next



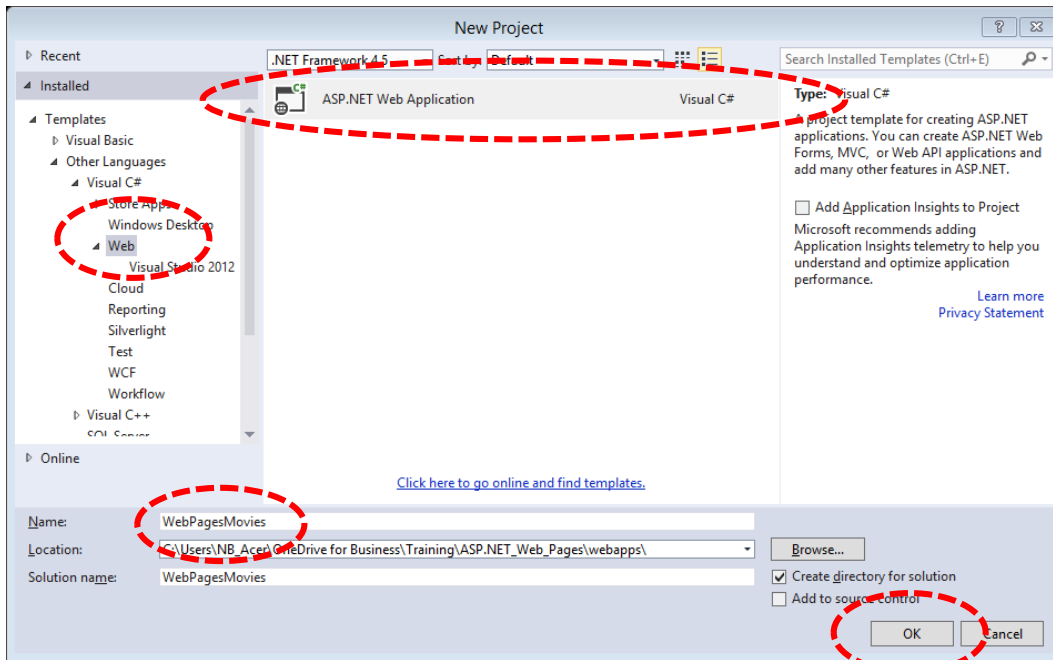
คลิกปุ่ม Next เพื่อเริ่มการติดตั้งจนแล้วเสร็จ

เริ่มใช้งาน Visual Studio 2013

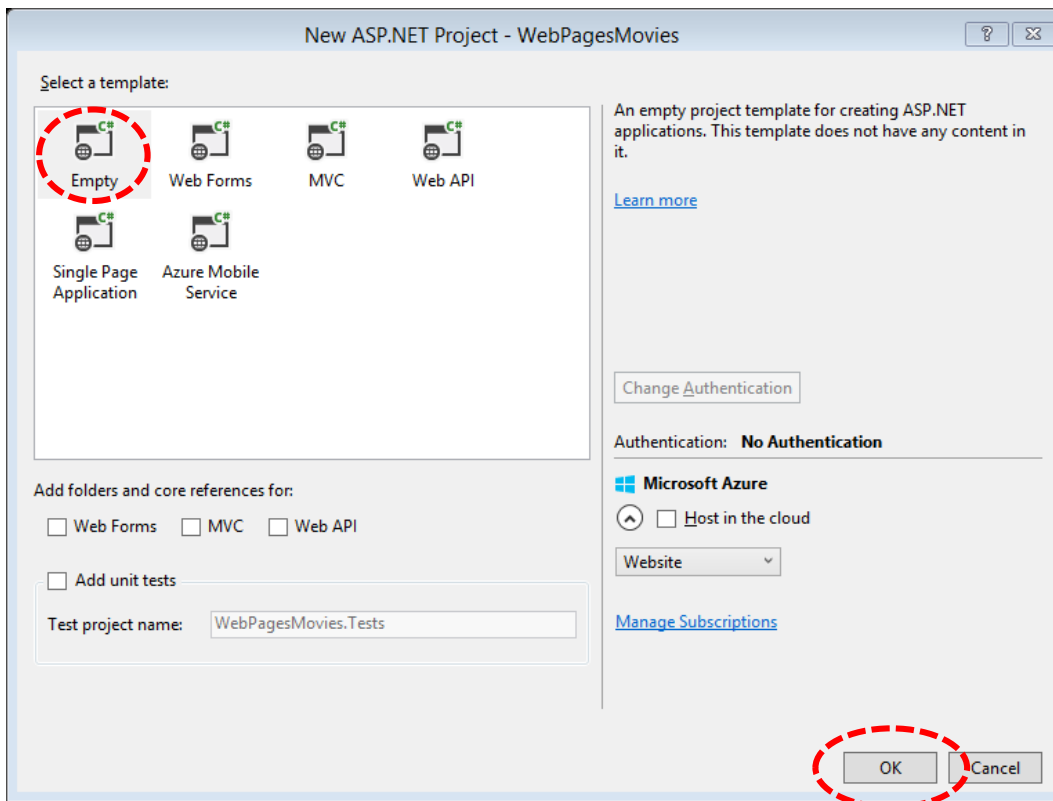
หลังจากติดตั้งซอฟต์แวร์ต่างๆครบถ้วนแล้วให้ทำการเปิดโปรแกรม Visual Studio 2013 โดยในการเปิดครั้งแรกให้ทำการ Login โดยใช้ Microsoft Account เพื่อ Activate ซอฟต์แวร์เพื่อใช้งานจริง หลังจากนั้นให้ทำการสร้างโครงการใหม่เพื่อเริ่มการสร้าง Web Application โดยการคลิกที่ New Project ในหน้าจอ Start Page



ที่หน้าต่าง New Project ในหัวข้อ Template ให้เลือก Visual C# > Web > ASP.NET Web Application ในช่อง Name ให้ตั้งชื่อ Project เป็น WebPagesMovies ในช่อง Location ให้เลือกแฟ้มที่จะจัดเก็บไฟล์โครงการ หลังจากนั้นคลิกปุ่ม OK เพื่อสร้างโครงการใหม่

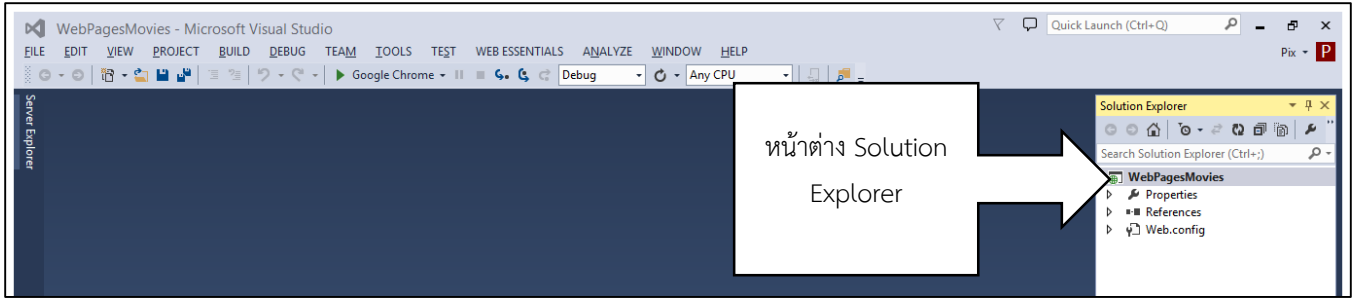


ในหน้าจอ New ASP.NET Project – WebPagesMovies ในช่อง Select a template ให้เลือก Empty แล้วคลิกปุ่ม OK



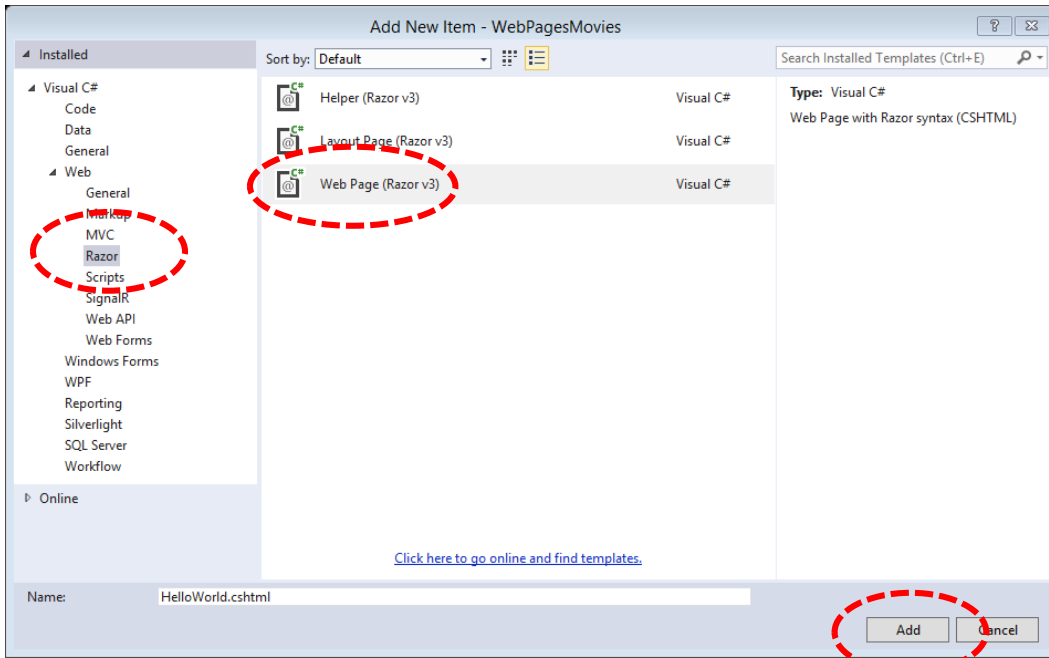
ทดสอบสร้างหน้าเว็บ

ต่อไปเราจะทดลองสร้างหน้าเว็บแบบง่าย ๆ 1 หน้าเพื่อเรียนรู้วิธีการสร้างหน้าเว็บใน Visual Studio 2013 โดยใช้หน้าต่าง Solution Explorer ที่ปรากฏด้านขวามือของโปรแกรม

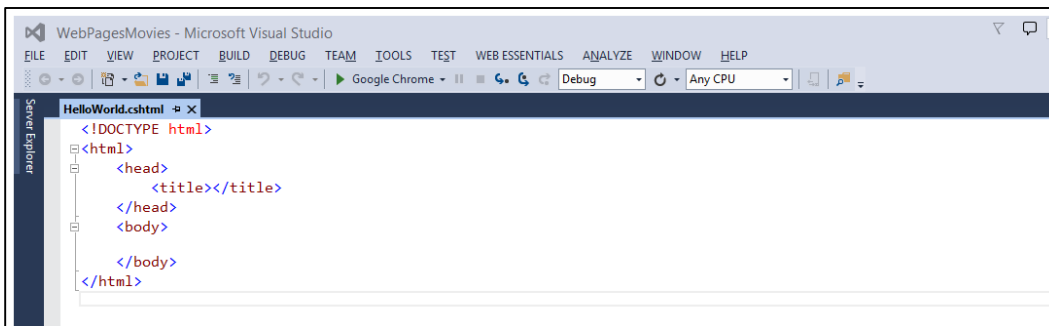


ที่หน้าต่าง Solution Explorer ทำการคลิกขวาที่ชื่อโครงการคือ WebPagesMovies เพื่อเลือกเพิ่มระดับบนสุดแล้วเลือกเมนูลอย Add > New Item

ที่หน้าต่าง Add New Item ให้เลือกที่ Razor > Web Page (Razor V3) ในช่อง Name ให้ตั้งชื่อไฟล์หน้าเว็บที่จะสร้างใหม่ว่า HelloWorld.cshtml แล้วคลิกปุ่ม Add



Visual Studio 2013 จะทำการสร้างไฟล์หน้าเว็บแบบ ASP.NET Web Pages และจะแสดงหน้าต่างเขียนคำสั่งสำหรับไฟล์ดังกล่าว



ในหน้าต่างเขียนคำสั่งเราสามารถแทรกคำสั่งภาษา HTML และภาษา C# ลงไปได้ในหน้าเดียวกันซึ่งจะกล่าวในภายหลัง แต่ให้สังเกตว่าในหน้าต่างเขียนคำสั่งจะมีการแสดงผลสีต่างๆที่แตกต่างกัน เรียกว่า syntax highlighting ซึ่งเป็นเครื่องมือช่วยอำนวยความสะดวกในการเขียนคำสั่ง

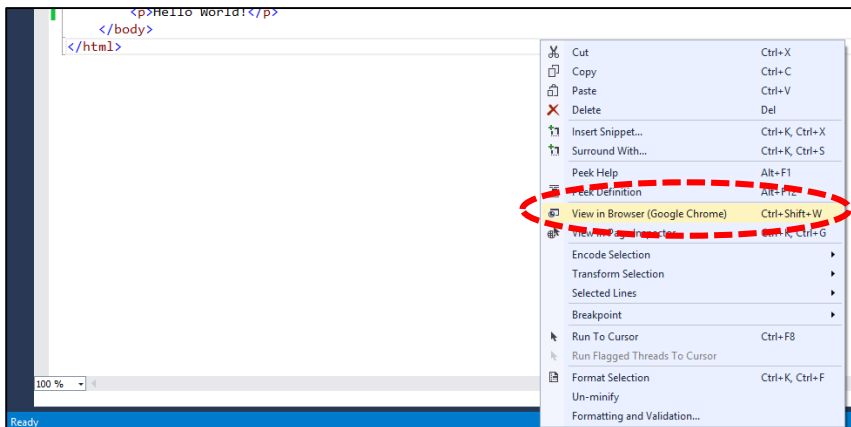
ขั้นต่อไปให้เขียนคำสั่งเพิ่มเติมภายในส่วนเนื้อหาของ <title> และ <body> ดังนี้

```
<!DOCTYPE html>
<html>
  <head>
    <title>Hello World Page</title>
  </head>
  <body>
    <h1>Hello World Page</h1>
    <p>Hello World!</p>
  </body>
</html>
```

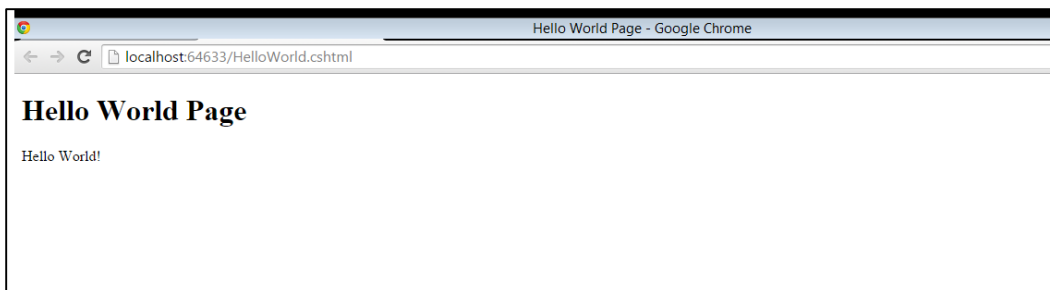
ให้ทำการบันทึกไฟล์ HelloWorld.cshtml ด้วยการเลือกเมนู File > Save หรือกดปุ่ม CTRL และ S บนคีย์บอร์ด

วิธีการทดสอบหน้าเว็บที่เราสร้าง

ให้ทำการเปิดไฟล์ HelloWorld.cshtml หากปิดไปแล้วด้วยการ Double Click ที่ชื่อไฟล์ในหน้าต่าง Solution Explorer หลังจากนั้นให้ทำการคลิกขวาแล้วเลือกเมนูล่อย View in browser



Visual Studio 2013 จะสั่งให้ Web Server สำหรับการทดสอบทำงาน (คือ IIS Express) ซึ่งสามารถใช้ในการทดสอบหน้าเว็บภายในเครื่องของเราก่อนที่จะติดตั้งใน Web Server จริง และจะทำการเปิดหน้าต่าง Web Browser ขึ้นมาและแสดงผลหน้าเว็บของเราโดยอัตโนมัติ



ตอนทดสอบหน้าเว็บของเราให้สังเกตว่าใน Web Browser จะแสดงที่อยู่ของหน้าเว็บว่า http://localhost:64633/HelloWorld.cshtml โดยที่หมายเลขท้ายคำว่า localhost จะเป็นหมายเลขที่สุ่มมาและแตกต่างกัน และคำว่า localhost หมายถึงใช้ Web Server ที่อยู่ภายในเครื่องของเรา (ในที่นี้คือ IIS Express) ซึ่งติดตั้งมาพร้อมกับ Visual Studio 2013

ตัวเลขท้ายคำว่า localhost คือ port number เป็นหมายเลขที่ใช้ติดต่อโดยตรงกับ IIS Express เฉพาะสำหรับเว็บแต่ละเว็บ โดยหากเราทำการเปิดโครงการหลายโครงการพร้อมกันก็จะได้ port number แตกต่างกันไปเพื่อให้สามารถทดสอบหน้าเว็บของเราแยกแต่ละโครงการได้ โดย port number จะถูกสุ่มเป็นตัวเลขระหว่าง 1024 ถึง 65536

หากคุณทำเว็บไซต์เสร็จและนำขึ้น Web Server จริงๆที่เป็นเครื่องอื่นคุณจะไม่สามารถใช้ localhost ในการเปิดหน้าเว็บของคุณบนเครื่อง Web Server ได้คุณจะต้องใช้ชื่อเครื่องของ Web Server นั้นแทน เช่น http://myhostingsite/mywebsite/HelloWorld.cshtml แทน โดยเราจะกล่าวถึงเพิ่มเติมในภายหลัง

ทดลองแทรกคำสั่งภาษา C#

ให้ทำการปิดหน้าจอ Web Browser ที่ทดสอบหน้าเว็บและกลับไปยัง Visual Studio 2013 และที่บรรทัดบนสุดของหน้าต่างเขียนคำสั่งของไฟล์ HelloWorld.cshtml ที่เปิดอยู่ให้พิมพ์คำสั่งดังนี้แทรกเข้าไปบนสุด

```
@{  
    var currentDateTime = DateTime.Now;  
}
```

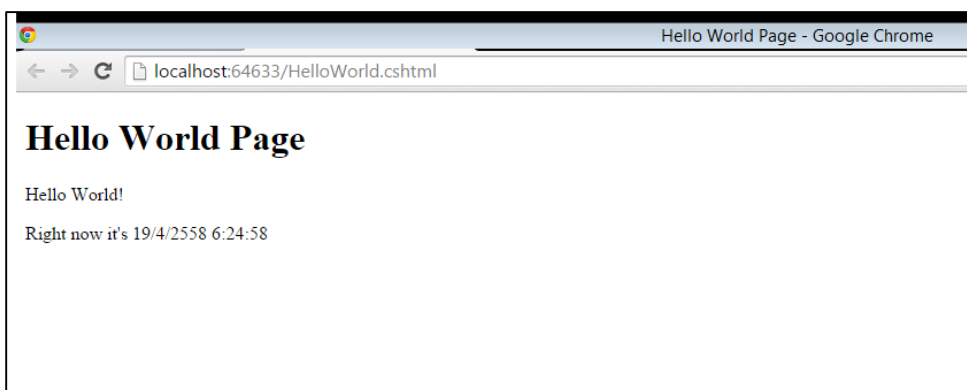
คำสั่งด้านบนเป็นภาษา C# ที่เขียนแทรกในหน้าเว็บโดยทั่วไปจะเรียกว่าคำสั่ง Razor (เรเซอร์) โดยความหมายของคำสั่งที่เพิ่มเข้าไปคือการดึงเอาวันและเวลา ณ ปัจจุบันไปเก็บไว้ในตัวแปรชื่อว่า currentDateTime เราจะกล่าวถึง Razor เพิ่มเติมในบทถัดๆไป

ต่อไปให้แทรกคำสั่งเพิ่มเติมใต้บรรทัด <p>Hello World!</p> ดังนี้

```
<p>Right now it's @currentDateTime</p>
```

คำสั่งบรรทัดนี้จะเป็นการดึงค่าวันและเวลาที่เก็บไว้ในตัวแปร currentDateTime ที่ได้ประกาศไว้ในบรรทัดบนสุดเพื่อมาแสดง ณ จุดที่เรากำหนดโดยการระบุเครื่องหมาย @ (แอท) นำหน้าหมายถึงการแทรกคำสั่ง Razor แบบด่วน (Inline code)

ทำการทดสอบหน้าเว็บนี้อีกทีตามกระบวนการที่คุณได้เคยทำแล้ว แต่คราวนี้ไม่ต้องทำการบันทึกไฟล์ HelloWorld.cshtml เนื่องจากทุกครั้งก่อนการทดสอบหน้าเว็บ Visual Studio 2013 จะทำการบันทึกให้เองอัตโนมัติ เมื่อหน้าเว็บที่แก้ไขปรากฏใน Web browser ก็จะมีการแสดงวันและเวลาปัจจุบันด้วย



เมื่อหน้าเว็บแสดงขึ้นมาให้ลองทำการ Refresh หน้าเว็บที่แสดงบน Web browser ด้วยการกดปุ่ม F5 บนคีย์บอร์ด คุณจะพบว่าเวลาจะถูกอัปเดตตลอดเวลา ให้ทดลองดูคำสั่ง HTML ที่แสดงบนหน้า Web browser ด้วยการคลิกขวาแล้วเลือกเมนูล้วย View source คุณจะพบกับคำสั่ง HTML ดังนี้

```
<!DOCTYPE html>

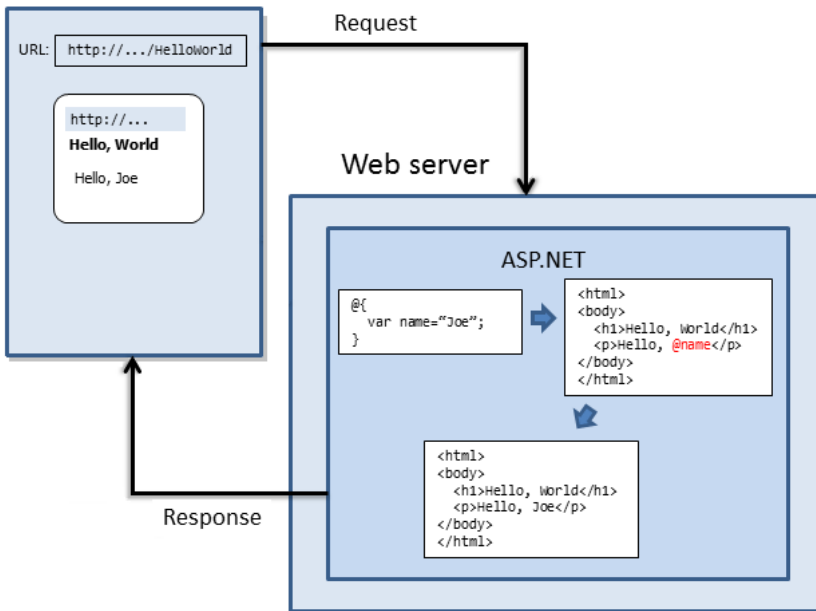
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <title>Hello World Page</title>
  </head>
  <body>
    <h1>Hello World Page</h1>
    <p>Hello World!</p>
    <p>Right now it's 1/18/2012 2:49:50 PM</p>
  </body>
</html>
```

สังเกตุดูให้ดีจะพบว่าคำสั่ง @{} ที่เราเพิ่มลงไปบรรทัดบนสุดไม่มีอยู่ และวันเวลาแสดงเป็นข้อความแทนที่จะเป็นคำสั่งว่า @currentDateTime โดยสิ่งนี้จะเกิดขึ้นเฉพาะกับไฟล์นามสกุล cshtml เท่านั้นโดยเป็นผลจากการประมวลผลคำสั่งของ ASP.NET ที่จะทำให้การประมวลผลคำสั่งต่างๆท้ายเครื่องหมาย @ แล้วนำผลลัพธ์ที่ได้แทรกกลับไปเป็นข้อความหรือคำสั่ง HTML ทั่วไปก่อนส่งกลับมาที่ Web browser เพื่อแสดงผล

การทำงานของ ASP.NET Web Pages

แนวคิดเกี่ยวกับการทำงานของ ASP.NET Web Pages ก็คือนอกเหนือจากการเขียนคำสั่ง HTML แล้วเรายังสามารถเขียนคำสั่ง Razor เพื่อประมวลผลข้อมูลต่างๆตามที่เราต้องการ โดยในตัวอย่างที่ผ่านมาคือการดึงเอาวันที่และเวลาปัจจุบันของระบบแล้วแสดงผล โดยเราสามารถแทรกคำสั่งนั้นเข้าไปโดยตรงในหน้าเว็บไม่ว่าจะเป็นส่วนไหนของหน้า เมื่อผู้ใช้งานเข้ามาดูหน้าเว็บของเราที่มีนามสกุลไฟล์เป็น cshtml ผ่าน Web browser ในเครื่องคอมพิวเตอร์ของเค้า ASP.NET จะทำการประมวลผลคำสั่งต่างๆและแทรกผลลัพธ์ที่ได้ ณ จุดที่เราเขียนคำสั่งไว้ แล้วทำการส่งผลลัพธ์หน้าเว็บที่ผ่านการประมวลผลแล้วในรูปแบบ HTML กลับมา โดยภาพด้านล่างแสดงกระบวนการทำงานที่ได้กล่าวไปนี้

Browser (Client)



แนวคิดตามภาพด้านบนเป็นแนวคิดพื้นฐานของ ASP.NET แต่ยังมีอีกหลายสิ่งที่เราสามารถทำได้ด้วยการเขียนคำสั่ง Razor รวมทั้งเราสามารถสร้างไฟล์ cshtml และเขียนคำสั่งอื่นได้เช่นเดียวกับไฟล์ HTML ปกติเช่นการใช้เขียนคำสั่ง JavaScript และการใช้งานกับ JQuery เป็นต้น สิ่งเหล่านี้คุณจะพบในบทถัดๆไป

บทถัดไป

ในบทถัดไปคุณจะได้ลงลึกลงไปในการเขียนคำสั่งต่างๆของ Razor เพิ่มขึ้นอีกนิดหน่อย

ผู้เขียน

- ต้นฉบับภาษาอังกฤษเขียนโดย Tom FitzMacken ซึ่งเป็น Senior Programing Writer ในหัวข้อ Web Platform & Tools Content team <http://www.asp.net/web-pages/overview/getting-started/introducing-aspnet-web-pages-2/getting-started>
- แปลเป็นไทยและปรับแต่งเนื้อหาโดยพลรบ สวัสดิ์ (phonrob.s@up.ac.th)