

---

# JavaScript

*Programming Guide*

---

สมบูรณ์ พัฒน์ธีรพงศ์



[www.kontentbluepub.com](http://www.kontentbluepub.com)

# JavaScript Programming Guide

โดย สมบูรณ์ พัฒน์ธีรพงศ์

ราคา : 250 บาท

พิมพ์ครั้งที่ 1 : ธันวาคม 2554

ข้อมูลทางบรรณานุกรมของสำนักหอสมุดแห่งชาติ

---

สมบูรณ์ พัฒน์ธีรพงศ์

JavaScript Programming Guide.--กรุงเทพฯ : คอนเทนต์บลู, 2554.

280 หน้า.

1. จาวาสคริปต์ (ภาษาคอมพิวเตอร์). I. ชื่อเรื่อง.

005.133

ISBN 978-616-91030-0-4

---

จัดทำโดย

: คอนเทนต์บลู พับลิชซิ่ง

เลขที่ 608/53-55 ถนนสาธุประดิษฐ์ แขวงบางโพงพาง

เขตยานนาวา กรุงเทพฯ 10120

โทรศัพท์ 0-2294-1663

<http://www.kontentbluepub.com>

พิมพ์ที่

: บริษัท เอ็น. วาย. ฟิล์ม จำกัด

เลขที่ 3 ซอยราชพฤกษ์ 4 ถนนราชพฤกษ์ แขวงบางจาก

เขตภาษีเจริญ กรุงเทพฯ 10160

โทรศัพท์ 0-2457-2596-7 โทรสาร 0-2868-6062

<http://www.nydigitalprint.net>

จัดจำหน่ายโดย

: บริษัท ซีอีดียูเคชั่น จำกัด (มหาชน)

อาคารเนชั่นทาวเวอร์ ชั้นที่ 19 เลขที่ 1858/87-90

ถนนบางนา-ตราด แขวงบางนา เขตบางนา กรุงเทพฯ 10260

โทรศัพท์ 0-2739-8222, 0-2739-8000 โทรสาร 0-2739-8356-9

<http://www.se-ed.com>

---

## แนะนำให้ผู้รู้จักกับ JavaScript

ในอดีตที่ผ่านมา JavaScript อาจเป็นเพียงแค่ภาษาสคริปต์ที่เพิ่มลูกเล่นและสีสันให้กับเว็บเพจเท่านั้น แต่ในปัจจุบัน JavaScript ถือได้ว่ามีบทบาทสำคัญอย่างยิ่งในการทำเว็บเพจและเว็บแอปพลิเคชัน เนื่องจากอินเทอร์เน็ตได้กลายมาเป็นส่วนหนึ่งของชีวิตประจำวันเสียแล้ว แต่เทคโนโลยีสำหรับการทำเว็บอย่าง HTML เพียงลำพัง ไม่สามารถรองรับความต้องการและความซับซ้อนของเว็บเพจได้ แม้ว่าเราจะมีเทคโนโลยีการทำเว็บฝั่งเซิร์ฟเวอร์ อาทิเช่น Java, PHP, Ruby, หรือ .Net ก็ตาม แต่นั่นไม่ได้ตอบโจทย์ทั้งหมดของความต้องการที่เพิ่มขึ้น เราต้องการให้เว็บเพจและเว็บแอปพลิเคชันสามารถตอบสนองผู้ใช้ได้รวดเร็วและมากยิ่งขึ้น ดังจะเห็นได้จากเทคนิคใหม่ๆ อาทิเช่น AJAX หรือ Widgets ต่างๆ ที่ทำให้ผู้ใช้ได้รับประสบการณ์ที่ดียิ่งขึ้น ซึ่งก็ล้วนแล้วแต่สร้างขึ้นมาจาก JavaScript ทั้งนี้ เว็บไซต์ที่ใหญ่ๆ อาทิเช่น Google, Hotmail, และ Facebook ต่างก็มีการนำ JavaScript มาประยุกต์ใช้ในเว็บเพจของตนเองอย่างมากมาย ลองปิดไฟเจอร์ JavaScript ในเว็บเบราว์เซอร์ของคุณซะครับ และลองเข้าเว็บไซต์ต่างๆ ที่เข้าใช้งานอยู่เป็นประจำ แล้วจะเห็นได้เลยว่า บางเว็บไซต์ไม่สามารถใช้งานได้เลย ดังนั้นจึงกล่าวได้ว่า JavaScript ได้กลายมาเป็นองค์ประกอบหลักองค์ประกอบหนึ่งของการทำเว็บในปัจจุบันแล้ว และในบทเรียนนี้ซึ่งเป็นบทเรียนแรกของการเรียนรู้ JavaScript เราจะเริ่มต้นด้วยหัวข้อต่างๆดังต่อไปนี้

- JavaScript คืออะไร?
- อะไรคือ Client-side JavaScript?
- ตัวอย่างของ JavaScript
- สิ่งที่ต้องใช้และต้องเรียนรู้

## JavaScript คืออะไร?

JavaScript คือภาษาสคริปต์ที่รันบนเว็บเบราว์เซอร์ เพื่อเพิ่มความสามารถให้กับเว็บเพจ ทำให้เว็บเพจสามารถตอบโต้กับผู้ใช้ได้มากขึ้น ยกตัวอย่างเช่น ใช้ JavaScript เพื่อสร้างเมนู เมื่อเลื่อนเมาส์ไปที่เมนูหลัก ก็จะเห็นเมนูย่อยปรากฏขึ้นมา, หรือใช้ JavaScript เพื่อทำ Tool Tips ให้กับฟอร์ม เพื่อใช้อธิบายการกรอกแบบฟอร์ม เป็นต้น ทั้งหมดนี้ HTML เพียงลำพังไม่สามารถทำได้ อันที่จริงแล้ว JavaScript เป็นมากกว่านั้น กล่าวคือ

- JavaScript คือภาษา ECMAScript ชนิดหนึ่ง !?! (โปรดดูที่หัวข้อ “ประวัติความเป็นมาของ JavaScript”)
- และเป็นภาษาโปรแกรมมิ่งที่รันบนและคอนโทรลอบเจกต์ต่างๆใน Host Environments เช่น รันบนเว็บเบราว์เซอร์, รันบนเว็บเซิร์ฟเวอร์, รันบนโปรแกรมกราฟฟิก เป็นต้น

อย่างไรก็ตาม JavaScript นั้นถูกใช้งานหลักๆบนเว็บเบราว์เซอร์หรือที่เรามักเรียกกันว่า Client-side JavaScript (Server-side Script ก็ได้แก่ JSP, PHP เป็นต้น) จึงทำให้เราเข้าใจว่า JavaScript ทำงานได้แค่บนเว็บเท่านั้น (ซึ่งไม่ใช่)

นอกจากนี้ JavaScript ยังมีคุณสมบัติและลักษณะทางภาษาดังต่อไปนี้

- เป็นภาษาสคริปต์ (ใช้การ Interpret แทนการ Compile นั่นคือรันโค้ดโดยตรงได้เลย โดยที่ไม่ต้องคอมไพล์ก่อนรันดังเช่นในภาษา C)
- เป็นภาษา Object-oriented ชนิดหนึ่ง (เป็น OO แบบ Prototype-based ไม่ใช่ Class-based)
- เป็นภาษา Dynamic-typing กล่าวคือชนิดของตัวแปรเปลี่ยนแปลงได้
- และเป็นภาษา Functional-oriented ด้วย เพราะฟังก์ชันถือได้ว่าเป็น Value ไม่ได้เป็นเพียงแค่ Syntax อย่างเดียวเท่านั้น

คุณสมบัติทางภาษาเหล่านี้ เราจะได้เรียนกันในบทถัดๆไป

## JavaScript ไม่ใช่ Java

JavaScript นั้นไม่ใช่ Java ทั้งสองภาษามีความแตกต่างกันโดยสิ้นเชิง เพียงแต่

JavaScript และ Java นั้นใช้ Syntax ที่ใกล้เคียงกัน คือรับอิทธิพลมาจากภาษา C นอกจากนี้ JavaScript ยังได้ใช้สไตล์การตั้งชื่อ (ตัวแปร, ฟังก์ชัน . . .) แบบเดียวกับ Java ด้วย และยังใช้ชื่อภาษาที่พ้องกันอีก (ดูความเป็นมาได้ที่หัวข้อถัดไป) สุดท้ายหากคุณสงสัยว่า JavaScript โดยมากใช้ทำเว็บ ซึ่ง Java ก็ใช้ทำเว็บได้เช่นกัน แล้วทั้งสองภาษาแตกต่างกันอย่างไร ก็ขอตอบดังนี้ JavaScript รันบนเว็บเบราว์เซอร์ (Client-side) ส่วน Java รันบนเซิร์ฟเวอร์ (Server-side) จึงทำให้ใช้งานทั้งคู่ไปพร้อมๆกันได้ ไม่ใช่เลือกอย่างใดอย่างหนึ่ง แต่ทั้งสองภาษาทำงานคนละส่วนกัน

---

**[Note]** Applet เป็นเทคโนโลยี Java ชนิดหนึ่ง ซึ่งรันบนเว็บเบราว์เซอร์เช่นกัน แต่เป็นคนละรูปแบบกับ JavaScript อีกทั้งปัจจุบัน Applet ไม่เป็นที่นิยมแล้ว อาจเป็นเพราะมันเกิดเร็วเกินไป !!!

---

**[Note]** เวลาพูดถึงคำว่า “เว็บเพจ” เราจะนึกถึงไฟล์ HTML ซึ่งถูกเขียนไว้ และเรียกดูที่เว็บเบราว์เซอร์ ในขณะที่คำว่า “เว็บแอปพลิเคชัน” เรานึกถึงการเขียนโปรแกรมฝั่งเซิร์ฟเวอร์เพื่อสร้างเอกสาร HTML ขึ้นมาตามการใช้งานของผู้ใช้

---

## ประวัติความเป็นมาของ JavaScript

JavaScript มีประวัติความเป็นมาโดยไล่เรียงแบบสรุปได้ดังนี้

- ถูกริเริ่มและพัฒนาโดย Brendan Eich ที่บริษัท Netscape เพื่อเพิ่มลูกเล่นให้กับการรันเว็บเพจบนเบราว์เซอร์ Netscape Navigator
- เริ่มต้นใช้ชื่อ Mocha และต่อมาได้เปลี่ยนไปเป็น LiveScript และสุดท้ายมาลงเอยที่ชื่อ JavaScript
- ชื่อ JavaScript นั้นเป็นผลมาจากการทำการตลาดร่วมกันระหว่างบริษัท Netscape และ Sun (เจ้าของเทคโนโลยี Java) ปัจจุบันทั้ง Sun และ Netscape ไม่มีตัวตนอีกแล้ว
- ในปี 1996 บริษัท Netscape ได้มอบ JavaScript ให้กับองค์กร ECMA International

เพื่อกำหนดและดูแลมาตรฐานของภาษา เนื่องจาก JavaScript เริ่มเป็นที่นิยมและรองรับในเว็บเบราว์เซอร์ต่างๆ และผลที่ตามมาคือ มาตรฐานภาษา ECMAScript (มันเป็นเพียงแค่มาตรฐานทางภาษาเท่านั้น ไม่ใช่ JavaScript Engine ที่ใช้ในเว็บเบราว์เซอร์แต่ประการใด) แต่คนทั่วไปก็ยังคงเรียกภาษานี้ว่า JavaScript

- ปัจจุบัน JavaScript Engine (ชื่อรหัส “SpiderMonkey”) ที่เคยเป็นของ Netscape ถูกพัฒนาต่อและดูแลโดยองค์กร Mozilla Foundation
- ส่วนเวอร์ชันของ JavaScript ดูได้ที่ตารางที่ 1-1

ตารางที่ 1-1: เวอร์ชันของ JavaScript

เวอร์ชัน	ปี	เว็บเบราว์เซอร์ที่รองรับ
1	มีนาคม ปีค.ศ. 1996	Netscape 2.0 และ IE 3.0
1.5 (ECMA-262 v3)	พฤศจิกายน ปีค.ศ. 2000	เว็บเบราว์เซอร์ส่วนใหญ่ในปัจจุบัน

หนังสือเล่มนี้ใช้ Syntax ของ JavaScript เวอร์ชัน 1.5 ซึ่งเป็นเวอร์ชันที่รองรับและใช้กันในทุกๆเว็บเบราว์เซอร์ แม้ว่าจะมีเวอร์ชันใหม่ออกมาแล้วก็ตาม แต่มีเพียงบางเว็บเบราว์เซอร์เท่านั้นที่รองรับ

---

**[Note]** JavaScript Engine เป็นหนึ่งใน Components ของการสร้างเว็บเบราว์เซอร์ ซึ่งทำหน้าที่รันโค้ด JavaScript โดยเฉพาะ แต่ละเว็บเบราว์เซอร์ก็จะมี Engine ของมันเอง ซึ่งประสิทธิภาพก็จะแตกต่างกันไป รายละเอียดเพิ่มเติมดูได้ที่ [http://en.wikipedia.org/wiki/JavaScript\\_engine](http://en.wikipedia.org/wiki/JavaScript_engine)

---

## อะไรคือ Client-side JavaScript?

JavaScript เป็นเพียงแค่ภาษาเท่านั้น ดังนั้นเราจึงต้องการตัวรัน JavaScript หรือ JavaScript Engine เนื่องจาก JavaScript Engine ถูกรวมเป็นส่วนหนึ่งของเว็บเบราว์เซอร์อยู่แล้ว (ส่วนใหญ่) ผลที่ได้ก็คือเราสามารถเขียน JavaScript และรันบนเว็บเบราว์เซอร์ได้

นั่นเอง และเรามักเรียกกันว่า Client-side JavaScript

Client-side JavaScript คือผลจากการรวมกันของภาษา JavaScript และออบเจกต์ต่างๆที่เว็บเบราว์เซอร์เตรียมไว้ให้ ทำให้เราสามารถเขียน JavaScript เพื่อคอนโทรลออบเจกต์ต่างๆหรือแม้แต่ตัวเอกสาร HTML ได้ ยกตัวอย่างเช่น เขียน JavaScript เพื่อควบคุมตัวตั้งเวลา (ออบเจกต์ Timer) เพื่อให้ทำงานตามช่วงเวลาที่เราต้องการ, เขียน JavaScript เพื่อสั่งเปิดหน้าต่างใหม่ (ออบเจกต์ Window), เขียน JavaScript เพื่อเปลี่ยนแปลงข้อความบนเว็บเพจ (ออบเจกต์ Node) เป็นต้น ออบเจกต์เหล่านี้มีอยู่บน Host Environment ที่เป็นเว็บเบราว์เซอร์เท่านั้น เราไม่สามารถไปควบคุมการกดปุ่ม Back บน Environments อื่นๆ ถ้าเรารัน JavaScript บน Environments อื่น เราก็ไม่เรียกมันว่า Client-side JavaScript

ด้วยเทคนิคการเขียน Client-side JavaScript นี้ ทำให้เราสามารถสร้างเว็บเพจแบบ Dynamic ได้ กล่าวคือเว็บเพจสามารถเปลี่ยนแปลงไปตามเหตุการณ์ต่างๆได้ JavaScript จึงเป็นหัวใจหลักของการทำ Dynamic HTML (DHTML) และ AJAX นั่นเอง และจากนี้ไปจะขอเรียก Client-side JavaScript สั้นๆว่า JavaScript เพราะหนังสือเล่มนี้เน้นใช้งานบนเว็บเบราว์เซอร์เท่านั้น

## JavaScript ทำอะไรได้บ้าง?

JavaScript สามารถทำดังต่อไปนี้ได้

- สามารถแทรกข้อความลงไปในเว็บไซต์ได้
- สามารถตอบโต้กับเหตุการณ์ต่างๆได้ เช่น เมื่อมีการขยับเมาส์
- สามารถเปลี่ยนแปลงส่วนต่างๆของเว็บไซต์ได้
- ใช้เพื่อตรวจสอบความถูกต้องของการกรอกข้อมูลในแบบฟอร์มได้
- ใช้ตรวจสอบชนิดของเว็บเบราว์เซอร์ได้

อันที่จริงแล้ว JavaScript ทำได้มากมาย ซึ่งคงไม่สามารถบรรยายได้หมด เขาเป็นว่าเว็บไซต์ใหญ่ๆที่เราเข้าใช้งานกันในปัจจุบัน ไม่ว่าจะเป็น Google Map, Gmail, Facebook, หรืออื่นๆ ต่างก็มีการนำ JavaScript มาประยุกต์ใช้อย่างมากมายมหาศาล รองปิด JavaScript ในเว็บเบราว์เซอร์ที่ใช้อยู่ขณะนี้ แล้วจะเห็นว่าไม่สามารถเข้าใช้งานบางเว็บไซต์ได้เลย

## ตัวอย่างแรกของ JavaScript

ในหัวข้อนี้เรามาลองดูตัวอย่าง JavaScript แบบง่ายๆกัน เพื่อให้เห็นภาพว่ามันมีหน้าตาเป็นอย่างไร ก่อนที่เราจะไปเจาะลึกกันในแต่ละบทต่อจากนี้ ตัวอย่างที่ 1-1 (JS\_Example.html) เป็นตัวอย่าง JavaScript ที่แสดงวันเวลาของการเปิดเว็บเพจ ดังแสดงในรูปที่ 1-1



รูปที่ 1-1: ผลลัพธ์การรัน JS\_Example.html บนเว็บเบราว์เซอร์

### ตัวอย่างที่ 1-1: JS\_Example.html

```
<html>
  <head>
    <title>JS_Example.html</title>
  </head>

  <body>
    <h1>Welcome to JavaScript</h1>
    <script type="text/JavaScript">
      // create a variable that holds value of
      // the present day and time
      var today = new Date();
      document.write( today );
    </script>
  </body>
</html>
```



จากตัวอย่างข้างต้นมีจุดที่น่าสนใจดังต่อไปนี้

- ตัวอย่างข้างต้นนี้แสดงข้อความ “Welcome . . .” พร้อมทั้งวันเวลาที่มีการเปิดเว็บเพจ
- โค้ด JavaScript ถูกเขียนอยู่ในไฟล์ HTML ด้วย Syntax ดังต่อไปนี้

```
<script type="text/JavaScript">  
    // for most browser  
</script>
```

หรือ

```
<script language="JavaScript">  
    // for very old browser  
</script>
```

ในปัจจุบันเรานิยมเขียนแบบแรกมากกว่า อันที่จริงแล้วเราสามารถเขียนสั้นๆดังต่อไปนี้ก็ได้

```
<script>  
    .  
    .  
    .  
</script>
```

- และโปรดสังเกตว่านามสกุลของไฟล์ยังคงเป็น **.html** ทั้งนี้เพราะเราเขียน

JavaScript แทรกลงไปในแท็ก HTML

- ตัวอย่างข้างต้นแสดงให้เห็นถึงสองสิ่งที่สำคัญต่อการเรียนรู้ JavaScript นั่นคือ ตัวภาษา JavaScript เอง และออบเจกต์จากเว็บเบราว์เซอร์
- ภาษา JavaScript และออบเจกต์สำเร็จรูป (Date) ที่มากับตัวภาษา
  - `var today = new Date();`
  - เป็นการประกาศตัวแปร `today` เพื่อใช้อ้างอิงออบเจกต์ `Date` และออบเจกต์ `Date` จะเก็บค่าวันที่ถูกรันโดยอัตโนมัติ
- ออบเจกต์จากเว็บเบราว์เซอร์ (Document)
  - `document.write( today );`
  - `document` คือตัวแปรที่ใช้อ้างอิงออบเจกต์ `Document` และเป็นตัวแปรที่มีให้ใช้ใน Client-side JavaScript โดยอัตโนมัติ (Implicit Variable) ไม่เหมือนกับตัวแปร `today` ที่ต้องประกาศขึ้นเอง
  - เมธอด `document.write()` ใช้สำหรับแทรกข้อความเข้าไปในเว็บเพจ

- เมื่อเราส่งตัวแปรชนิด Date เป็นพารามิเตอร์ไปยังเมธอด `document.write()` ออบเจกต์ Date จะถูกแปลงเป็นตัวอักษร (string) ให้โดยอัตโนมัติ
- JavaScript เป็น Case-sensitive ดังนั้นตัวอักษรพิมพ์เล็กหรือพิมพ์ใหญ่ถือว่าแตกต่างกัน
- เครื่องหมาย Semicolon “;” ถูกใช้เพื่อจบ Statement ซึ่งใน JavaScript จะเขียนหรือไม่ก็ได้ อย่างไรก็ตามผมแนะนำให้เขียนนะครับ
- เครื่องหมาย “//” ใช้สำหรับใส่คอมเมนต์แบบบรรทัดเดียว คือจบบรรทัด
- เครื่องหมาย “/\* . . . \*/” ใช้สำหรับใส่คอมเมนต์แบบหลายบรรทัด คือตั้งแต่ตัวเปิด “/\*” ไปจนกระทั่งเจอตัวปิด “\*/”

หากคุณไม่สามารถเข้าใจโค้ดได้ทุกบรรทัด ก็ไม่ต้องกังวล เพราะเราจะค่อยๆเรียนกันไปในแต่ละบท

---

**[Note]** ไฟล์ HTML สามารถประกอบไปด้วยแท็ก `<script>` มากกว่าหนึ่งชุดได้ และโค้ดในแต่ละชุดสามารถเรียกใช้ข้ามกันได้ เสมือนเขียนอยู่ในแท็ก `<script>` เดียวกัน เพียงแต่เรามักแยกเขียนตามความเหมาะสมและความจำเป็น

---

## สิ่งที่จำเป็นต้องใช้ในการเขียน JavaScript

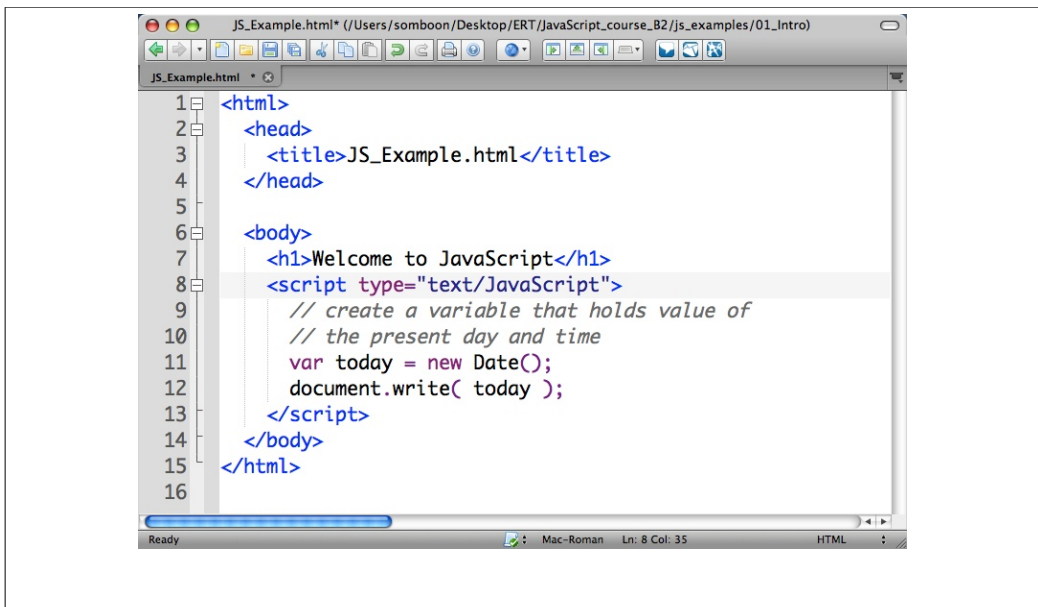
ในการเขียน JavaScript นั้น มีความต้องการเพียงแค่สองอย่างเท่านั้นคือ

- **เว็บเบราว์เซอร์** ซึ่งจะเป็น Firefox, Chrome, Safari, Opera, Internet Explorer, หรืออะไรก็ได้ที่รองรับ JavaScript
- **Text Editor** จะเป็น Text Editor อะไรก็ได้ แต่ขอให้ เป็น Text Editor ที่เข้าใจโค้ด HTML และ JavaScript ซึ่งจะทำให้สามารถแบ่งสีเส้นตาม Syntax ต่างๆได้ ทำให้ง่ายต่อการเขียนมากขึ้น

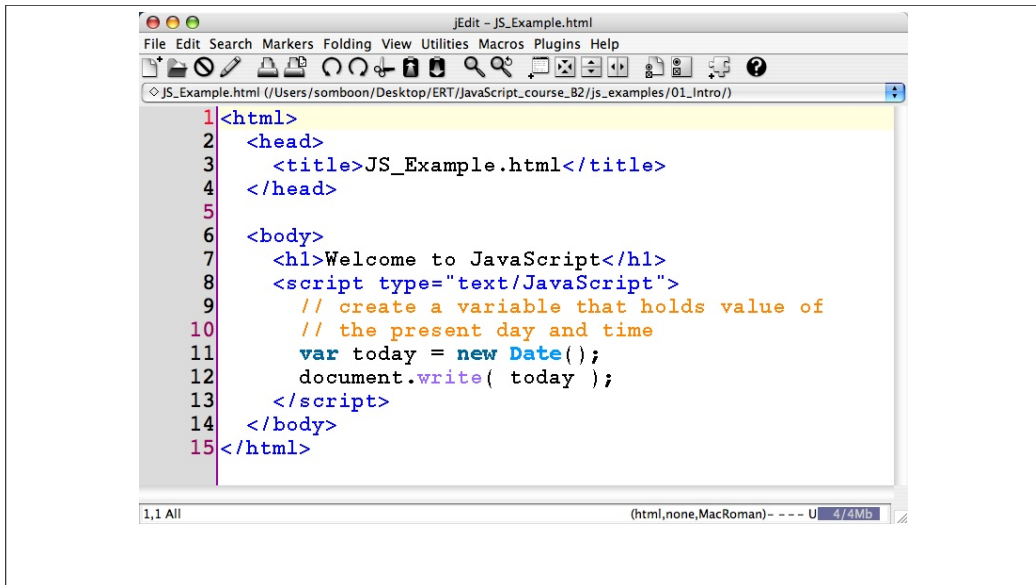
เราไม่ต้องการเว็บเซิร์ฟเวอร์แต่ประการใด เราใช้เพียงแค่เว็บเบราว์เซอร์ในการรันตัวอย่าง

เท่านั้น ยกเว้นในบทที่ 14 “AJAX” ที่เราต้องการเว็บเซิร์ฟเวอร์ และหากคุณยังไม่มี Text Editor ที่ถูกใจ ก็ลองพิจารณาสองตัวต่อไปนี้

- **Komodo Edit** สามารถดาวน์โหลดฟรีได้ที่ [www.activestate.com](http://www.activestate.com) Komodo Edit เป็นมากกว่า Text Editor คือมีความสามารถอื่นๆ IDE นั้นคือมีฟีเจอร์ที่สำคัญอย่าง Code Auto-complete ด้วย ซึ่งจะช่วยลดปัญหาการสะกดคำได้ นอกจากนี้เรายังสามารถเลือกดาวน์โหลดตัวติดตั้งได้ทั้งบน Window, Mac, และ Linux รูปที่ 1-2 แสดงตัวอย่าง Komodo Edit
- **JEdit** สามารถดาวน์โหลดได้ที่ [www.jedit.org](http://www.jedit.org) JEdit เป็น Text Editor แบบโอเพ่นซอร์สที่รองรับหลากหลายภาษาโปรแกรมมิ่ง ซึ่งแน่นอนว่ารวม HTML และ JavaScript ด้วย เนื่องจาก JEdit ถูกพัฒนาขึ้นด้วย Java จึงทำให้สามารถใช้ JEdit ได้บน Windows, Mac, Linux, และ Unix ได้ รูปที่ 1-3 แสดงตัวอย่าง JEdit



รูปที่ 1-2: ตัวอย่าง Komodo Edit



รูปที่ 1-3: ตัวอย่าง JEdit

---

**[Note]** การติดตั้ง JEdit จำเป็นจะต้องติดตั้ง JRE หรือ Java SE DK เสียก่อน ซึ่งการติดตั้ง Java SE DK สามารถดูได้ที่ภาคผนวกท้ายเล่ม

---

## Roadmap ของการเรียนรู้

การเรียนรู้ JavaScript นั้นก็ไม่ต่างไปจากการเรียนรู้ภาษาโปรแกรมมิ่งอื่นๆ กล่าวคือ เราจะต้องเรียนรู้สองสิ่งหลักๆ ดังนี้

- **ตัวภาษา** อันได้แก่ Syntax และกฎข้อบังคับทางภาษาต่างๆ เช่นการประกาศตัวแปร, การสร้างฟังก์ชัน เป็นต้น เนื้อหาที่เกี่ยวข้องจะอยู่ในบทเรียนบทต่างๆ ในภาคที่ 1 “Core JavaScript Language”
- **ไลบรารี** เมื่อเรียนรู้โครงสร้างทางภาษาแล้ว ก็ต้องเรียนรู้ไลบรารี มิเช่นนั้นแล้ว ลำพังแค่ภาษาคงจะทำอะไรมาไม่ได้ ไลบรารีในที่นี้ก็คือออบเจกต์ต่างๆ ที่สามารถเรียกใช้งานเว็บเบราว์เซอร์ได้นั่นเอง เนื้อหาที่เกี่ยวข้องจะอยู่ในบทเรียนบทต่างๆ ในภาคที่ 2 “Client-side JavaScript”

## การรัน JavaScript บนเว็บเบราว์เซอร์ที่เก่ามากๆ

แม้ว่าปัจจุบันเว็บเบราว์เซอร์จะรองรับ JavaScript กัน (แทบ) ทั้งหมดแล้ว แต่ก็อาจมีบ้างบางคนที่ยังใช้เบราว์เซอร์รุ่นเก่ามากๆอยู่ ที่ไม่รองรับ JavaScript ทำให้โค้ด JavaScript ถูกตีความและแสดงเป็นเพียงแค่อัฒมรรคาเท่านั้น ดังนั้นหากเรากำลังตรงจุดนี้ เราก็สามารถหลีกเลี่ยงปัญหานี้ได้โดยเขียนโค้ด JavaScript ไว้ในแท็กพิเศษดังนี้

```
<script type="text/JavaScript">  
  <!--  
    . . . JavaScript goes here . . .  
  // -->  
</script>
```

เท่านี้เว็บเบราว์เซอร์ที่ไม่รองรับ JavaScript ก็จะไม่โชว์โค้ดบนเว็บเพจแล้ว (อย่างไรก็ตาม โอกาสน้อยมากนะครับ ที่เว็บเบราว์เซอร์จะไม่รองรับ JavaScript และในกรณีนี้ก็ไม่จำเป็นรวมเว็บเบราว์เซอร์ที่ถูกปิดไฟเจอร์ JavaScript ด้วย การไม่รองรับกับรองรับแต่ปิดไฟเจอร์นั้นไม่เหมือนกัน แต่เอาเป็นว่าถ้ากังวลแล้วหละก็ ให้ใส่แท็กพิเศษข้างต้นไว้เสมอที่แล้วก็แล้วกัน)