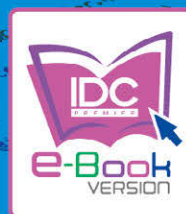


Basic



iOS App

Development



- เรียนรู้การพัฒนา iOS App ด้วย Xcode 5
- มีตัวอย่างประกอบทุกหัวข้อ พร้อม Workshops ที่สามารถทำงานได้จริง
- รองรับการพัฒนาทั้ง iPhone, iPad ทุกรุ่นที่ทำงานบน iOS 7
- เหมาะสำหรับผู้เรียน นักศึกษา และผู้ที่สนใจพัฒนา iOS App ด้วยตัวเอง

Free : Source Code

<http://www.infopress.co.th/devbook/iOSDevBook.rar>

■ สุกชัย สมพานิช ■ บรรณาธิการ กิตตินันท์ พลสวัสดิ์

special Thanks

Infopress Group ขอขอบคุณทุกการสนับสนุน
ที่มีเสมอมา จากประสบการณ์การถ่ายทอด
ความรู้ผ่านตัวหนังสือ เราจะมุ่งมั่นสร้างสรรค์งาน
ที่มีคุณภาพเพื่อคุณผู้อ่านตลอดไป ภาวักดิ์ชื่อที่ว่า
" The Best of Knowledge Books "



Basic iOS App Development

ผู้แต่ง

ศุภชัย สมพานิช

<http://www.facebook.com/thaiyb.net>

บรรณาธิการ

กิตินันท์ พลสวัสดิ์

kitinan_p@idcpremier.com

ออกแบบปก

वलันต์ พึ่งพูลผล

ออกแบบและจัดรูปเล่ม

วุฒิพันธ์ สมพระเมฆ, จิตราภรณ์ เหมาะจันทร์

พิสูจน์อักษร

สุนทรี บรรลือศักดิ์, มนฤดี ศรีอุทโยภาส

eBook Production

สุริยรัตน์ จิ๋ว, จิราภรณ์ โสภกา

Xcode เป็นเครื่องหมายการค้าของบริษัท Apple Inc. และเครื่องหมายการค้าอื่นๆ ที่อ้างถึงเป็นของบริษัทนั้นๆ

สงวนลิขสิทธิ์ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537 โดยบริษัท ไอดีซี พรีเมียร์ จำกัด ห้ามลอกเลียนไม่ว่าส่วนใดส่วนหนึ่งของหนังสือเล่มนี้ ไม่ว่าในรูปแบบใดๆ นอกจากจะได้รับอนุญาตเป็นลายลักษณ์อักษรจากผู้จัดพิมพ์เท่านั้น

บริษัท ไอดีซี พรีเมียร์ จำกัด จัดตั้งขึ้นเพื่อเผยแพร่ความรู้เทคโนโลยีสารสนเทศทั้งในระดับพื้นฐานและระดับสูง เรายินดีรับงานเขียนของนักวิชาการและนักเขียนทุกท่าน โดยเฉพาะงานที่เกี่ยวข้องกับสารสนเทศ ท่านผู้สนใจกรุณาติดต่อผ่านทางอีเมลที่ editor@infopress.co.th หรือโทรศัพท์หมายเลข 0-2962-1081 (อัตรา 10 คู่สาย) โทรสาร 0-2962-1084

ข้อมูลทางบรรณานุกรม



ศุภชัย สมพานิช

Basic iOS App Development

นนทบุรี : ไอดีซี, 2556

312 หน้า

1. ไอโอเอส (ระบบปฏิบัติการ)

2. การเขียนโปรแกรม

1 ชื่อเรื่อง

006.76

ISBN

978-616-200-494-0

พิมพ์ครั้งที่ 1

พฤศจิกายน 2556

จัดพิมพ์และจัดจำหน่ายโดย



บริษัท ไอดีซี พรีเมียร์ จำกัด

200 หมู่ 4 ชั้น 19 ห้อง 1901 อาคารจัสมินอินเตอร์เนชั่นแนลทาวเวอร์

ถ.แจ้งวัฒนะ อ.ปากเกร็ด จ.นนทบุรี 11120

โทรศัพท์ 0-2962-1081 (อัตรา 10 คู่สาย) โทรสาร 0-2962-1084

ราคา 275 บาท

สำหรับร้านค้าและตัวแทนจำหน่าย

โทรศัพท์ 0-2962-1081-3 ต่อ 112-114

โทรสาร 0-2962-1084

สมาชิกสัมพันธ์

โทรศัพท์ 0-2962-1081-3 ต่อ 121

โทรสาร 0-2962-1084

บรรณาธิการ

ในปัจจุบันหากจะกล่าวถึงอุปกรณ์ทาง IT คงไม่มีใครไม่รู้จัก iPhone หรือ iPad ซึ่งถือได้ว่าเป็นอุปกรณ์ Devices ที่ได้รับความนิยมสูงสุดในขณะนี้ และเมื่อพูดถึง iPhone หรือ iPad สิ่งที่ตามมาก็คือ App การใช้งานต่างๆ ที่มีนับหมื่นนับแสน อีกทั้งมียอดดาวน์โหลดนับล้านโหลด จึงปฏิเสธไม่ได้เลยว่า App สำหรับใช้งานบน iPhone, iPad หรืออุปกรณ์ Devices อื่นๆ บน iOS กำลังเป็นที่ต้องการของผู้ใช้งาน และนั่นก็คือทางเลือกที่น่าสนใจของนักพัฒนานั่นเอง ซึ่งปัจจุบันระบบปฏิบัติการของ iPhone, iPad หรืออุปกรณ์ Devices ต่างๆ ได้พัฒนาขึ้นมาเป็น iOS 7 โดยมี Xcode 5 เป็นเครื่องมือในการพัฒนา App ต่างๆ และนั่นเป็นที่มาของหนังสือเล่มนี้

หนังสือเล่มนี้จะแนะนำการพัฒนา App บน iOS โดยใช้ Xcode 5 เป็นเครื่องมือในการพัฒนา และรองรับการทำงานบน iOS 7 ซึ่งแนะนำให้ผู้อ่านเรียนรู้ตั้งแต่การติดตั้งเครื่องมือสำหรับพัฒนา, พื้นฐานภาษา Objective-C, การใช้งานฟีเจอร์ต่างๆ ของ iOS7 ตลอดจนไปถึงสามารถพัฒนา App ให้สามารถใช้งานได้จริง เหมาะสำหรับผู้เริ่มต้นที่ต้องการเรียนรู้การพัฒนา App อย่างละเอียดเป็นขั้นตอน และมีตัวอย่างประกอบทุกหัวข้อ

ทางสำนักพิมพ์หวังเป็นอย่างยิ่งว่า หนังสือเล่มนี้จะเป็นพื้นฐานที่ดีในการพัฒนา App บน iOS ทางเราพร้อมรับฟังข้อติชมทุกประการ หากหนังสือเล่มนี้มีข้อผิดพลาดประการใด ต้องขออภัยมา ณ ที่นี้

กิตินันท์ พลสวัสดิ์

พฤศจิกายน 2556

คำนำ

ในปัจจุบันคงไม่มีใครปฏิเสธได้เลยว่า การพัฒนา iPhone App ถือเป็นอีก 1 แพลตฟอร์มที่มีความสำคัญเป็นอย่างมากของนักพัฒนา App แต่สิ่งหนึ่งที่ยังมีปัญหาค่อนข้างมากก็คือ การเริ่มต้นนั่นเอง เนื้อหาที่นำเสนอในหนังสือเล่มนี้เป็นการปูฐานตั้งแต่ระดับง่ายที่สุด เรียงลำดับไปเรื่อยๆ จนกระทั่งถึงการทำ Workshops ให้คุณสร้าง iPhone App บน iOS 7 เห็นผลลัพธ์ได้อย่างชัดเจน สำหรับผู้อ่านที่ยังไม่เคยศึกษาภาษาใดๆ มาก่อนเลย ผู้เขียนขอแนะนำเป็นกรณีพิเศษว่า ให้คุณศึกษาเรียงตามลำดับบทที่ปรากฏในหนังสือเล่มนี้ ก็จะทำให้คุณเข้าใจได้ไม่ยากนัก ในกรณีที่ผู้อ่านต้องการสอบถามปัญหา หรือติชมเนื้อหาในหนังสือ สามารถโพสต์ได้ที่แฟนเพจของผู้เขียนที่ <https://www.facebook.com/thaivb.net>

ศุภชัย สมพานิช
พฤศจิกายน 2556

Contents

บทที่ 1 ก้าวแรกในโลกของการเขียน iOS App..... 1

เริ่มต้นอย่างไรดีกับการเขียน iOS App.....	1
แนวทางการศึกษาของการเขียน iOS App	2
ขอบเขตเนื้อหาของหนังสือเล่มนี้.....	2
การเตรียมสภาพแวดล้อมในการพัฒนา.....	3
การตรวจสอบเวอร์ชันของระบบปฏิบัติการ OS X.....	3
การสมัคร Apple ID.....	5
การดาวน์โหลด Xcode	6
ขั้นตอนการติดตั้ง Xcode	8

บทที่ 2 เริ่มต้นใช้งาน Xcode สำหรับพัฒนา iOS App..... 11

เริ่มต้นสร้าง App	11
ทดสอบรันโปรเจกต์.....	14
การเปิด/ปิดหน้าต่าง All Output.....	14
พื้นฐานการใช้ภาษา Objective-C ใน Xcode	15
เบื้องหลังการทำงานของ Xcode.....	17
การเปิดโปรเจกต์เดิม.....	17
การบันทึกโปรเจกต์.....	19
การปิดโปรเจกต์.....	20
วิธีการปิด Xcode	20
การกำหนดให้ iOS Simulator สามารถป้อนภาษาไทย.....	20
การกำหนดให้ใช้คีย์บอร์ดกับ iOS Simulator.....	22
เริ่มเขียนโค้ดภาษา Objective-C	23
ระบบช่วยเหลือในการเขียนโค้ด.....	25
ทำความเข้าใจกับโครงสร้างของโปรเจกต์แบบ Command Line Tool.....	26
การกำหนดให้ iOS Simulator กลับสู่ค่าเริ่มต้น.....	27



บทที่ 3 พื้นฐานไวยากรณ์ของภาษา Objective-C 29

ชนิดข้อมูลในโลกของ Objective-C	30
พื้นฐานการใช้งานตัวแปร	30
ชนิดข้อมูล (Data Type).....	31
ทำความเข้าใจกับตัวอักษรแสดงค่า (format code หรือ format string)	32
ขอบเขตของชนิดข้อมูลตัวเลข	34
การใช้งานข้อมูลชนิดตัวอักษร (char) และข้อความ (NSString)	34
การเขียนโปรแกรมเชิงวัตถุคืออะไร.....	36
OOP ในโลกของ Objective-C	36
พื้นฐานการสร้างคลาส.....	37
การกำหนดให้ตัวแปรออบเจกต์ไม่ใช่ออบเจกต์ด้วยคำสั่ง nil.....	43
ฟังก์ชันใน Objective-C	44
การสร้างฟังก์ชันแบบไม่มีการส่งค่ากลับด้วยคำสั่ง void (เมธอดของออบเจกต์)	44
การสร้างฟังก์ชันแบบมีค่าส่งกลับ.....	47
การสร้างฟังก์ชันแบบมีพารามิเตอร์.....	49
การสร้างฟังก์ชันโดยอาศัย Function Prototype	52
การสร้างเมธอดให้กับออบเจกต์.....	54

บทที่ 4 พื้นฐานการสร้าง iOS App แบบ Single View Application..... 59

การสร้าง iOS App แบบ Single View Application	59
ทำความเข้าใจกับสภาพแวดล้อมของ Xcode แบบพัฒนา iPhone App.....	62
ทำความเข้าใจโครงสร้างโปรเจกต์สำหรับ iOS Application	63
พื้นฐานการออกแบบส่วนแสดงผลใน iPhone App.....	64
การทดสอบ iPhone App ด้วย iOS Simulator.....	66
การปิด iOS Simulator	67
แนวความคิดของการเขียนโปรแกรมแบบ MVC.....	67
การสร้าง iPhone App แบบรับการส่งข้อมูลจากผู้ใช้.....	68
การกำหนดโครงสร้างในไฟล์ ViewController.h.....	70
การเขียนโค้ดทำงานในไฟล์ ViewController.m	72
การยกเลิกการเชื่อมต่อส่วนแสดงผล (Main.storyboard) กับโค้ดโครงสร้างในไฟล์ (*.h)	77
แนวทางการตั้งชื่อฟังก์ชันแบบ IBAction	77
พื้นฐานการรับและแสดงข้อมูล	82
การเชื่อมต่อส่วนแสดงผล (Main.storyboard) เข้ากับส่วนโครงสร้าง (*.h) ด้วยวิธีลากแล้ววาง	87

บทที่ 5 การใช้แอปพลิเคชันใน iOS App 93

การเพิ่มเติมความสามารถให้กับคลาสด้วยโปรโตคอล (Protocol).....	93
พื้นฐานการสร้างไอ้ดล็อกแจ้งเตือนด้วยคลาส UIAlertView	94
การสร้างไอ้ดล็อกแจ้งเตือนด้วยคลาส UIActionSheet	100
การสร้างไอ้ดล็อกแบบมีหลายปุ่ม	105
การแสดงไอ้ดล็อกคอยการทำงานด้วยคลาส UIActivityIndicator	109

บทที่ 6 พื้นฐานการสร้างส่วนแสดงผลแบบแถวด้วยออบเจ็กต์ Table View 115

ขั้นตอนการสร้างส่วนแสดงผลแบบ Table View.....	115
การเพิ่มข้อมูลในส่วนแสดงผลแบบ Table View และวิธีตรวจสอบแถวที่ถูกเลือก.....	121
การแสดงผลข้อมูลใน Table View ตั้งแต่ 2 sections ขึ้นไป	126
การแสดงผลข้อมูลตั้งแต่ 2 บรรทัดขึ้นไปในแต่ละเซลล์	133
พื้นฐานการตกแต่งเซลล์	138

บทที่ 7 พื้นฐานการสร้าง iPhone App แบบหลายหน้าจอด้วย Storyboard ...141

Storyboard คืออะไร.....	141
พื้นฐานการสร้างส่วนแสดงผลแบบ View ที่มี 2 หน้าจอ.....	142
พื้นฐานการสร้างส่วนแสดงผลแบบ Tab Bar ด้วย Tab Bar Controller.....	147
การกำหนดรูปให้กับปุ่มในแถบ Tab Bar โดยใช้รูปมาตรฐาน	150
การสร้าง iPhone App ด้วยโปรเจกต์แบบ Tabbed Application.....	153
การสร้างส่วนแสดงผลแบบบังคับให้ต้องตอบสนอง (Modal View).....	155
การกำหนดส่วนแสดงผลเริ่มต้น.....	161

บทที่ 8 การประยุกต์ใช้งานส่วนแสดงผลแบบแถว..... 163

พื้นฐานการสร้างส่วนแสดงผลแบบแถวในส่วนแสดงผลแบบ View	163
การค้นหาค่าข้อมูลในส่วนแสดงผลแบบ Table View.....	168
การสร้างส่วนแสดงผลแบบหลายหน้าจอแบบแถวด้วย Table View.....	176
การแสดงผลรูปภาพกำกับแต่ละรายการ.....	187
การสร้างส่วนแสดงผลแบบแถวตายตัว (Static Table View).....	198

บทที่ 9 การรับ/ส่งข้อมูลระหว่างกัน 203

พื้นฐานการรับ/ส่งข้อมูลระหว่างกัน.....	203
การรับ/ส่งข้อมูลกับส่วนแสดงผลแบบ Table View	211

บทที่ 10 พัฒนา iPhone App ด้วยโปรเจกต์ชนิดอื่นๆ ที่น่าสนใจ..... 221

การใช้งานโปรเจกต์แบบ Utility Application	221
การใช้งานโปรเจกต์แบบ Master-Detail	228
การแสดงผลละเอียดของแต่ละแถว	229
การเพิ่มรายการใหม่ใน Table View	230
การแก้ไข (ลบ) แถวใน Table View	230

บทที่ 11 การทำงานกับข้อมูลประเภท Media..... 239

การสร้าง App ดูคลิปวิดีโอ.....	239
การเล่นไฟล์คลิปวิดีโอจาก YouTube	246
การเล่นไฟล์เสียง.....	250

บทที่ 12 ระบบแผนที่และ Location..... 257

การกำหนดให้ iOS Simulator อ้างอิงพื้นที่ของโซนประเทศไทย	257
พื้นฐานการแสดงผลแผนที่ด้วย MapKit.framework.....	259
การแปลงที่อยู่ให้เป็นระบบพิกัด Latitude กับ Longitude	263
การแปลงระบบพิกัด Latitude กับ Longitude ให้เป็นที่อยู่	266
การปักหมุดในแผนที่ด้วยวิธีการระบุพิกัด Latitude กับ Longitude	269

บทที่ 13 Workshops 275

Workshop ที่ 1 : การสร้างไดอะล็อก Login	275
Workshop ที่ 2 : การสร้างระบบเมนูแบบไดอะล็อก	280
Workshop ที่ 3 : การแสดงผลรายการสินค้า	283

ตัวอย่างที่ 2-1	เริ่มต้นใช้งาน Xcode	11
ตัวอย่างที่ 2-2	เริ่มเขียนโค้ดภาษา Objective-C	23
ตัวอย่างที่ 3-1	พื้นฐานการใช้งานตัวแปร.....	31
ตัวอย่างที่ 3-2	ทำความเข้าใจกับตัวอักษรแสดงค่า (format code หรือ format string)	33
ตัวอย่างที่ 3-3	การใช้งานข้อมูลชนิดตัวอักษร (char) และข้อความ (NSString).....	35
ตัวอย่างที่ 3-4	พื้นฐานการสร้างคลาส.....	37
ตัวอย่างที่ 3-5	การสร้างฟังก์ชันแบบไม่มีการส่งค่ากลับด้วยคำสั่ง void (เมธอดของออบเจกต์).....	45
ตัวอย่างที่ 3-6	การสร้างฟังก์ชันแบบมีค่าส่งกลับ	47
ตัวอย่างที่ 3-7	การสร้างฟังก์ชันแบบมีพารามิเตอร์	49
ตัวอย่างที่ 3-8	การสร้างฟังก์ชันโดยอาศัย Function Prototype.....	52
ตัวอย่างที่ 3-9	การสร้างเมธอดให้กับออบเจกต์.....	54
ตัวอย่างที่ 4-1	การสร้าง iPhone App แบบรับการสัมผัสจากผู้ใช้งาน.....	68
ตัวอย่างที่ 4-2	แนวทางการตั้งชื่อฟังก์ชันแบบ IBAction.....	78
ตัวอย่างที่ 4-3	พื้นฐานการรับและแสดงข้อมูล.....	82
ตัวอย่างที่ 5-1	พื้นฐานการสร้างไดอะล็อกแจ้งเตือนด้วยคลาส UIAlertView	95
ตัวอย่างที่ 5-2	การสร้างไดอะล็อกแจ้งเตือนด้วยคลาส UIActionSheet	101
ตัวอย่างที่ 5-3	การสร้างไดอะล็อกแบบมีหลายปุ่ม	105
ตัวอย่างที่ 5-4	การแสดงผลไดอะล็อกคอยการทำงานด้วยคลาส UIActivityIndicator	109
ตัวอย่างที่ 6-1	พื้นฐานการสร้างส่วนแสดงผลแบบ Table View	115
ตัวอย่างที่ 6-2	การแสดงผลข้อมูลใน Table View ตั้งแต่ 2 sections ขึ้นไป	126
ตัวอย่างที่ 6-3	การแสดงผลข้อมูลตั้งแต่ 2 บรรทัดขึ้นไปในแต่ละเซลล์.....	133
ตัวอย่างที่ 7-1	พื้นฐานการสร้างส่วนแสดงผลแบบ View ที่มี 2 หน้าจอ.....	143
ตัวอย่างที่ 7-2	พื้นฐานการสร้างส่วนแสดงผลแบบ Tab Bar ด้วย Tab Bar Controller...147	
ตัวอย่างที่ 7-3	การสร้างส่วนแสดงผลแบบบังคับให้ต้องตอบสนอง (Modal View)	156
ตัวอย่างที่ 8-1	พื้นฐานการสร้างส่วนแสดงผลแบบแถวในส่วนแสดงผลแบบ View.....	163
ตัวอย่างที่ 8-2	การค้นหาข้อมูลในส่วนแสดงผลแบบ Table View.....	168
ตัวอย่างที่ 8-3	การสร้างส่วนแสดงผลแบบหลายหน้าจอแบบแถวด้วย Table View.....	176
ตัวอย่างที่ 8-4	การแสดงรูปภาพกำกับแต่ละรายการ.....	187
ตัวอย่างที่ 8-5	การสร้างส่วนแสดงผลแบบแถวตายตัว (Static Table View).....	199
ตัวอย่างที่ 9-1	พื้นฐานการรับ/ส่งข้อมูลระหว่างกัน.....	203
ตัวอย่างที่ 9-2	การรับ/ส่งข้อมูลกับส่วนแสดงผลแบบ Table View	211
ตัวอย่างที่ 10-1	การใช้งานโปรเจกต์แบบ Utility Application	221

ตัวอย่างที่ 10-2	การใช้งานโปรเจกต์แบบ Master-Detail	228
ตัวอย่างที่ 11-1	การสร้าง App ดูคลิปวิดีโอ	239
ตัวอย่างที่ 11-2	การเล่นไฟล์คลิปวิดีโอจาก YouTube	246
ตัวอย่างที่ 11-3	การเล่นไฟล์เสียง.....	250
ตัวอย่างที่ 12-1	พื้นฐานการแสดงผลแผนที่ด้วย MapKit.framework.....	259
ตัวอย่างที่ 12-2	การแปลงที่อยู่ให้เป็นระบบพิกัด Latitude กับ Longitude	263
ตัวอย่างที่ 12-3	การแปลงระบบพิกัด Latitude กับ Longitude ให้เป็นที่อยู่.....	266
ตัวอย่างที่ 12-4	การปักหมุดในแผนที่ด้วยวิธีการระบุพิกัด Latitude กับ Longitude.....	269
Workshop ที่ 1	การสร้างไดอะล็อก Login.....	276
Workshop ที่ 2	การสร้างระบบเมนูแบบไดอะล็อก	280
Workshop ที่ 3	การแสดงรายการสินค้า.....	284

ก้าวแรกในโลกของการเขียน iOS App

Chapter

1

หนังสือ Basic iOS App Development เล่มนี้ เป็นการศึกษาการใช้งานภาษา Objective-C และการพัฒนา iOS App อย่างง่าย เหมาะสำหรับผู้อ่านที่ต้องการศึกษาไวยากรณ์ของภาษา Objective-C ก่อนที่จะนำไปใช้ในการพัฒนา iOS App ในลำดับต่อไป

ผู้อ่านหลายคนอาจจะมีความคิดว่า ภาษา Objective-C ใช้พัฒนาแอปพลิเคชันสำหรับ iPhone ได้อย่างเดียวหรือไม่ สามารถนำไปพัฒนาแอปพลิเคชันสำหรับ iPad ได้หรือไม่ คำตอบคือ ภาษา Objective-C สามารถพัฒนาแอปพลิเคชันที่ทำงานได้ทั้งบน iPhone และ iPad หรือจะกล่าวอีกอย่างหนึ่งก็คือ ภาษา Objective-C สามารถใช้พัฒนาแอปพลิเคชันสำหรับทำงานบน iOS ได้ทุก iDevice นั่นเอง

แต่สำหรับหนังสือเล่มนี้ขอยกตัวอย่างการใช้ภาษา Objective-C ในการพัฒนาแอปพลิเคชันบน iPhone เพื่อเป็นแนวทางในการเรียนรู้เท่านั้น

จากประสบการณ์ของผู้เขียน ปัญหาของผู้เริ่มต้นที่ผู้เขียนให้ความสำคัญก็คือ “เริ่มต้นอย่างไรดี” และ “เริ่มต้นอย่างถูกต้อง” เป็นสิ่งที่ผู้เขียนต้องการนำเสนอให้ผู้อ่านเห็นว่าถ้าผู้อ่านเริ่มต้นอย่างถูกต้อง มีวิธีการที่ดี การพัฒนา iOS App ก็ไม่ใช่เรื่องยากอีกต่อไป

เริ่มต้นอย่างไรดีกับการเขียน iOS App

คำถามแรกที่หลายคนอยากทราบมากที่สุดก็คือ เราจะเริ่มต้นเขียน iOS App อย่างไรดี ถือเป็นคำถามที่ตอบยากมากที่สุดเช่นกัน เพราะว่าผู้อ่านแต่ละคนมีที่มาของปัญหาแตกต่างกัน แต่มีคำถามเดียวกัน ผู้เขียนขอยกตัวอย่างจุดเริ่มต้นของผู้เขียนมาเป็นกรณีศึกษาว่าผู้เขียนเริ่มต้นกับ iOS App อย่างไร

ผู้เขียนแยกที่มาของปัญหาของผู้อ่านแต่ละคนออกเป็น 2 กลุ่มใหญ่คือ

1. ผู้อ่านที่เคยศึกษาภาษาอื่นๆ มาบ้างในขั้นต้น ในขณะที่เรียนหรือเกิดจากการศึกษาด้วยตัวเอง เขียนโค้ดได้บ้างไม่ได้บ้าง เช่น ภาษา C, C++, Java, PHP, VB, C#, ASP.NET เป็นต้น ศึกษาไม่มากนักแตกต่างกันไปตามแต่ละบุคคล แล้วต้องการมาเขียน iOS App
2. ผู้อ่านที่เป็นโปรแกรมเมอร์ที่มาจากสายอื่นๆ แล้วต้องการเขียน iOS App

สำหรับตัวผู้เขียนเองถือว่าเป็นกลุ่มที่ 2 กล่าวคือ เป็นโปรแกรมเมอร์สาย .NET แล้วต้องการเขียน iOS App ซึ่งวงจรของการเรียนรู้โดยทั่วไปของผู้เขียนก็คือ เริ่มต้นหัดเขียนโปรแกรมง่ายๆ ก่อน จากนั้นศึกษาเพิ่มเติมเพื่อให้รู้มากยิ่งขึ้นว่า ถ้าเราต้องการทำโปรแกรมที่ซับซ้อนมากขึ้นจะมีวิธีการอย่างไร

แต่สำหรับการพัฒนา iOS App แล้ว มีข้อแตกต่างตรงกันข้ามอย่างสิ้นเชิงกล่าวคือ ไม่ว่าเราจะเขียนภาษา Objective-C เป็นหรือไม่เป็นก็ตาม โครงสร้างของ iOS App ถูกกำหนดให้เป็นมาตรฐานตั้งแต่เริ่มต้นเลย ถึงแม้ว่าเราจะเขียน App ง่ายๆ ก็ตาม

หมายความว่า ณ จุดเริ่มต้นของการเขียน iOS App มีทั้งการศึกษาภาษา Objective-C, การเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming-OOP) และการแบ่งหน้าที่การทำงานแบบ Model-View-Controller (MVC) ในเวลาเดียวกัน

เห็นได้ว่าถ้าพึ่งแค่การศึกษาภาษา Objective-C ก็ยากลำบากแล้ว ยังมีโครงสร้างที่เป็นมาตรฐานเข้ามาบังคับอีกชั้นหนึ่ง จึงทำให้การเริ่มต้นเขียน iOS App ค่อนข้างลำบากกว่าภาษาอื่นๆ อย่างแน่นอน

สิ่งที่กล่าวมาในข้างต้นคือ โจทย์ของหนังสือเล่มนี้ที่ผู้เขียนต้องตอบคุณผู้อ่านให้ได้ ต้องแก้ไขปัญหาที่ผู้เขียนเคยประสบมาให้ได้ ผู้เขียนอยากให้ผู้อ่านเริ่มต้นเขียน iOS App แบบยั่งยืน ไม่ฉาบฉวย เรียนรู้ และเข้าใจอย่างแท้จริง

แนวทางการศึกษาของการเขียน iOS App

แนวทางการนำเสนอเนื้อหาของหนังสือเล่มนี้ เหมาะสำหรับคุณผู้อ่านมือใหม่ที่ต้องการเริ่มต้นศึกษาโครงสร้างไวยากรณ์ของภาษา Objective-C และต้องการพัฒนา iOS App ในขั้นต้น ประกอบด้วย 2 โครงสร้างหลักคือ

1. **ศึกษาภาษา Objective-C โดยตรง** ประกอบด้วยการเรียนรู้ไวยากรณ์, หลักการของตัวภาษา, ชุดคำสั่งต่างๆ, คลาสสำเร็จรูปต่างๆ ที่ Apple เตรียมไว้ให้ เป็นต้น จากนั้นจึงเข้าสู่การเรียกใช้คลาสต่างๆ ของ iOS เพื่อสร้าง iOS App ตามที่ต้องการ
2. **ศึกษาการสร้าง iPhone App โดยตรง** เป็นการใช้งานกลุ่มคลาสต่างๆ ของ iOS เพื่อนำไปสร้าง iPhone App ตามที่ต้องการ การศึกษาตามแนวทางนี้ ผู้อ่านจะได้เรียนรู้ไวยากรณ์ของ Objective-C ในขณะที่พัฒนา iPhone App

ผู้อ่านสามารถเริ่มต้นจากทางใดก่อนก็ได้ ตามความเห็นของผู้เขียนแล้ว ไม่ว่าจะเลือกเดินทางไหนก่อนก็ตาม ท้ายที่สุดแล้วผู้อ่านก็หลีกเลี่ยงไม่พ้นที่จะต้องเรียนรู้ทั้ง 2 อย่าง เพื่อให้สามารถเขียนโค้ดอย่างมีคุณภาพนั่นเอง

ขอบเขตเนื้อหาของหนังสือเล่มนี้

แนวทางการนำเสนอเนื้อหาของหนังสือเล่มนี้ เหมาะสำหรับผู้อ่านมือใหม่ที่ต้องการเริ่มต้นพัฒนา iOS App เกิดจากแนวความคิดของผู้เขียนที่ว่าเป็นคู่มือแนวทางที่ดีให้กับผู้อ่านเพื่อมุ่งไปสู่การสร้าง iOS App ที่มีประสิทธิภาพในลำดับต่อไป

โค้ดตัวอย่างที่ลงในหนังสือเล่มนี้ทุกตัวอย่าง จะมีการบอกว่าโค้ดดังกล่าวเป็นของไฟล์ที่ชื่อว่าอะไร เพื่อให้ง่ายต่อการศึกษาโค้ดที่ผู้เขียนนำเสนอในแต่ละหัวข้อ

เนื้อหาที่นำเสนอในหนังสือเล่มนี้ครอบคลุมการพัฒนา iPhone App บน iOS 7 ขึ้นไป โดยอาศัย Xcode เวอร์ชัน 5 ขึ้นไป

การเตรียมสภาพแวดล้อมในการพัฒนา

ในการเขียนโปรแกรมภาษา Objective-C เราจะใช้เครื่องมือที่เรียกว่า Xcode ที่ติดตั้งบนระบบปฏิบัติการ OS X ซึ่งเป็นระบบปฏิบัติการที่ทำงานอยู่บนเครื่อง Mac และปัจจุบัน iOS ถูกพัฒนามาจนถึงเวอร์ชัน iOS 7 เพื่อให้การพัฒนา iOS App รองรับ iOS 7 อย่างสมบูรณ์แบบ ให้ผู้อ่านอัปเดต OS X เป็นเวอร์ชัน 10.8.x ขึ้นไป เพื่อให้สามารถรองรับ Xcode 5 ได้

NOTE



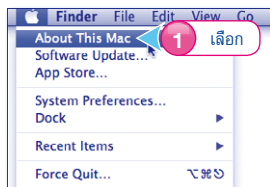
ในกรณีที่ผู้อ่านมีแต่เครื่อง PC (ติดตั้งระบบปฏิบัติการ Windows) ผู้อ่านสามารถใช้งาน Xcode ร่วมกับระบบปฏิบัติการ OS X บน PC ได้เช่นกัน ซึ่งผู้อ่านสามารถค้นหาวีธีการได้ใน Google

การตรวจสอบเวอร์ชันของระบบปฏิบัติการ OS X

1. ตรวจสอบระบบปฏิบัติการ OS X ว่าเป็นเวอร์ชันล่าสุดหรือไม่ สามารถทำได้โดยการคลิกปุ่ม  > About This Mac ดังรูปที่ 1-1

รูปที่ 1-1

แสดงเวอร์ชัน
ปัจจุบันของ OS X



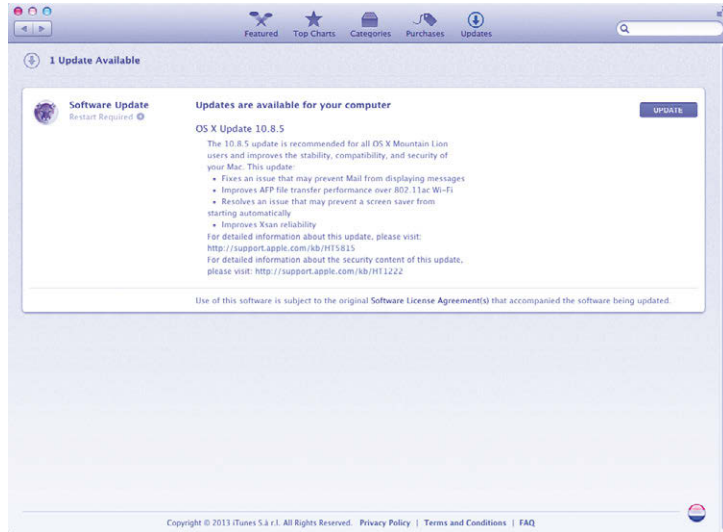
จากรูปที่ 1-1 พบว่าระบบปฏิบัติการ OS X ปัจจุบันของเครื่องที่ทดสอบเป็น OS X เวอร์ชัน 10.8.1 ไม่ใช่เวอร์ชันใหม่ตามที่ผู้เขียนต้องการ เนื่องจากเนื้อหาที่นำเสนอในหนังสือเล่มนี้ใช้ OS X เวอร์ชัน 10.8.5 ขึ้นไป เพื่อใช้ทำงานร่วมกับ Xcode 5 ขึ้นไป

2. คลิกปุ่ม  เพื่อตรวจสอบว่ามีรายการ Update หรือไม่ ดังรูปที่ 1-2



รูปที่ 1-2

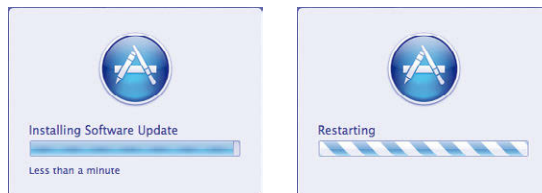
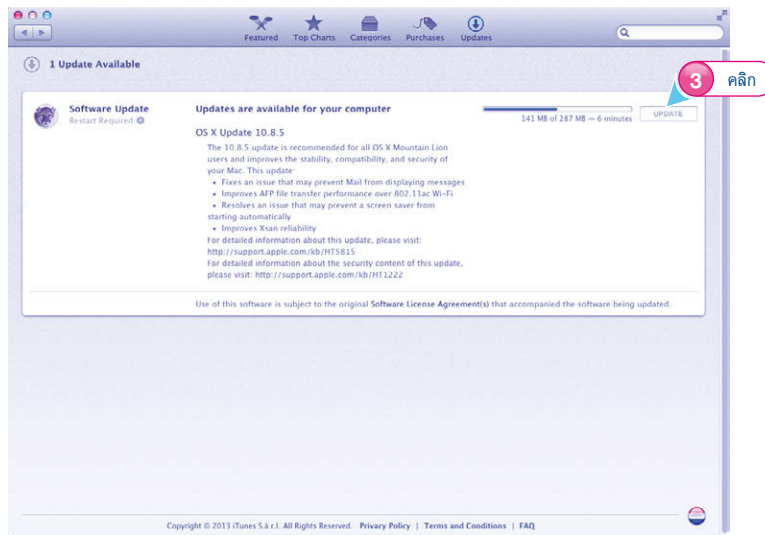
แสดงรายการ
อัปเดตที่ปรากฏ
ขึ้นมา



3. ท้ายที่สุดหลังจากดาวน์โหลดเสร็จแล้ว ก็จะเข้าสู่การอัปเดตระบบปฏิบัติการ OS X ให้เป็นเวอร์ชันใหม่โดยอัตโนมัติ ดังรูปที่ 1-3

รูปที่ 1-3

แสดงการอัปเดต
OS X เป็นเวอร์ชัน
10.8.5



ณ จุดนี้ OS X ของเราพร้อมพัฒนา iPhone App สำหรับ iOS 7 แล้ว

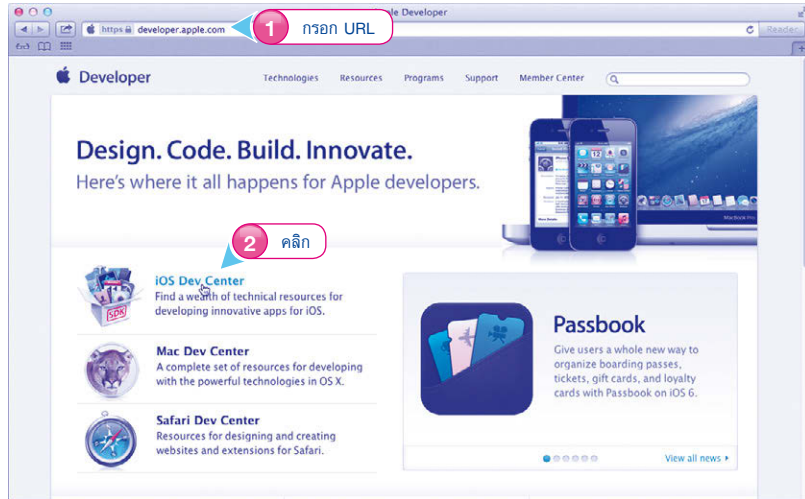
การสมัคร Apple ID

การดาวน์โหลด Xcode จากเว็บไซต์ของ Apple ผู้อ่านจะต้องมี Apple ID ก่อน ผู้อ่านสามารถสมัครได้ฟรีโดยไม่เสียค่าใช้จ่ายใดๆ ทั้งสิ้น โดยมีขั้นตอนดังนี้

1. ไปที่เว็บไซต์ของ Apple ที่ <https://developer.apple.com/>
2. คลิกที่ iOS Dev Center ดังรูปที่ 1-4

รูปที่ 1-4

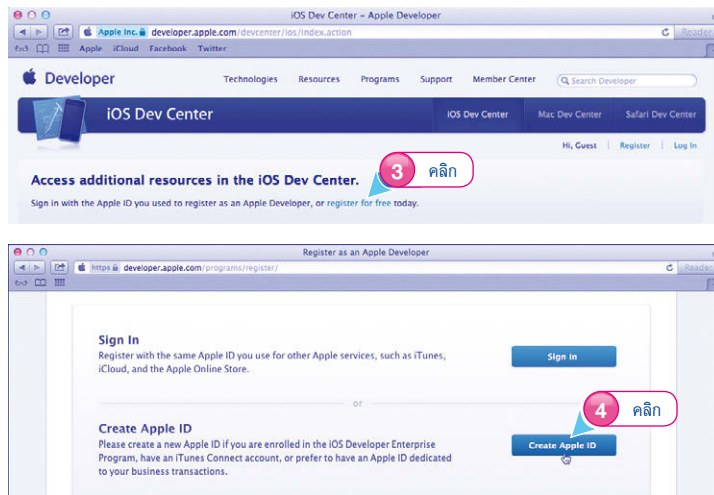
แสดงเว็บไซต์ของ
Apple



3. คลิกที่ register for free
4. คลิกปุ่ม Create Apple ID เพื่อเริ่มเข้าสู่ขั้นตอนการลงทะเบียน

รูปที่ 1-5

แสดงการลง
ทะเบียนเพื่อขอรับ
Apple ID



จากรูปที่ 1-5 ในระหว่างการลงทะเบียน ขอให้ผู้อ่านป้อนข้อมูลจริงตามที่ปรากฏขึ้นมา เพื่อใช้ในการยืนยัน iOS App ในโอกาสต่อไป

ขั้นตอนสุดท้ายของการลงทะเบียน ผู้อ่านต้องยืนยันตัวเองด้วย E-mail ของผู้อ่าน หากไม่มีข้อผิดพลาดใดๆ ในขั้นตอนการลงทะเบียนก็จะพบกับหน้าจอดังรูปที่ 1-6

รูปที่ 1-6

กรณีลงทะเบียน
เสร็จสมบูรณ์



จากรูปที่ 1-6 ณ จุดนี้ผู้อ่านจะมี Apple ID พร้อมใช้งานแล้ว

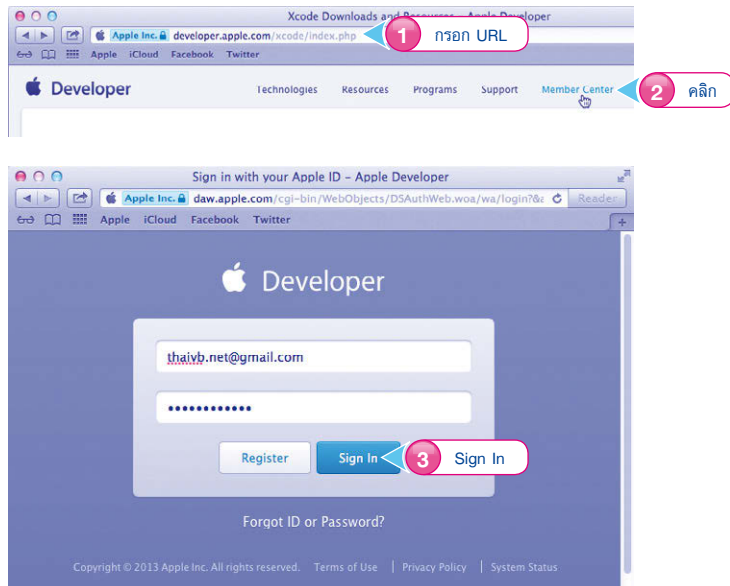
การดาวน์โหลด Xcode

หลังจากที่ผู้อ่านได้ Apple ID มาแล้ว ก็สามารถดาวน์โหลด Xcode มาใช้งานได้ฟรีโดยมีขั้นตอนดังนี้

1. กรอก URL ดังนี้ <https://www.developer.apple.com/xcode/index.php>
2. ให้คลิกที่ Member Center เพื่อ Sign In เข้าสู่ระบบด้วย Apple ID
3. Sign In เข้าสู่ระบบด้วย Apple ID ดังรูปที่ 1-7

รูปที่ 1-7

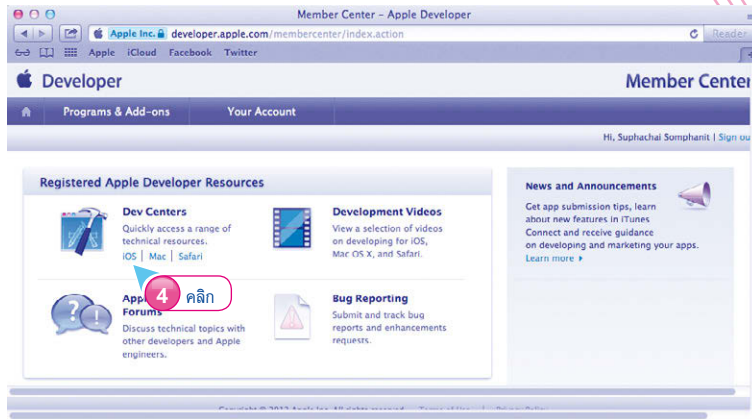
แสดงการ Sign
In ด้วย Apple
ID เข้าสู่ระบบ



4. ให้ผู้อ่านคลิกที่หัวข้อ iOS

รูปที่ 1-8

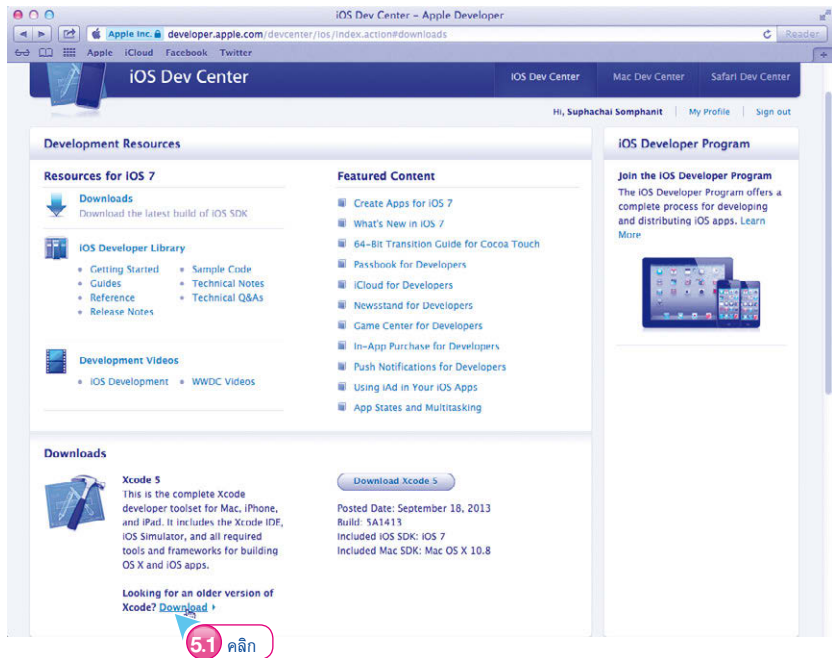
แสดงส่วนของ
Developer
สำหรับสมาชิก



5. ให้ผู้อ่านคลิกที่ Download ก็จะเข้าสู่หน้าจอต้อนรับ Xcode 5 สำหรับพัฒนา App บน iOS 7 ดังรูปที่ 1-9

รูปที่ 1-9

แสดงลิงค์
ดาวน์โหลด
Xcode 5 สำหรับ
iOS 7



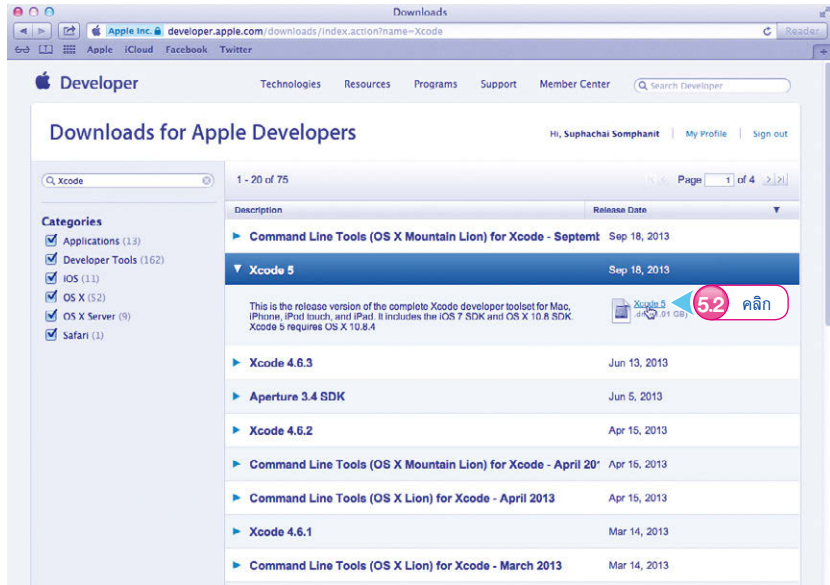
รูปที่ 1-9 (ต่อ)

แสดงลิงค์

ดาวน์โหลด

Xcode 5 สำหรับ

iOS 7



จากรูปที่ 1-9 ขอบเขตเนื้อหาของหนังสือเล่มนี้คือ การใช้งาน Xcode เวอร์ชัน 5 ขึ้นไป หลังจากที่ผู้อ่านดาวน์โหลด Xcode เสร็จเรียบร้อยแล้ว โดยปกติจะถูกเก็บอยู่ในโฟลเดอร์ Downloads ของ OS X

ขั้นตอนการติดตั้ง Xcode

ขั้นตอนการติดตั้ง Xcode มีดังนี้

1. ดับเบิลคลิกไฟล์ xcodedxxxxxxx.dmg เพื่อเริ่มต้นติดตั้ง ดังรูปที่ 1-10

รูปที่ 1-10

แสดงการตรวจสอบ

ความสมบูรณ์ของ

ไฟล์ติดตั้ง Xcode



2. ลาก Xcode เข้าสู่โฟลเดอร์ Applications เพื่อติดตั้ง Xcode เข้าสู่ระบบปฏิบัติการ OS X

รูปที่ 1-11

ระหว่าง

ติดตั้ง Xcode



รูปที่ 1-11 (ต่อ)

ระหว่าง
การติดตั้ง
Xcode



ในกรณีที่ OS X ของผู้อ่านมีการติดตั้ง Xcode เวอร์ชันเดิมอยู่ ผู้อ่านสามารถคลิกปุ่ม **Replace** เพื่อติดตั้ง Xcode เวอร์ชันใหม่ทับ Xcode เวอร์ชันเดิมได้เลย

3. ท้ายที่สุดถ้าการติดตั้งไม่มีข้อผิดพลาดใดๆ เกิดขึ้น ก็จะมีไอคอน Xcode ปรากฏใน Dock Bar ด้านล่าง ให้ผู้อ่านคลิกเพื่อทดลองเปิด Xcode ขึ้นมา ณ จุดนี้ Xcode ของผู้อ่านพร้อมใช้งานแล้ว ดังรูปที่ 1-12



รูปที่ 1-12

แสดงหน้าจอ
แรกของ Xcode



สรุปท้ายบท

เนื้อหาที่นำเสนอในบทนี้เป็นเพียงการเตรียมตัวสภาพแวดล้อมของ OS X ของเราให้พร้อมใช้งาน Xcode เพื่อพัฒนา iOS App ในลำดับต่อไปเท่านั้น



เริ่มต้นใช้งาน Xcode สำหรับพัฒนา iOS App

Chapter

2

เนื้อหาที่น่าสนใจในบทนี้ ถูกจัดอยู่ในระดับพื้นฐานสำหรับผู้ใช้อุปกรณ์ใหม่ควรทราบในขั้นต้น ผู้เขียนขอให้อ่านทำความเข้าใจกับโครงสร้างของโปรเจกต์ที่เกิดจาก Xcode ว่ามีส่วนประกอบใดบ้างที่เกี่ยวข้องกับการทำงานของเรา

เริ่มต้นสร้าง App

ในหัวข้อนี้ผู้อ่านจะได้เริ่มทดสอบว่า Xcode ของเราพร้อมใช้งานแล้วหรือไม่ โดยมีขั้นตอนดังนี้

ตัวอย่างที่ 2-1 เริ่มต้นใช้งาน Xcode

1. เริ่มต้นให้ดับเบิลคลิกไอคอน Xcode เพื่อเริ่มต้นเขียน App ดังรูปที่ 2-1

รูปที่ 2-1

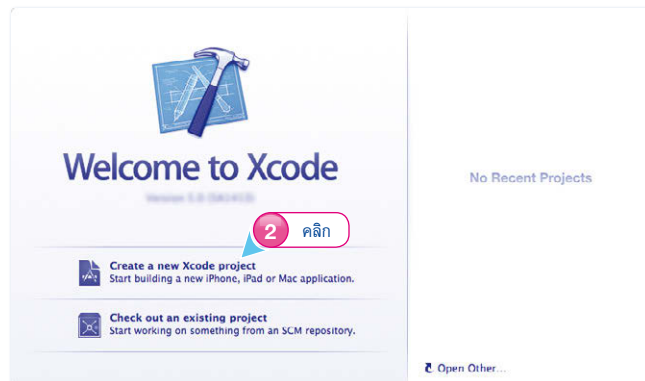
แสดงไอคอนของ
Xcode



2. คลิกที่ Create a new Xcode project เพื่อเริ่มต้นสร้างโปรเจกต์ว่างขึ้นมาใหม่ ดังรูปที่ 2-2

รูปที่ 2-2

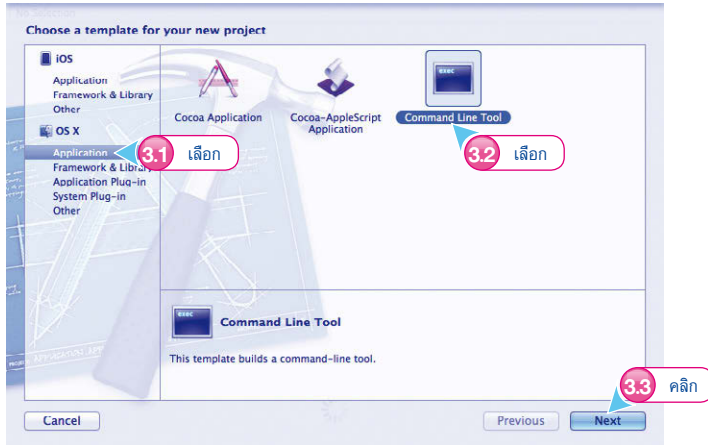
แสดงการคลิก
ที่ Create a new
Xcode project



3. สำหรับโปรเจกต์แรกของเราไม่ใช่ของ iPhone App แต่เป็นของ OS X ผู้เขียนต้องการทดสอบ Xcode ของผู้อ่านว่าพร้อมใช้งานแล้วหรือไม่เท่านั้น โดยการเลือกหัวข้อ Application ภายใต้ Template แบบ OS X และเลือกประเภทโปรเจกต์แบบ Command Line Tool (ป้อนคำสั่งแบบข้อความ)

รูปที่ 2-3

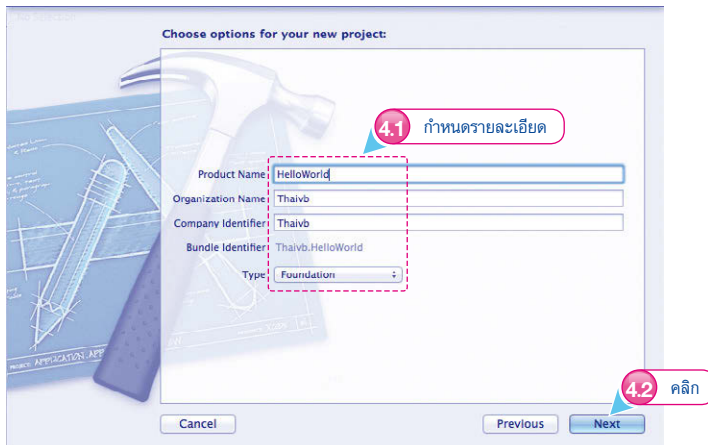
แสดงการเลือก
โปรเจกต์แบบ
Command Line
Tool



4. กำหนดรายละเอียดขั้นต้นของโปรเจกต์ปัจจุบัน โดยกำหนดดังนี้
- ◆ ช่อง **Product Name** หมายถึง ชื่อ App ของเรา ในกรณีนี้ตั้งชื่อว่า HelloWorld
 - ◆ ช่อง **Organization Name** และช่อง **Company Identifier** หมายถึง ชื่อองค์กรและบริษัท
 - ◆ ช่อง **Type** หมายถึง ประเภทของโปรเจกต์ ในกรณีนี้คลิกเลือก Foundation

รูปที่ 2-4

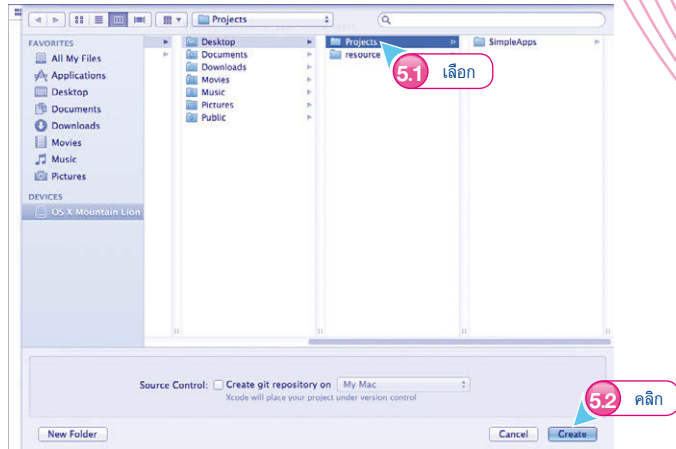
แสดงการกำหนด
รายละเอียด
ขั้นต้นของ
โปรเจกต์
ปัจจุบัน



5. เลือกโฟลเดอร์ที่ใช้เก็บโปรเจกต์ปัจจุบัน ในกรณีนี้คือ โฟลเดอร์ Projects หรือโฟลเดอร์ที่ผู้อ่านต้องการจัดเก็บโปรเจกต์ก็ได้ โดยคลิกปุ่ม **Create** เพื่อเริ่มต้นสร้างโปรเจกต์ต่างๆ ขึ้นมา ดังรูปที่ 2-5

รูปที่ 2-5

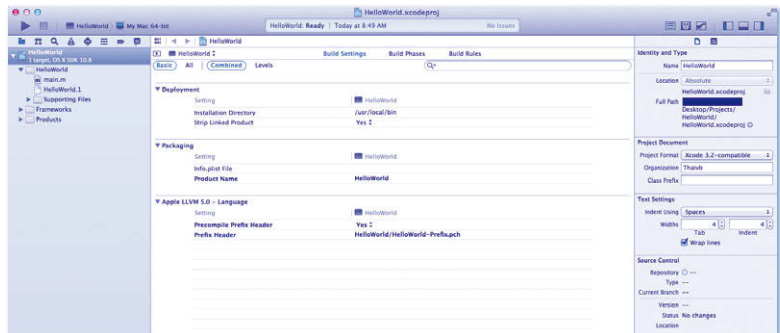
แสดงการสร้าง
โปรเจกต์ว่างๆ
ขึ้นมาในโฟลเดอร์
Projects



ตัวอย่างโปรเจกต์ต่างๆ ที่นำเสนอในหนังสือเล่มนี้ ผู้เขียนจัดเก็บไว้ในโฟลเดอร์ที่ชื่อว่า Projects บน Desktop ของ OS X สำหรับผู้อ่านก็แล้วแต่ว่าต้องการจัดเก็บไว้ที่ใด

รูปที่ 2-6

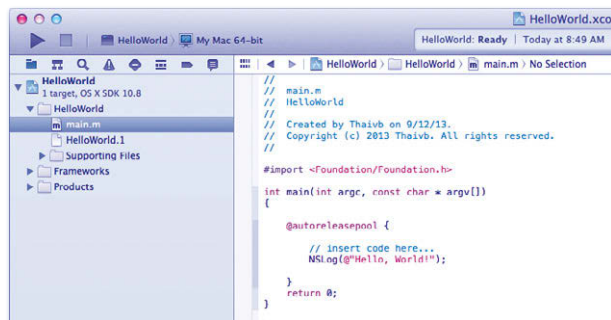
แสดงสภาพแวดล้อม
ของ Xcode แบบ
Command Line
Tool



6. ที่หน้าต่างด้านซ้ายแสดงโครงสร้างของโปรเจกต์ปัจจุบัน ซึ่งเป็นโปรเจกต์แบบ Command Line Tool

รูปที่ 2-7

แสดงโครงสร้าง
ของโปรเจกต์ที่ชื่อว่า
HelloWorld



จากรูปที่ 2-7 ไฟล์ main.m คือ ไฟล์แรก ที่ถูกสั่งให้ทำงานในโปรเจกต์แบบ Command Line Tool ให้ผู้อ่านคลิกที่ไฟล์ main.m เพื่อแสดงรายละเอียดโค้ด Objective-C ของไฟล์ main.m ดังได้ต่อไปนี่

โค้ดภาษา Objective-C ที่ 2-1 App แรกของเรา (main.m)

```
#import <Foundation/Foundation.h>

int main(int argc, const char * argv[])
{
    @autoreleasepool {
        // insert code here...
        NSLog(@"Hello, World!");
    }
    return 0;
}
```

จากโค้ด Objective-C ข้างต้นเกิดจาก Xcode เขียนคำสั่งให้ผู้อ่านโดยอัตโนมัติ ซึ่งก็คือ ฟังก์ชัน NSLog() ที่ทำหน้าที่สั่งให้พิมพ์ข้อความ “Hello, World!” ในช่อง All Output

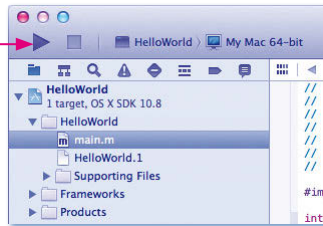
ทดสอบรันโปรเจกต์

การทดสอบรันโปรเจกต์ ให้ผู้อ่านคลิกปุ่ม  ที่อยู่มุมซ้ายด้านบน ดังรูปที่ 2-8

รูปที่ 2-8

แสดงปุ่มสำหรับ
รันโปรเจกต์

คลิก



รูปที่ 2-9

การข้อความแสดง
สถานะการรัน
โปรเจกต์ปัจจุบัน

แสดงสถานะการรันโปรเจกต์



จากรูปที่ 2-9 ในขณะที่เราสั่งให้รันโปรเจกต์ Xcode จะแสดงสถานะของการรันโปรเจกต์ให้เราทราบด้วย

การปิด/ปิดหน้าต่าง All Output

หน้าต่าง All Output ทำหน้าที่แสดงผลการทำงานที่เกิดจากการใช้คำสั่ง NSLog() การเปิดหรือปิดหน้าต่าง All Output ให้ผู้อ่านคลิกปุ่ม  เพื่อแสดงหน้าต่าง All Output ด้านล่าง ดังรูปที่ 2-10

รูปที่ 2-10

แสดงการเปิด/ปิด
หน้าต่าง All
Output

