

การเขียนโปรแกรมด้วย Python

สำหรับผู้เริ่มต้น



เริ่มต้นการเขียน
โปรแกรมภาษาไพธอน
ตั้งแต่ขั้นพื้นฐาน



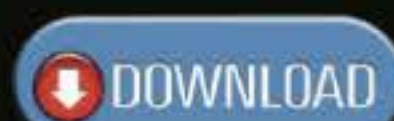
เขียนโค้ดบนเครื่องมือ
ที่ชาญฉลาดด้วย
PyCharm



ใช้เครื่องมือมาตรฐาน
ของไพธอนอย่าง
Python IDLE



พร้อมตัวอย่าง
และแบบฝึกหัด
ที่หลากหลาย



ดาวน์โหลดโค้ดหนังสือเล่มนี้ได้ที่:
<http://www.developerthai.com>

บัญชา ปะสีละเตสัง

ผู้เขียนหนังสือขายดีระดับ Best Seller
ด้าน Programming

การเขียนโปรแกรมด้วย Python สำหรับผู้เริ่มต้น



บัญชา ปะสีละเตสัง

ผู้เขียนหนังสือขายดีระดับ Best Seller
ด้าน Programming



บริษัท ซีเอดยูเคชั่น จำกัด (มหาชน)
SE-EDUCATION PUBLIC COMPANY LIMITED

ค้นหาหนังสือที่ต้องการ (รวม e-book และสินค้าที่น่าสนใจ) ได้เร็ว ทันใจ

- บน PC และ Notebook ที่ **www.se-ed.com**
- สำหรับ SmartPhone และ Tablet ทุกยี่ห้อ ที่ **http://m.se-ed.com** (ผ่าน browser เข้าอินเทอร์เน็ตแล้วทำ Bookmark บนจอ Home จะใช้งานได้เหมือน App ทุกประการ) หรือติดตั้ง **SE-ED Application** ได้จาก **Play Store** บน **Android** หรือจาก **App Store** บน **iOS**

- ในกรณีที่ต้องการซื้อเป็นจำนวนมาก เพื่อใช้ในการสอน การฝึกอบรม การส่งเสริมการขาย หรือเป็นของขวัญพิเศษ เป็นต้น กรุณาติดต่อสอบถามราคาพิเศษได้ที่ ฝ่ายขาย บริษัท ซีเอ็ดดูเคชั่น จำกัด (มหาชน) เลขที่ 1858/87-90 ถนนเทพรัตน แขวงบางนาใต้ เขตบางนา กรุงเทพฯ 10260 โทรศัพท์ 0-2826-8222 โทรสาร 0-2826-8356-9
- หากมีคำแนะนำหรือติชม สามารถติดต่อได้ที่ comment@se-ed.com

การเขียนโปรแกรมด้วย Python สำหรับผู้เริ่มต้น

โดย บัญชา ปะสีละเตลัง

ราคา 270 บาท

สงวนลิขสิทธิ์ตามกฎหมาย โดย บัญชา ปะสีละเตลัง

ห้ามคัดลอก ลอกเลียน ดัดแปลง ทำซ้ำ จัดพิมพ์ หรือกระทำการอื่นใด โดยวิธีการใดๆ ในรูปแบบใดๆ ไม่ว่าส่วนหนึ่งส่วนใดของหนังสือเล่มนี้ เพื่อเผยแพร่ในสื่อทุกประเภท หรือเพื่อวัตถุประสงค์ใดๆ นอกจากจะได้รับอนุญาต

4 1 0 - 9 0 4 - 3 8 0
0 2 6 6 0 9 8 7 6 5 4 3 2

ข้อมูลทางบรรณานุกรมของหอสมุดแห่งชาติ

บัญชา ปะสีละเตลัง.

การเขียนโปรแกรมด้วย Python สำหรับผู้เริ่มต้น.--กรุงเทพฯ : ซีเอ็ดดูเคชั่น, 2562.
380 หน้า.

1. ไพธอน (ภาษาคอมพิวเตอร์).

I. ชื่อเรื่อง.

005.133

ISBN : 978-616-08-3397-9

จัดพิมพ์และจัดจำหน่ายโดย



บริษัท ซีเอ็ดดูเคชั่น จำกัด (มหาชน)
SE-EDUCATION PUBLIC COMPANY LIMITED

เลขที่ 1858/87-90 ถนนเทพรัตน แขวงบางนาใต้ เขตบางนา กรุงเทพฯ 10260
โทรศัพท์ 0-2826-8000

พิมพ์ที่ บริษัท ส.เอเซียเพรส (1989) จำกัด

เลขที่ 143, 145 ซอยรามคำแหง 42 แขวงหัวหมาก เขตบางกะปิ กรุงเทพฯ 10240 โทรศัพท์ 0-2732-3101-6

นายสุวัฒน์ ปิยะธนาพงษ์ ผู้พิมพ์ผู้โฆษณา พ.ศ. 2562



คำนำ



ไพธอน (Python) เป็นหนึ่งในภาษาคอมพิวเตอร์ที่มีจำนวนผู้ใช้งานสูงสุดในปัจจุบัน เนื่องจากลักษณะโครงสร้างที่ไม่ซับซ้อน เรียนรู้ง่าย และความสามารถอันโดดเด่นในอีกหลายๆ ด้าน ซึ่งนอกจากจะเหมาะสำหรับผู้เริ่มต้นศึกษาด้านการเขียนโปรแกรมแล้ว ในภาคธุรกิจก็นำไพธอนไปพัฒนาระบบงานทางด้าน Data Science กันอย่างแพร่หลาย รวมถึงภาคอุตสาหกรรมแขนงต่างๆ ก็เริ่มนำไพธอนไปสร้างซอฟต์แวร์กันมากขึ้น นอกจากนี้สถาบันการศึกษาหลายๆ แห่ง ก็ยังเลือกไพธอนเป็นภาษาสำหรับการเรียนการสอนอีกด้วย ดังนั้น การเขียนโปรแกรมด้วยไพธอน จึงเป็นสิ่งสำคัญอีกอย่างหนึ่งสำหรับผู้ต้องการก้าวสู่สายงานด้านการพัฒนาซอฟต์แวร์ระดับมืออาชีพที่ควรต้องเรียนรู้เอาไว้

หนังสือเล่มนี้ได้รวบรวมเนื้อหาที่จำเป็นต้องรู้ทั้งหมดในขั้นพื้นฐาน พร้อมทั้งตัวอย่าง และแบบฝึกหัดท้ายบท รวมถึงเวิร์คช็อปอีกบางส่วน โดยใช้เครื่องมือยอดนิยมอย่าง Python IDLE และ PyCharm ซึ่งผู้เริ่มต้นการเขียนโปรแกรมด้วยไพธอน สามารถเรียนรู้ด้วยตนเอง และเป็นแนวทางสำหรับศึกษาเพิ่มเติมจากแหล่งข้อมูลอื่นๆ ต่อไปในอนาคตได้

บัญชา ปะสีละเตลัง

banchar_pa@yahoo.com

facebook.com/DeveloperThai



Python

Python

สารบัญ



unที่ 1	Python และเครื่องมือเขียนโค้ด.....	15
1-1:	เกี่ยวกับภาษา Python	15
1-2:	การติดตั้งและใช้งาน Python IDLE.....	16
	• การติดตั้ง IDLE	17
	• การใช้ IDLE ในเบื้องต้น	18
	• การกำหนดฟอนต์	21
1-3:	การติดตั้งและใช้งาน PyCharm	22
	• การติดตั้ง PyCharm	23
	• พื้นฐานการใช้ PyCharm.....	24
	• การกำหนดฟอนต์และธีม.....	28
unที่ 2	พื้นฐานการเขียนภาษา Python.....	31
2-1:	เริ่มต้นการเขียนและรันโปรแกรม.....	31
	• การเขียนโปรแกรมบน Python Shell Window.....	31
	• การเขียนโปรแกรมบน Python Edit Window.....	33
	• การเขียนโปรแกรมบน PyCharm.....	35
2-2:	คีย์เวิร์ดของภาษา Python.....	37
2-3:	Statement ในภาษา Python	38

2-4:	การเขียนคำอธิบายโค้ด.....	39
	• Line Comment.....	39
	• Block Comment.....	40
2-5:	ลักษณะของตัวแปร	41
2-6:	กำหนดค่าของตัวแปร.....	42
	• หลักพื้นฐานของการกำหนดค่าตัวแปร.....	42
	• การกำหนดข้อมูลชนิดตัวเลข	43
	• การกำหนดข้อมูลชนิดสตริง	44
2-7:	การแสดงผลด้วยฟังก์ชัน print().....	46
	• การใช้ฟังก์ชัน print() แบบพื้นฐาน	46
	• การใช้ฟังก์ชัน print() ที่ซับซ้อนขึ้น	47
2-8:	ลักษณะพื้นฐานของสตริงที่ควรรู้เพิ่มเติม.....	49
	• การเชื่อมต่อสตริง.....	49
	• การเขียนอักขระพิเศษบางตัวในสตริง	50
	• การกำหนดสตริงด้วย Triple Quotes.....	51
	• การแทรกข้อมูลในสตริง (Python 3.6+)	52
2-9:	การรับข้อมูลทางคีย์บอร์ดด้วยฟังก์ชัน input()	53
2-10:	แบบฝึกหัด	55

บทที่ 3 โอเปอเรเตอร์และข้อมูลตัวเลข 57

3-1:	ลักษณะของโอเปอเรเตอร์.....	57
3-2:	โอเปอเรเตอร์สำหรับการกำหนดค่า.....	58
3-3:	โอเปอเรเตอร์ทางคณิตศาสตร์.....	59
3-4:	โอเปอเรเตอร์สำหรับการคำนวณและกำหนดค่า	64
3-5:	ลำดับการประมวลผลของโอเปอเรเตอร์.....	68
3-6:	ฟังก์ชันเกี่ยวกับตัวเลข.....	69
	• ฟังก์ชันแบบ Built-in.....	69
	• ฟังก์ชันของโมดูล math	70
3-7:	การสร้างเลขสุ่ม	72
3-8:	การจัดรูปแบบของตัวเลข	74
3-9:	แบบฝึกหัด	75

unที่ 4	การเปรียบเทียบและกำหนดเงื่อนไข	79
4-1:	โอเปอเรเตอร์เปรียบเทียบ	79
4-2:	ลักษณะพื้นฐานของคำสั่ง if	82
4-3:	การกำหนดเงื่อนไขด้วย Comparison Operator.....	85
4-4:	การใช้คำสั่ง if-else	87
4-5:	การใช้คำสั่ง if-elif	89
4-6:	การใช้คำสั่ง if-elif-else.....	92
4-7:	การใช้คำสั่ง if ซ้อนกัน	93
4-8:	โอเปอเรเตอร์ทางตรรกะ.....	95
4-9:	ลำดับการประมวลผลของโอเปอเรเตอร์กลุ่มบูลีน.....	96
4-10:	การใช้ if ร่วมกับโอเปอเรเตอร์ not	97
4-11:	การกำหนดหลายเงื่อนไขด้วย Logical Operator.....	98
4-12:	การใช้ if-else แบบ Conditional Expression.....	100
4-13:	การกำหนดค่าตัวแปรตามเงื่อนไขโดยไม่ใช้ if	101
4-14:	แบบฝึกหัด	103
unที่ 5	การซ้ำแบบวนรอบ.....	107
5-1:	ฟังก์ชัน range().....	107
5-2:	การใช้รูปแบบ for-in.....	109
5-3:	การใช้รูปแบบ while.....	112
5-4:	การใช้รูปแบบซ้อนกัน.....	115
5-5:	การใช้คำสั่ง continue	116
5-6:	การใช้คำสั่ง break.....	117
5-7:	แบบฝึกหัด	119
unที่ 6	ข้อมูลแบบลำดับรายการ	121
6-1:	รายการข้อมูลแบบ List	121
	• การสร้างลิสต์	121
	• การอ้างถึงสมาชิกและใช้รูปแบบ for-in ร่วมกับลิสต์	122
	• การใช้โอเปอเรเตอร์ร่วมกับลิสต์.....	124
	• ฟังก์ชันแบบ Built-in ที่สามารถใช้ร่วมกับลิสต์.....	126
	• ฟังก์ชันของลิสต์.....	127

6-2:	List แบบ 2 มิติ	130
6-3:	รายการข้อมูลแบบ Tuple.....	133
	• การสร้างทูเพิล	133
	• การเข้าถึงสมาชิกของทูเพิล	134
	• การใช้โอเปอเรเตอร์และฟังก์ชันร่วมกับทูเพิล.....	135
	• ฟังก์ชันของทูเพิล.....	135
6-4:	รายการข้อมูลแบบ Set.....	136
	• การสร้างเซต	137
	• การเข้าถึงและจัดการเซต.....	138
	• ฟังก์ชันและปฏิบัติการเกี่ยวกับเซต.....	139
6-5:	รายการข้อมูลแบบ Dictionary.....	141
	• การสร้างดิกชันนารี.....	141
	• การเข้าถึงสมาชิกของดิกชันนารี.....	142
	• การใช้โอเปอเรเตอร์และฟังก์ชันร่วมกับดิกชันนารี.....	143
	• ฟังก์ชันของดิกชันนารี.....	143
6-6:	แบบฝึกหัด	145

บทที่ 7 การใช้สตริงเพิ่มเติมและวันเวลา 147

7-1:	ออบเจกต์ชนิด String.....	147
7-2:	ลำดับอักขระในสตริง	148
7-3:	จัดการสตริงด้วยโอเปอเรเตอร์และฟังก์ชัน	149
7-4:	ฟังก์ชันของออบเจกต์สตริง.....	155
	• การตรวจสอบสตริง	155
	• การค้นหาและแทนที่ภายในสตริง.....	158
	• การแยกและตัดช่องว่างในสตริง	160
	• การเปลี่ยนลักษณะตัวพิมพ์	160
	• การจัดแนวขอบของสตริง.....	161
7-5:	ข้อมูลชนิดวันเวลา	162
	• การกำหนดวันเวลาอ้างอิง	162
	• การอ่านค่าจากส่วนของวันเวลา.....	164
	• การหาระยะห่างของวันเวลา	165
	• การเพิ่มหรือลดวันเวลา.....	166

• การเปรียบเทียบวันเวลา	168
• การจัดรูปแบบวันเวลา.....	169
7-6: แบบฝึกหัด	170

บทที่ 8 ฟังก์ชันและโมดูล 173

8-1: ลักษณะของฟังก์ชัน.....	173
8-2: ฟังก์ชันแบบไม่ส่งค่ากลับ.....	175
• การสร้างฟังก์ชันแบบไม่ส่งค่ากลับ.....	176
• การเรียกใช้ฟังก์ชันแบบไม่ส่งค่ากลับ	176
8-3: ฟังก์ชันแบบส่งค่ากลับ	177
• การสร้างฟังก์ชันแบบส่งค่ากลับ.....	178
• การเรียกฟังก์ชันแบบส่งค่ากลับ	178
• การส่งคืนผลลัพธ์มากกว่า 1 ค่า.....	179
• วิธีหยุดการทำงานของฟังก์ชัน	180
8-4: พารามิเตอร์และอาร์กิวเมนต์	182
• Positional Argument.....	184
• Keyword Argument.....	184
• Default Parameter	185
• Variadic Parameter	186
8-5: การเรียกฟังก์ชันแบบ Recursion.....	188
8-6: ตัวแปรแบบ Global และ Local	189
8-7: ฟังก์ชันแบบ Lambda	192
8-8: การสร้างและใช้โมดูล.....	195
• การสร้างไฟล์ของโมดูล.....	195
• การกำหนดโค้ดของโมดูล	196
• การใช้โมดูล	197
8-9: แบบฝึกหัด	198

บทที่ 9 การตรวจสอบข้อผิดพลาด 201

9-1: ลักษณะของ Exception.....	201
9-2: การใช้คำสั่ง try-except และ else	202
9-3: การใช้คำสั่ง try ร่วมกับ finally.....	205

9-4: การระบุชนิดของข้อผิดพลาด.....	207
• การตรวจสอบข้อผิดพลาดชนิดเดียว	208
• การตรวจสอบข้อผิดพลาดหลายชนิด.....	209
9-4: การใช้คำสั่ง raise.....	211
9-5: การใช้คำสั่ง assert.....	212
9-6: แบบฝึกหัด	213

บทที่ 10 การอ่านและเขียนไฟล์ 215

10-1: การกำหนดตำแหน่งไฟล์.....	215
• การสร้างไฟล์ทดสอบ	215
• การระบุตำแหน่งไฟล์ในโค้ด Python.....	217
10-2: การเปิดไฟล์และตรวจสอบ Exception.....	218
10-3: การอ่านไฟล์.....	221
10-4: การเขียนไฟล์.....	223
10-5: ไฟล์แบบไบนารี	226
10-6: การจัดการไฟล์และไดเรกทอรี.....	228
• การตรวจสอบเกี่ยวกับไฟล์และไดเรกทอรี	228
• การกระทำกับไฟล์และไดเรกทอรี.....	229
10-7: แบบฝึกหัด	230

บทที่ 11 การสร้างคลาสในเบื้องต้น 231

11-1: คลาสและอินสแตนซ์.....	231
11-2: Instance Method.....	233
11-3: Instance Attribute และ Initializer.....	235
11-4: Class Attribute.....	238
11-5: Static Method.....	242
11-6: การสร้างคลาสในโมดูล	245

บทที่ 12 การสร้าง GUI ด้วย Tkinter (ตอนที่ 1) 247

12-1: พื้นฐานเกี่ยวกับ Tkinter	247
12-2: การจัดโครงสร้างแบบต่างๆ.....	249

• การจัดโครงร่างด้วยเมธอด pack().....	250
• การจัดโครงร่างด้วยเมธอด place().....	253
• การจัดโครงร่างด้วยเมธอด grid().....	255
12-3: การจัดโครงร่างซ้อนกันโดยใช้ Frame.....	259
12-4: การกำหนดสีและฟอนต์	260
• การกำหนดสี.....	260
• การกำหนดฟอนต์	261
• การกำหนดออปชันให้มีผลกับหลายวิดเจ็ต.....	262
12-5: องค์ประกอบพื้นฐานของ Widget.....	263
• ออปชันพื้นฐานของ Widget.....	263
• เมธอดพื้นฐานของ Widget.....	264
12-6: Button และออปชัน command.....	265
12-7: การใช้ Lambda กับออปชัน command	267
12-8: Label และ Message	269
12-9: Entry และ Text.....	272
• วิดเจ็ต Entry.....	272
• วิดเจ็ต Text.....	278
12-10: Message Box.....	279
12-11: Label Frame	282
12-12: Checkbutton และ Radiobutton.....	283
• วิดเจ็ต Checkbutton.....	283
• วิดเจ็ต Radiobutton	285
12-13: แบบฝึกหัด	290

unที่ 13 การสร้าง GUI ด้วย Tkinter (ตอนที่ 2) 293

13-1: Spinbox.....	293
13-2: Scale และ Scrollbar	295
• วิดเจ็ต Scale	295
• วิดเจ็ต Scrollbar.....	298
13-3: Listbox	300
13-4: Combobox.....	304

13-5: Progressbar	307
13-6: PhotoImage และ Canvas	310
13-7: Color chooser	312
13-8: File Dialog	313
13-9: Simple Dialog	317
13-10: Notebook	318
13-11: Menubar และ Menubutton	321
• การสร้าง Menubar	321
• การสร้าง Menubutton	322
13-12: แบบฝึกหัด	323

บทที่ 14 จัดการฐานข้อมูลด้วย Python..... 325

14-1: ฐานข้อมูล SQLite	325
14-2: การติดตั้ง DB Browser for SQLite	326
14-3: ฐานข้อมูลและตาราง	327
• การสร้างไฟล์ฐานข้อมูล	328
• ข้อกำหนดพื้นฐานของตาราง	328
• ชนิดคอลัมน์ใน SQLite (Column Type)	328
• ข้อกำหนดของคอลัมน์ (Column Constraint)	329
• การสร้างตาราง	329
• การเพิ่มข้อมูลสำหรับทดสอบ	330
• การทดสอบคำสั่ง SQL ใน SQLite Browser	332
14-4: การใช้ Python ร่วมกับ SQLite	334
• การนำเข้าไฟล์ฐานข้อมูล	334
• การเชื่อมต่อไฟล์ฐานข้อมูลกับ Python	335
• ออบเจกต์ Cursor	336
14-5: การอ่านและแสดงผลข้อมูล	337
• คำสั่ง SQL สำหรับการอ่านและกำหนดเงื่อนไข	337
• การอ่านข้อมูลจาก Cursor	340
• การแสดงผลข้อมูลแบบตารางโดยใช้โมดูล scrolltable	342
14-6: การกำหนด SQL แบบพารามิเตอร์	348

14-7: การเพิ่มข้อมูล.....	349
• คำสั่ง SQL สำหรับการเพิ่มข้อมูล.....	350
• การเขียนโค้ด Python สำหรับการเพิ่มข้อมูล	352
14-8: การแก้ไขและลบข้อมูล	353
• คำสั่ง SQL สำหรับการแก้ไขข้อมูล	353
• คำสั่ง SQL สำหรับการลบข้อมูล.....	353
• การเขียนโค้ด Python สำหรับแก้ไขและลบข้อมูล	354
14-9: แบบฝึกหัด	358

Unit 15 เวิร์กช็อป: เกม Craps 359

Workshop 15-1: เกม Craps	359
• การจัดโครงสร้างของ UI.....	361
• แนวคิดในการเขียนโค้ด.....	361
• ขั้นตอนการเขียนโค้ด	362
• การทดสอบ.....	365

Unit 16 เวิร์กช็อป: Address Book 367

Workshop 16-1: Address Book.....	367
• การสร้างฐานข้อมูล.....	368
• การจัดโครงสร้างของ UI.....	369
• แนวคิดในการเขียนโค้ด.....	370
• ขั้นตอนการเขียนโค้ด	371
• การทดสอบ.....	376




```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) (MSC v.1914 64 bit (AMD64)) on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Admin\Documents\PythonProjects\Exercises\1.py =====
กรรณาส่งรหัสได้ผ่าน (*9999*XXXXXXXXXX#) >>>*7777*1234567890#
ตำแหน่งที่ 2 ถึง 4 ต้องเป็นเลข 9
>>>
===== RESTART: C:\Users\Admin\Documents\PythonProjects\Exercises\1.py =====
กรรณาส่งรหัสได้ผ่าน (*9999*XXXXXXXXXX#) >>>*7777*1234567890#
ตำแหน่งที่ 1 และ 6 ต้องเป็น *
>>>
===== RESTART: C:\Users\Admin\Documents\PythonProjects\Exercises\1.py =====
กรรณาส่งรหัสได้ผ่าน (*9999*XXXXXXXXXX#) >>>*7777*1234567890#
ตำแหน่งที่ 7 ถึง 16 ต้องเป็นเลข 0 - 9
>>>
===== RESTART: C:\Users\Admin\Documents\PythonProjects\Exercises\1.py =====
กรรณาส่งรหัสได้ผ่าน (*9999*XXXXXXXXXX#) >>>*7777*1234567890#
ตำแหน่งที่ 17 ต้องเป็น #
>>>
===== RESTART: C:\Users\Admin\Documents\PythonProjects\Exercises\1.py =====
กรรณาส่งรหัสได้ผ่าน (*9999*XXXXXXXXXX#) >>>*7777*1234567890#
เราบันทึกข้อมูลท่านเรียบร้อยแล้ว
>>>
```

Python

Python

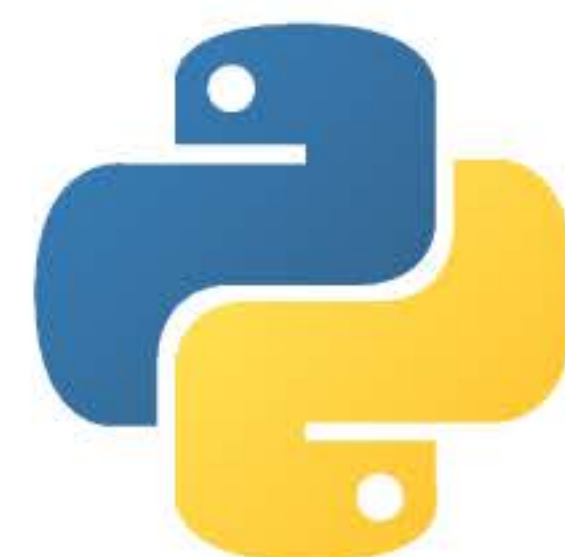
Python และเครื่องมือเขียนโค้ด



Python เป็นหนึ่งในภาษาคอมพิวเตอร์ที่ได้รับความนิยมสูงสุดในปัจจุบัน เพราะนอกจากจะง่ายต่อการเรียนรู้แล้ว ยังสามารถนำไปประยุกต์ใช้งานได้หลากหลายรูปแบบ โดยในบทนี้จะกล่าวถึงการติดตั้งเครื่องมือหรือโปรแกรมสำหรับการเขียนโค้ด พร้อมทั้งการตั้งค่าที่จำเป็นในเบื้องต้น ซึ่งผู้อ่านจะเลือกโปรแกรมใดก็ได้ตามความถนัด แต่ควรฝึกการใช้งานให้เกิดความคุ้นเคยในระดับหนึ่งก่อน ทั้งนี้เพื่อความรวดเร็วต่อการเรียนรู้ในบทต่อไป

1-1: เกี่ยวกับภาษา Python

ภาษาไพธอน (Python Language) สร้างขึ้นเมื่อปี ค.ศ. 1994 โดยโปรแกรมเมอร์ชาวดัตช์ที่ชื่อ Guido van Rossum ด้วยพื้นฐานจากหลายภาษารวมกัน เช่น C, ABC, Modular-3, Algol-68, SmallTalk, Unix Shell เป็นต้น ซึ่งมีวัตถุประสงค์เพื่อให้เป็นภาษาคอมพิวเตอร์ที่เรียนรู้ได้ง่าย ไม่มีกฎเกณฑ์หรือหลักไวยากรณ์ที่ซับซ้อน และสามารถนำไปใช้บนระบบปฏิบัติการที่แตกต่างกันได้ ปัจจุบันไพธอนจัดอยู่ในกลุ่ม Top-5 ของภาษาคอมพิวเตอร์ที่มีผู้ใช้งานมากที่สุด จึงเป็นสิ่งที่ยืนยันถึงความสำเร็จได้เป็นอย่างดี และนอกจากนี้ก็ยังมีลักษณะที่น่าสนใจอื่นๆ คือ



- **Free** - สามารถนำมาใช้งานได้ฟรี โดยไม่มีค่าใช้จ่าย หรือเงื่อนไขใดๆ
- **Easy to Use & Learn** - มีโครงสร้างทางภาษาที่เรียบง่าย สามารถเรียนรู้ภายในระยะเวลาอันสั้น จึงเหมาะอย่างยิ่งสำหรับผู้เริ่มต้นศึกษาการเขียนโปรแกรม

- **General Purpose** - ไพธอนเป็นภาษาแบบอเนกประสงค์ที่เราสามารถใช้ในการโปรแกรมได้หลากหลายรูปแบบ นับตั้งแต่แอปพลิเคชันทั่วไปสำหรับ Desktop เว็บแอปพลิเคชัน การโปรแกรมสำหรับเครือข่าย งานด้านกราฟิก หรือการคำนวณทางวิทยาศาสตร์ เป็นต้น
- **Portable** - สามารถนำโปรแกรมที่เขียนด้วยไพธอนบนระบบปฏิบัติการหนึ่ง ไปใช้ยังอีกระบบหนึ่งได้ เช่น เราอาจเขียนโปรแกรมบน Windows แล้วนำไปใช้งานบนเครื่อง Mac หรือ Linux เป็นต้น หรือเรียกว่าการทำงานข้ามแพลตฟอร์ม (Platform) นั่นเอง
- **Open Source** - ไพธอนเป็นภาษาเปิดเผย Source Code ภายใต้อิทธิพลของ Python Software Foundation ดังนั้น จึงมีกลุ่มอาสาสมัครเข้าร่วมพัฒนาอย่างมากมาย ช่วยให้ภาษามีวิวัฒนาการอย่างต่อเนื่อง
- **Functional & OOP** - สามารถเขียนโปรแกรมได้ทั้งแบบ Functional และแบบ Object-Oriented
- **GUI Programming** - ไพธอนสนับสนุนการเขียนโปรแกรมแบบ Graphic User Interface เพื่อติดต่อและโต้ตอบกับผู้ใช้แบบกราฟิก ซึ่งสามารถนำไปใช้งานบนระบบปฏิบัติการที่แตกต่างกันได้
- **Databases** - ไพธอนมีไลบรารีที่สนับสนุนการเชื่อมต่อกับโปรแกรมฐานข้อมูลหลักๆ เกือบทั้งหมด
- **Automatic Memory Management** - ภาษาไพธอนมีกลไกจัดการหน่วยความจำแบบอัตโนมัติ จึงช่วยลดข้อผิดพลาดจากการเกิดปัญหา Memory Leak และส่งผลให้โปรแกรมทำงานได้อย่างคงทน
- **Large Community & Support** - ภาษาไพธอนมีชุมชนหรือกลุ่มผู้ใช้ขนาดใหญ่ ที่พร้อมให้การสนับสนุนและช่วยเหลือเมื่อเราเกิดปัญหา
- **Support Libraries** - นอกจากไลบรารีแบบ Built-in ที่มากับภาษาไพธอนเองแล้ว เราสามารถนำไลบรารีจากภายนอกหรือ Third-Party Libraries เข้ามาเสริมการทำงานได้อีกด้วย จึงเกิดทางเลือกในการพัฒนาแอปพลิเคชันที่หลากหลายยิ่งขึ้น

1-2: การติดตั้งและใช้งาน Python IDLE

ก่อนที่จะเราสามารถทดสอบการทำงานต่างๆ ได้ จำเป็นต้องติดตั้งตัวประมวลผลของภาษาไพธอนรวมทั้งเครื่องมือสำหรับการเขียนโค้ดลงไปก่อน ซึ่งในหนังสือเล่มนี้ จะใช้เครื่องมือ 2 อย่างควบคู่กันคือ **Python IDLE** (Integrated DeveLopment Environment) และ **PyCharm** โดยจะกล่าวถึง Python IDLE ก่อน (ต่อไปจะเรียกสั้นๆ ว่า IDLE) ส่วน PyCharm ให้ดูที่หัวข้อถัดไป ซึ่ง IDLE เป็นเครื่องมือที่สร้างโดยทีมงานผู้พัฒนาภาษาไพธอนโดยตรง จึงอาจถือว่าเป็นมาตรฐานของภาษานี้ก็ได้ สำหรับรายละเอียดการติดตั้ง รวมทั้งการใช้งานที่ควรรู้จักในเบื้องต้น มีดังต่อไปนี้

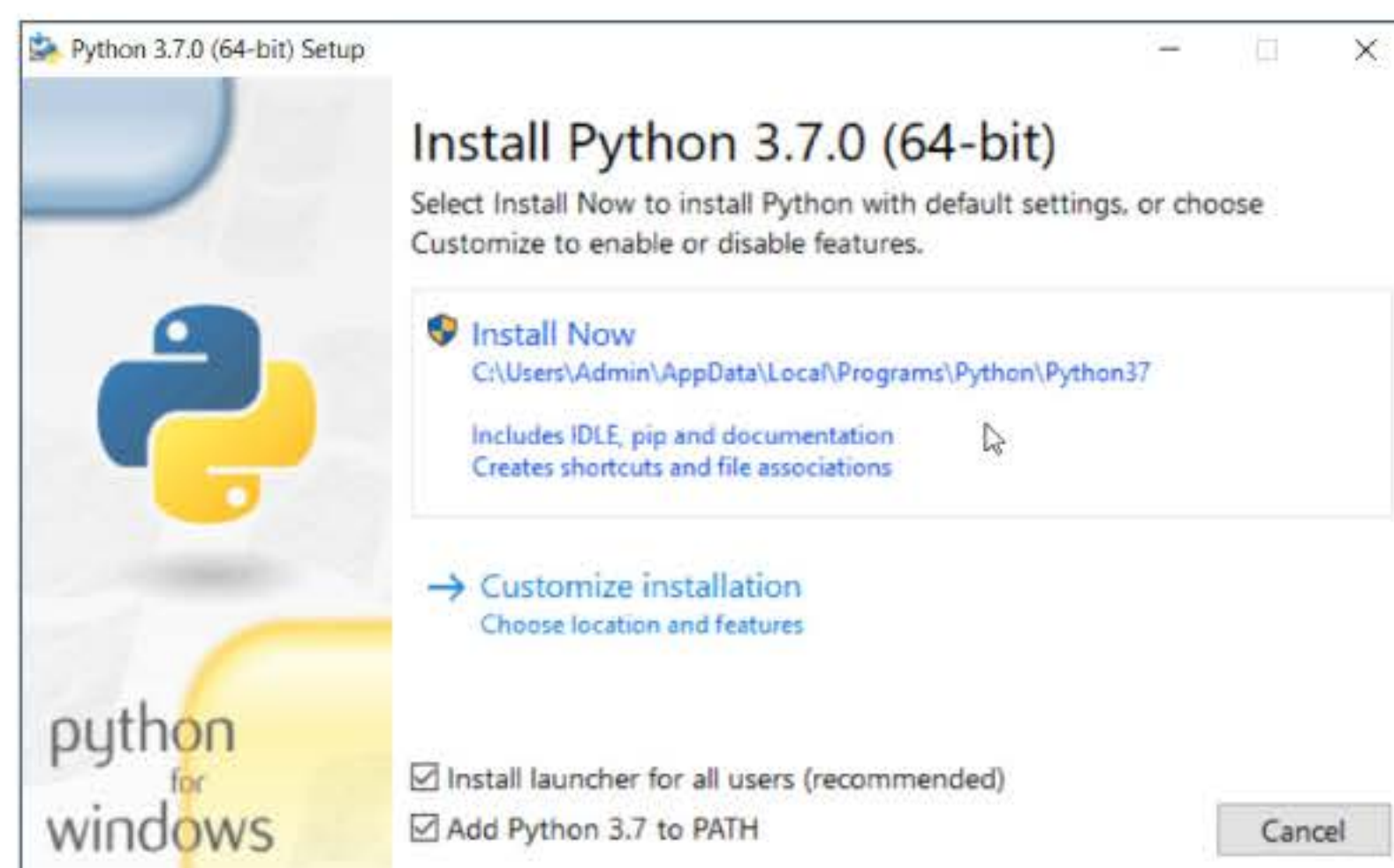
การติดตั้ง IDLE

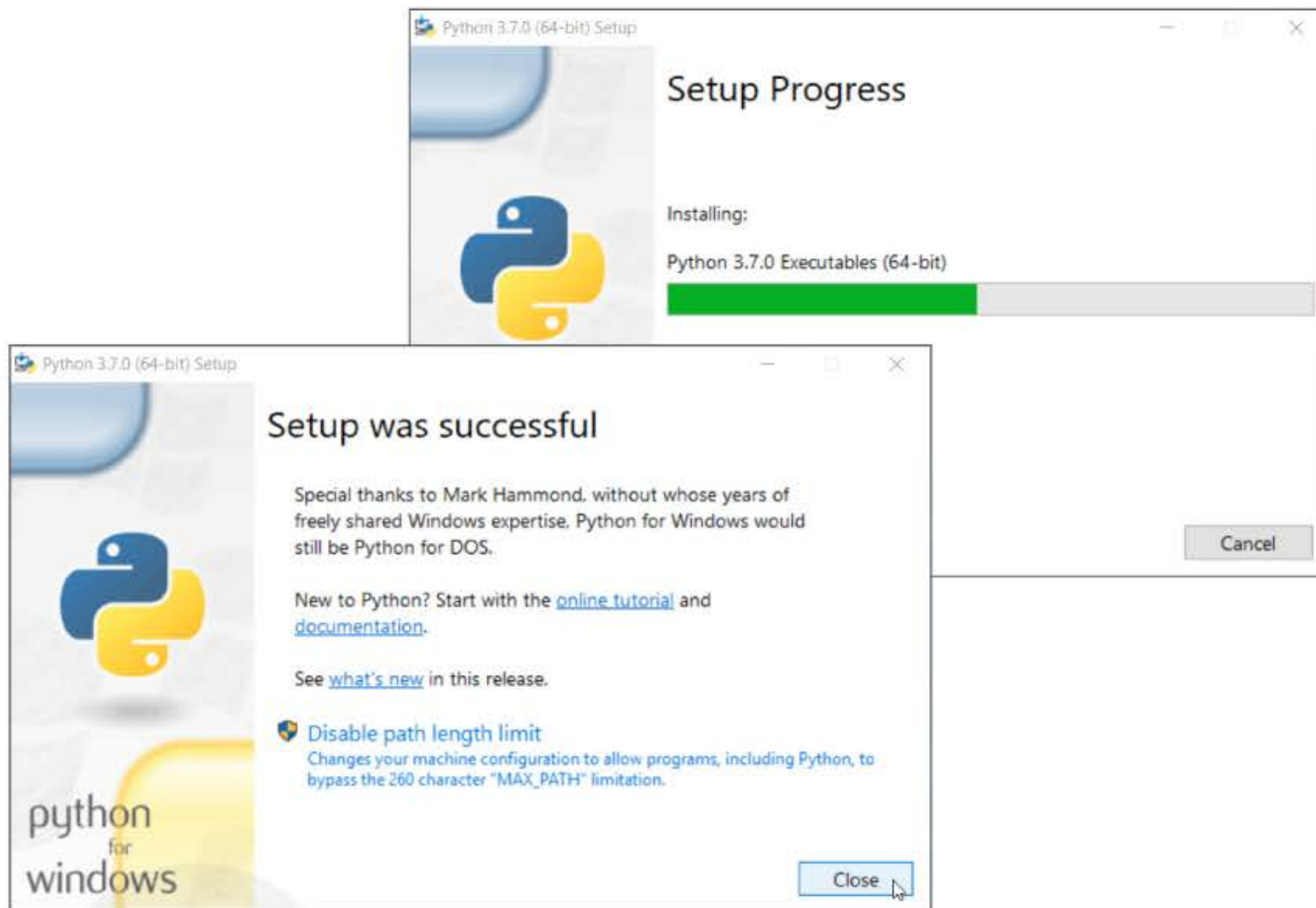
การติดตั้ง IDLE บนระบบ Windows มีขั้นตอนที่สำคัญคือ

1. เราสามารถดาวน์โหลด IDLE ได้จากเว็บไซต์ <https://www.python.org/downloads/windows> (ถ้าใช้ระบบอื่นก็คลิกที่ลิงก์ Download แล้วเลือกเป้าหมายที่ต้องการ)



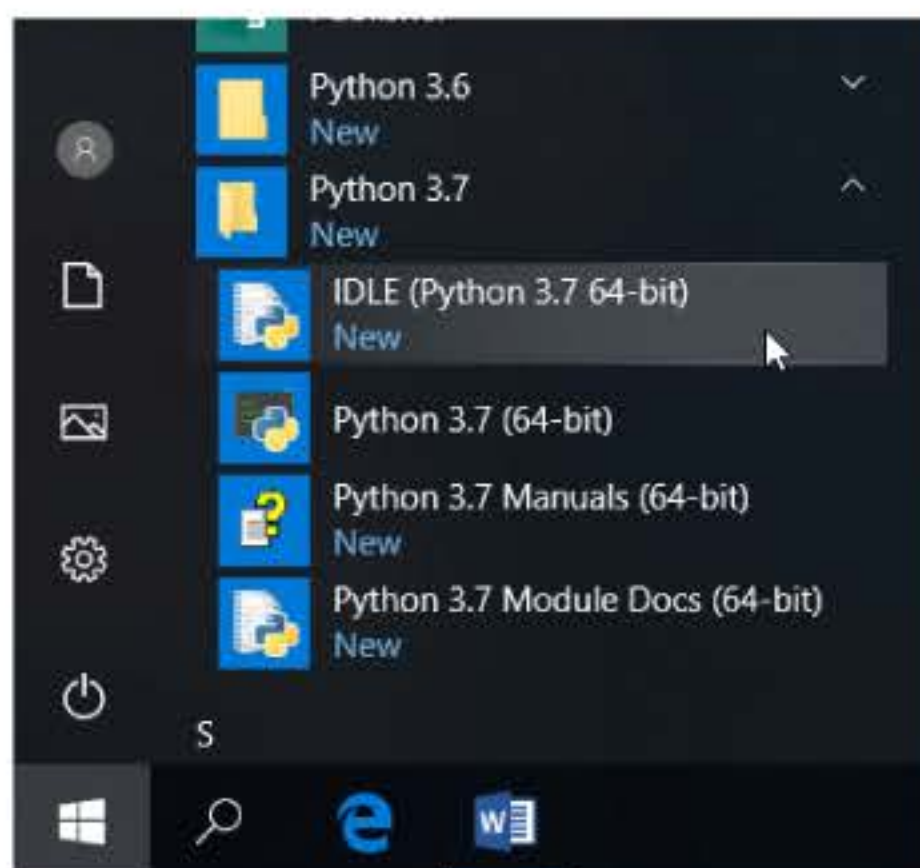
2. เมื่อดาวน์โหลดเสร็จ ก็ดับเบิลคลิกไฟล์ที่ได้มา ก็จะเข้าสู่กระบวนการติดตั้งเหมือนโปรแกรมทั่วไป โดยมีขั้นตอนที่สำคัญตามภาพต่อไปนี้





การใช้ IDLE ในเบื้องต้น

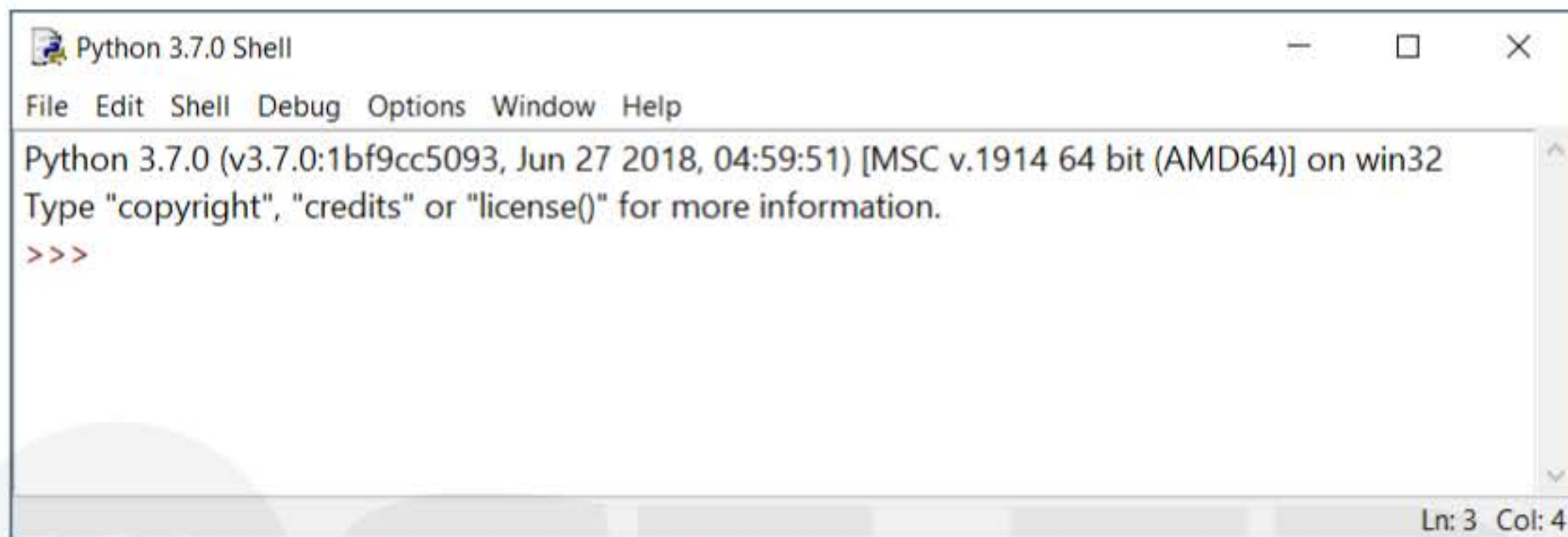
เนื่องจาก IDLE ไม่ได้สร้าง Shortcut บน Desktop เพื่อเข้าสู่โปรแกรมไว้ให้ ดังนั้น เมื่อจะเริ่มใช้งานสำหรับระบบวินโดวส์ ก็ให้คลิกที่ปุ่ม Start แล้วเลื่อนไปคลิกที่เมนู IDLE (Python x.x) ดังภาพถัดไป (ซ้าย) แต่ในคราวต่อไป หากเราต้องการเข้าสู่โปรแกรมให้รวดเร็วขึ้น เราก็อาจสร้าง Shortcut ของ IDLE โดยเปิดเมนูของปุ่ม Start ขึ้นมา จากนั้นคลิกที่เมนู IDLE ค้างเอาไว้ แล้วลากไปวางบนพื้นที่ว่างของ Desktop หลังจากปล่อยมือก็จัดตำแหน่งใหม่ตามต้องการ ดังภาพถัดไป (ขวา)



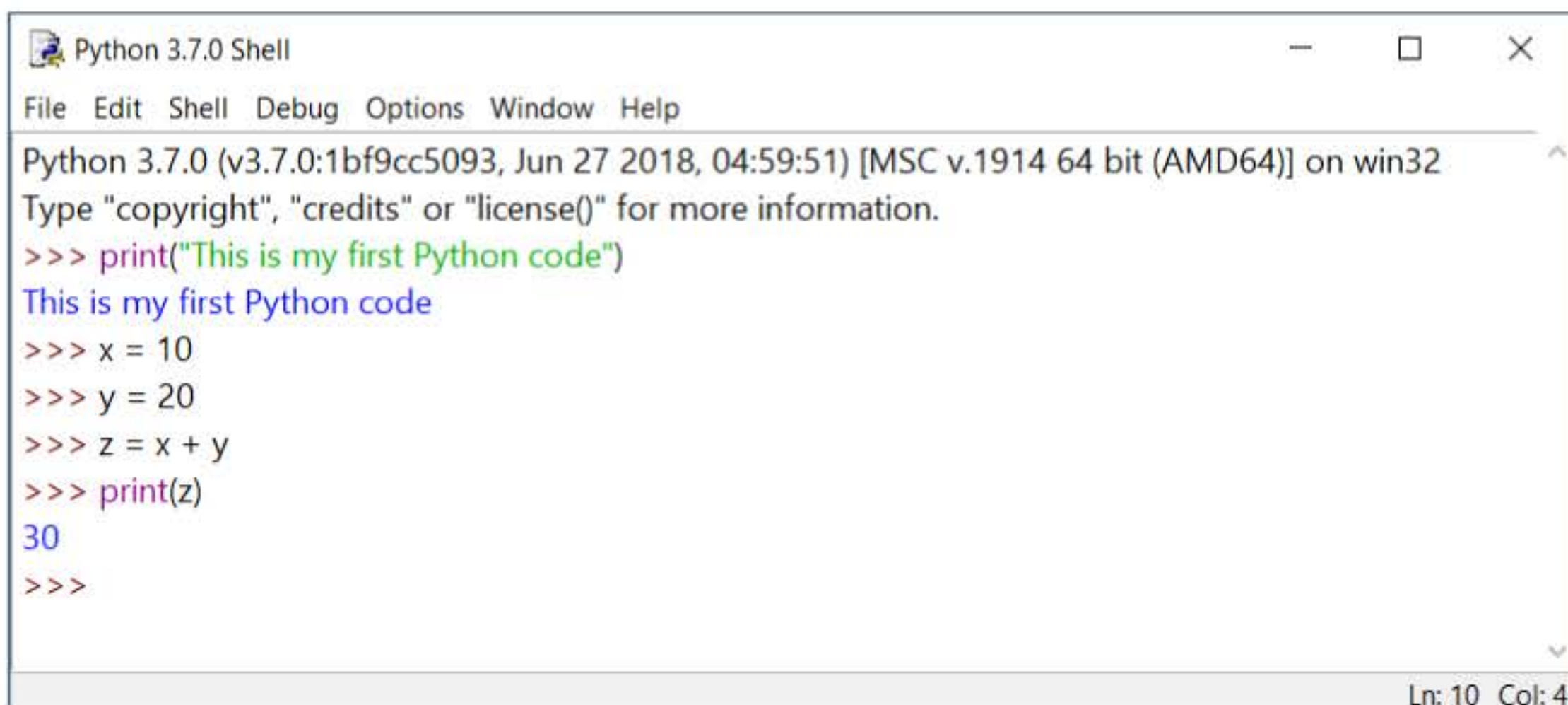
IDLE จะแบ่งหน้าจอออกเป็น 2 โหมดหลักๆ ดังนี้

- **Shell Window** โดยทั่วไป เราจะใช้งานใน 2 กรณีคือ
 - ใช้สำหรับเขียนโค้ดสั้นๆ เพื่อทดสอบการใช้งานบางอย่าง
 - ใช้แสดงผลลัพธ์ของการรันโค้ดที่เขียนใน Edit Window
- **Edit Window** ใช้ในการเขียนโค้ดคำสั่งที่ยาวๆ แบบต่อเนื่องกัน

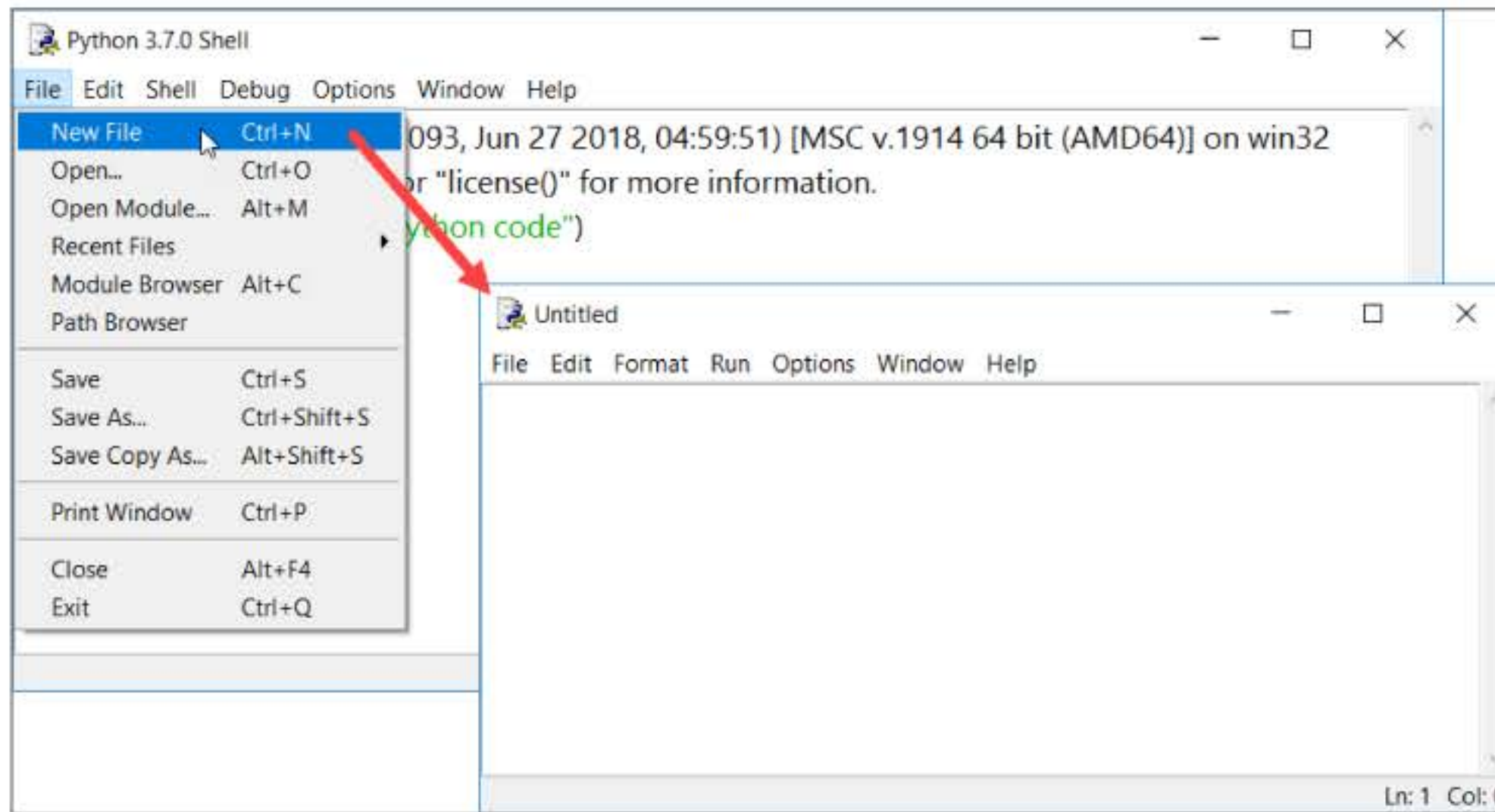
ตามปกติ เมื่อเราเปิดโปรแกรม IDLE ก็เข้าสู่ Shell Window ดังภาพ



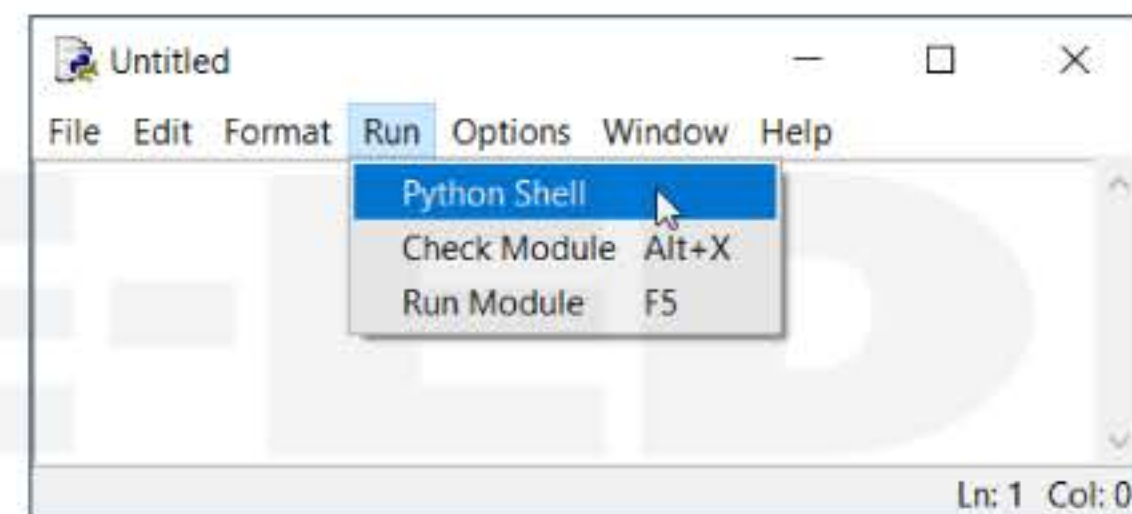
เราสามารถทดลองเขียนคำสั่งง่ายๆ เพื่อดูลักษณะใช้งานได้ดังภาพถัดไป โดยเมื่อเขียนคำสั่งในแต่ละบรรทัดเสร็จ ให้กดปุ่ม <Enter>



หากเราต้องการเปิด Edit Window จาก Shell Window ให้เลือกที่เมนู **File > New File**

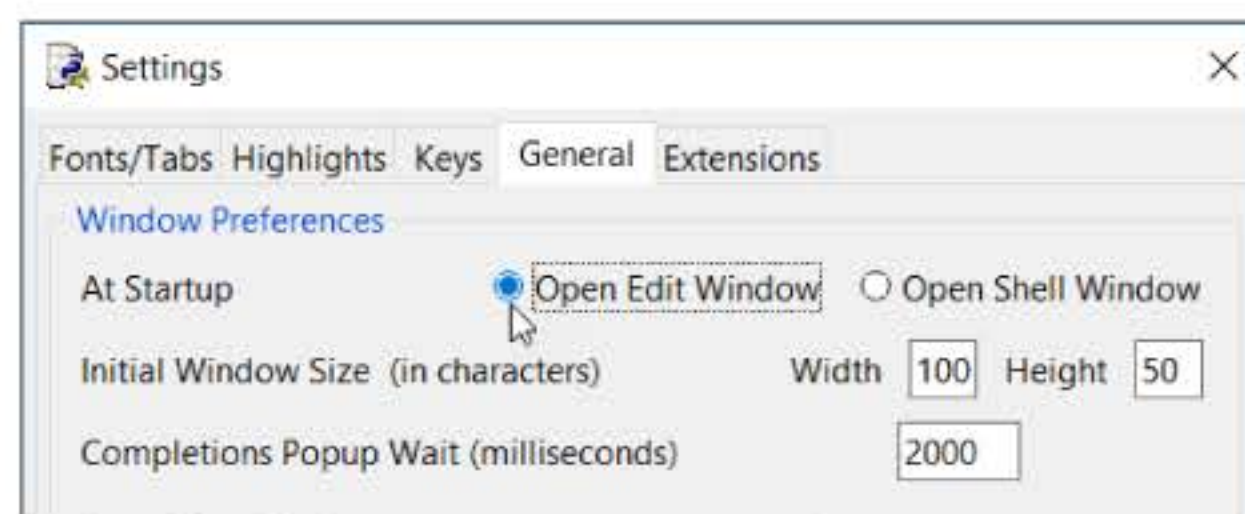
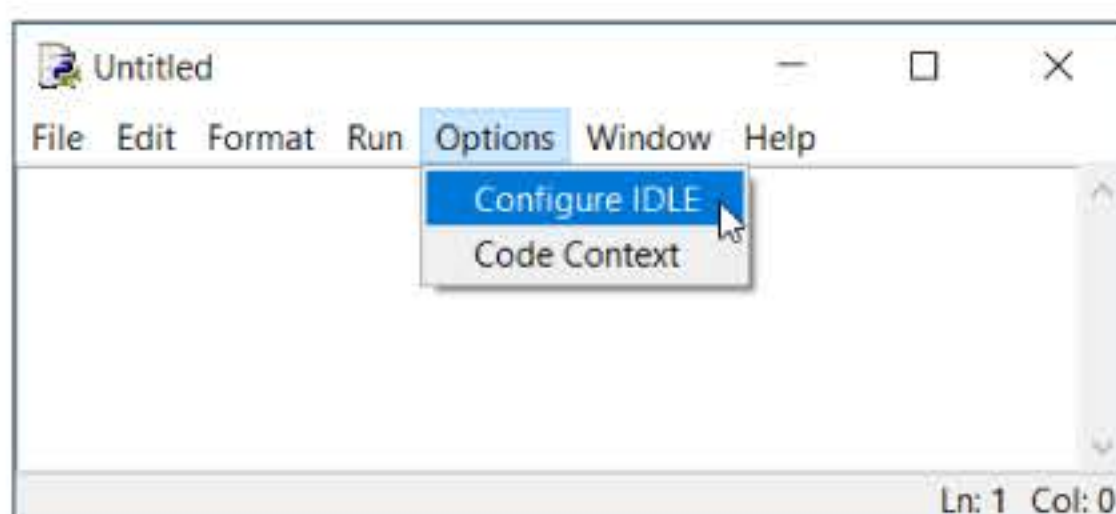


ถ้าในขณะนั้นเราเปิด Edit Window และหากปิด Shell Window ไปแล้ว แต่ต้องการเปิด Shell Window ขึ้นมาแสดงอีกครั้ง ให้เลือกที่เมนู **Run > Python Shell**



อย่างไรก็ตาม เนื่องจากการเขียนโค้ดโดยส่วนใหญ่ เรามักทำบน Edit Window แต่ตามปกติแล้ว เมื่อเปิดเข้าสู่ IDLE ก็จะมาที่ Shell Window ก่อน ซึ่งเราต้องเสียเวลาเปิดต่อมายัง Edit Window อีกที ทั้งนี้หากเราต้องการเข้าสู่ Edit Window โดยตรงเมื่อเปิด IDLE ก็สามารถตั้งค่าได้ดังนี้

1. ที่ Shell Window หรือ Edit Window ให้เลือกที่เมนู **Options > Configure IDLE**
2. เลือกที่แท็บ **General**
3. ที่บรรทัด **At Startup** ให้เลือก **Open Edit Window**

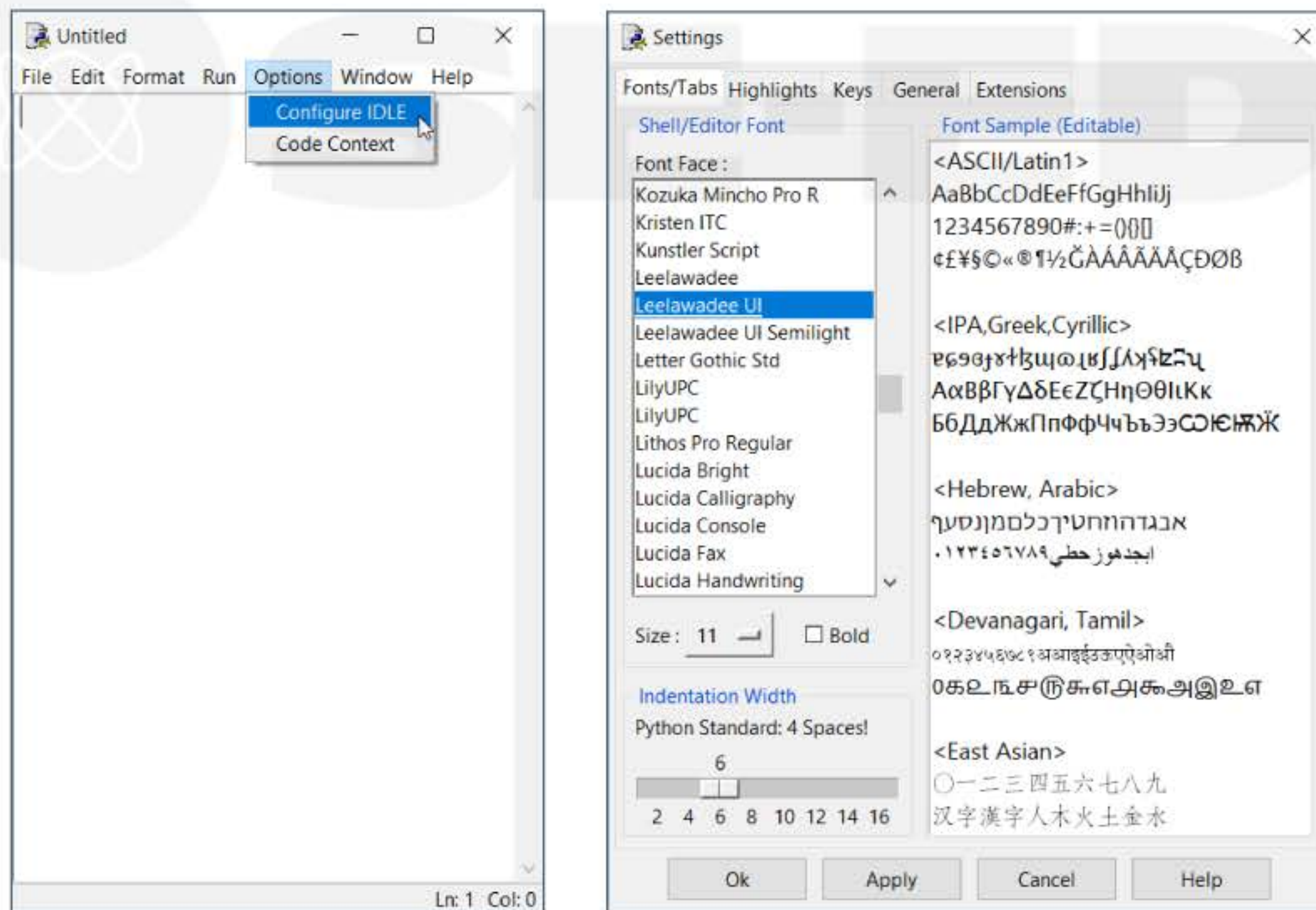


ถ้าเราต้องการเปลี่ยนไปเริ่มต้นที่ Shell Window เหมือนเดิม ก็ทำตามขั้นตอนเดียวกัน แต่เปลี่ยนไปเลือกที่ Open Shell Window แทน

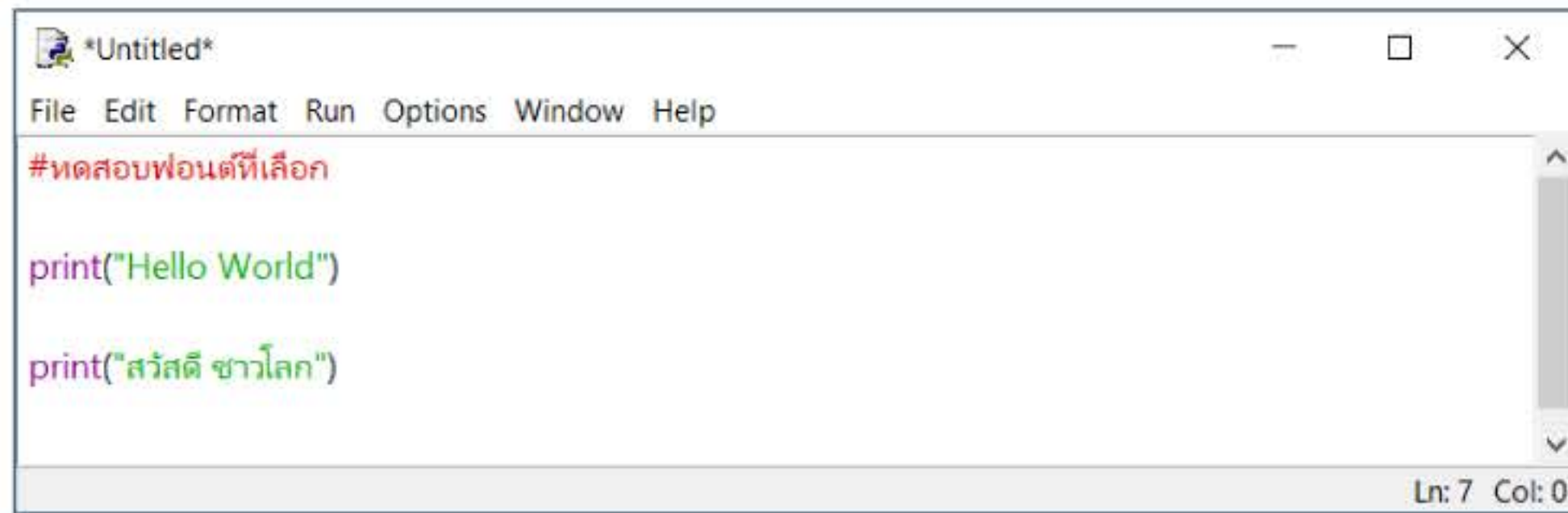
การกำหนดฟอนต์

ฟอนต์หรือตัวอักษรที่เป็นค่าปกติตามที่ IDLE กำหนดมาให้ล่วงหน้านั้น มักจะมีปัญหาในการแสดงผลอักขระภาษาไทยเป็นคันละชนิดกับอักขระภาษาอังกฤษ ซึ่งดูไม่กลมกลืนกันนัก ดังนั้น เราควรเปลี่ยนไปใช้ฟอนต์ที่รองรับภาษาไทยจะเหมาะสมกว่า โดยมีขั้นตอนดังนี้

1. ที่ Shell Window หรือ Edit Window ให้เลือกเมนู **Options > Configure IDLE**
2. คลิกที่แท็บ **Font/Tabs** ซึ่งรายการฟอนต์ที่มีให้เลือก ก็ขึ้นอยู่กับว่าเราติดตั้งฟอนต์ชนิดใดไว้ในเครื่องนั้นบ้าง
3. เลือกชนิดฟอนต์และขนาดที่เราต้องการ (ควรเลือกฟอนต์ไทย) สำหรับกรณีของผู้เขียนเลือกชนิด Leelawadee UI ขนาด 11



4. เมื่อเลือกเสร็จ ก็ลองเขียนข้อความลงไป ถ้ารูปแบบฟอนต์ยังไม่ถูกใจ ก็กลับไปเลือกใหม่

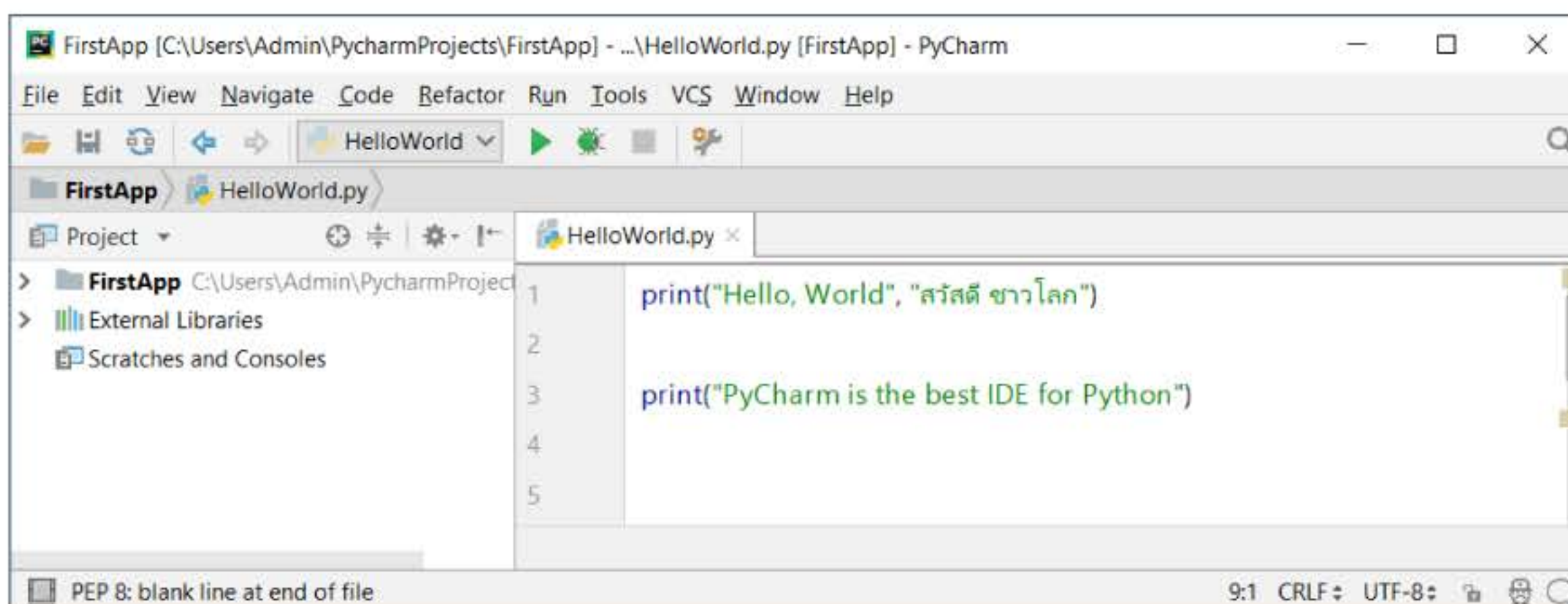


```
File Edit Format Run Options Window Help
#ทดสอบฟอนต์ที่เลือก
print("Hello World")
print("สวัสดี ชาวโลก")
Ln: 7 Col: 0
```

1-3: การติดตั้งและใช้งาน PyCharm

PyCharm เป็นโปรแกรมสำหรับการเขียนโค้ดภาษาไพธอนที่กำลังได้รับความนิยมอย่างมากจากนักพัฒนา ระดับมืออาชีพ เพราะมีลักษณะเด่นหลายประการ เช่น

- มีรุ่นที่แจกให้ใช้งานฟรีแบบไม่มีเงื่อนไข (Community Edition)
- มีรูปร่างหน้าตาที่สวยงาม น่าใช้งานมากกว่า
- มีเทคนิคการทำงานในหลายๆ กรณีที่ชาญฉลาดกว่าโปรแกรมตัวอื่นๆ อย่างชัดเจน
- เป็นโปรแกรมในกลุ่มเดียวกันกับ Android Studio สำหรับการสร้างแอปบนแอนดรอยด์ หรือ IntelliJ IDEA ที่ใช้สำหรับเขียนภาษา Java ดังนั้น จึงเหมาะอย่างยิ่งกับผู้ที่ศึกษาไพธอนเพื่อก้าวไปสู่การเขียนจาวา รวมทั้งการสร้างแอปบนระบบแอนดรอยด์ในอนาคต

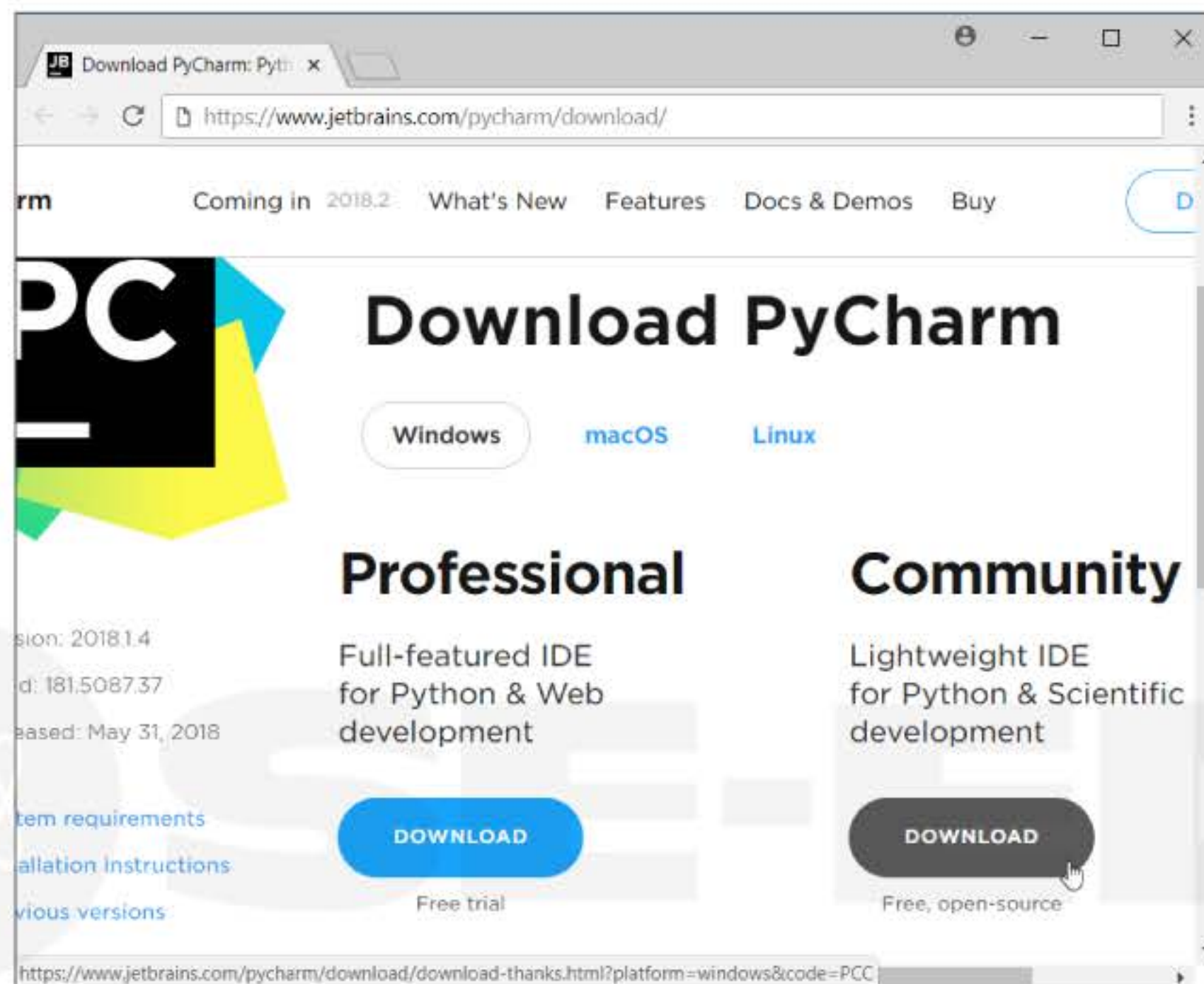


```
FirstApp [C:\Users\Admin\PycharmProjects\FirstApp] - ...\HelloWorld.py [FirstApp] - PyCharm
File Edit View Navigate Code Refactor Run Tools VCS Window Help
HelloWorld
FirstApp HelloWorld.py
Project
FirstApp C:\Users\Admin\PycharmProject
External Libraries
Scratches and Consoles
1 print("Hello, World", "สวัสดี ชาวโลก")
2
3 print("PyCharm is the best IDE for Python")
4
5
PEP 8: blank line at end of file 9:1 CRLF UTF-8
```

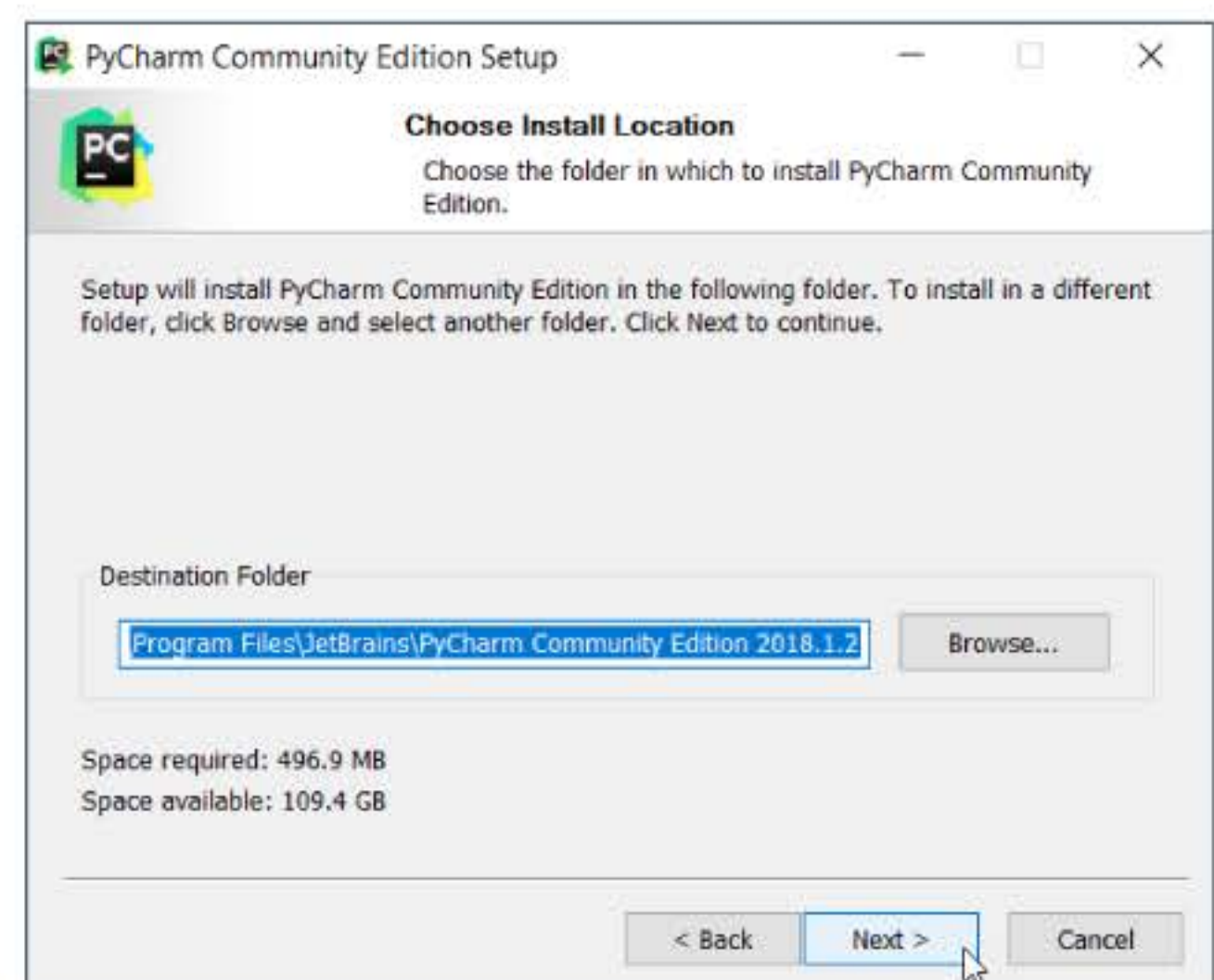
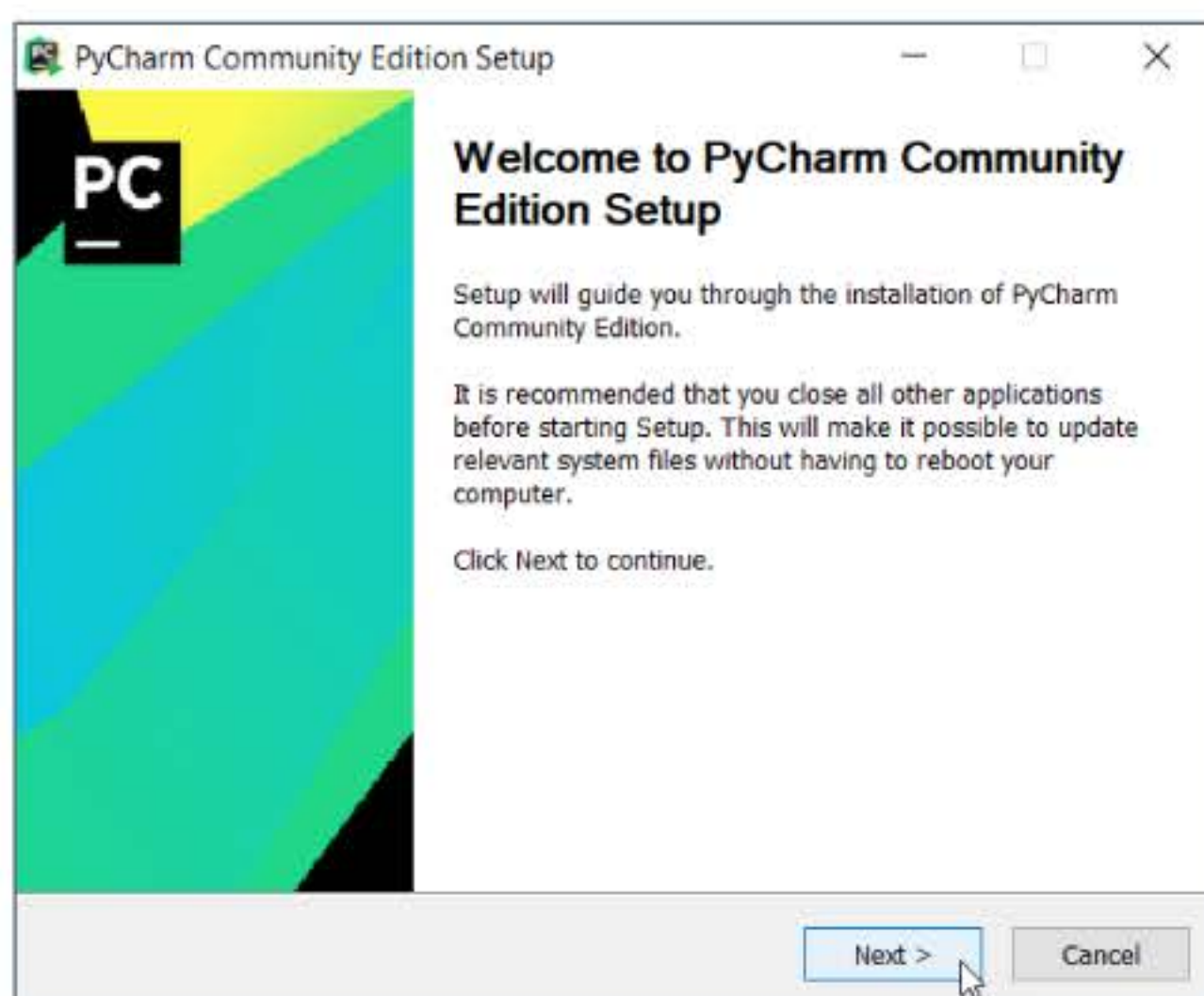

การติดตั้ง PyCharm

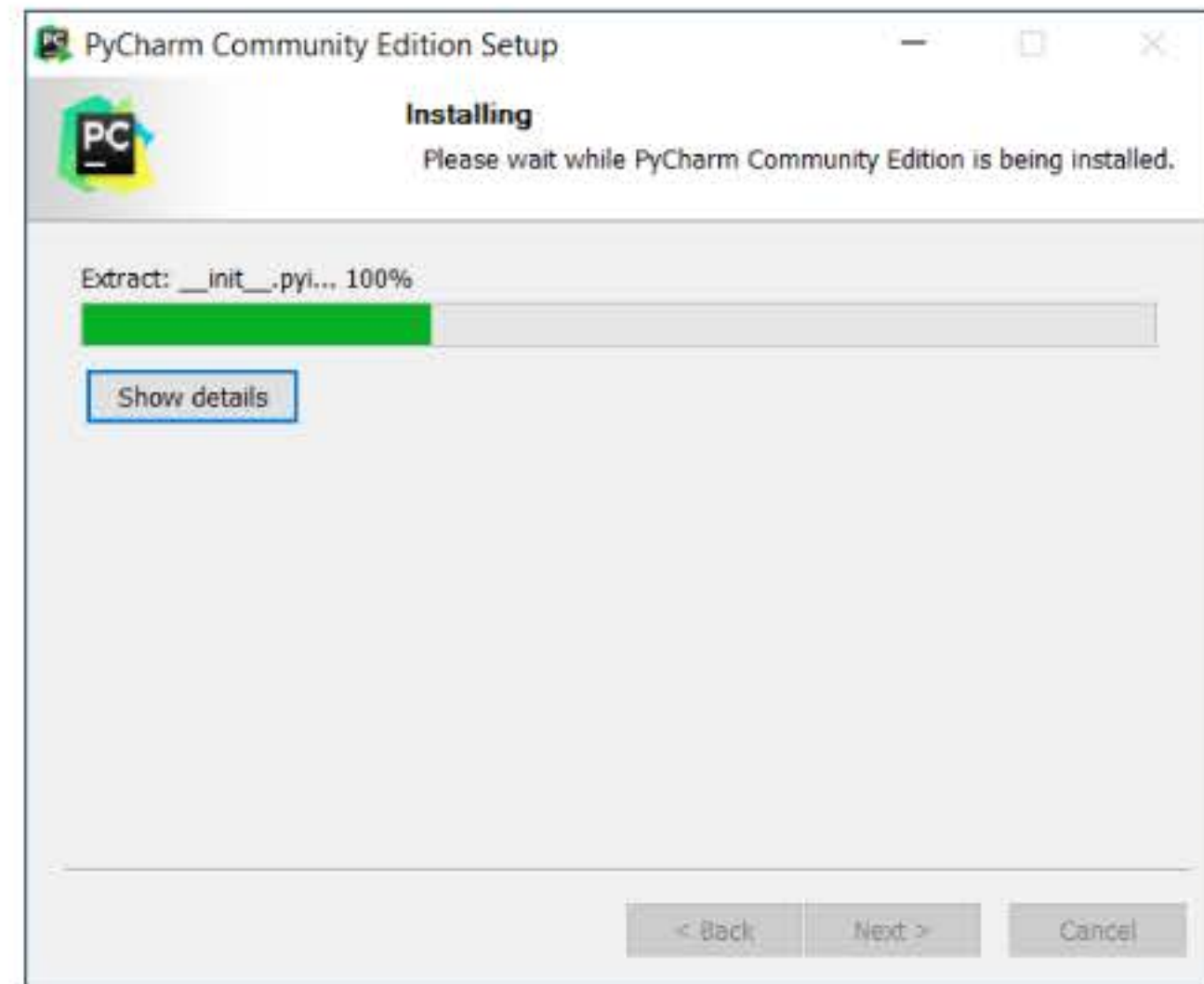
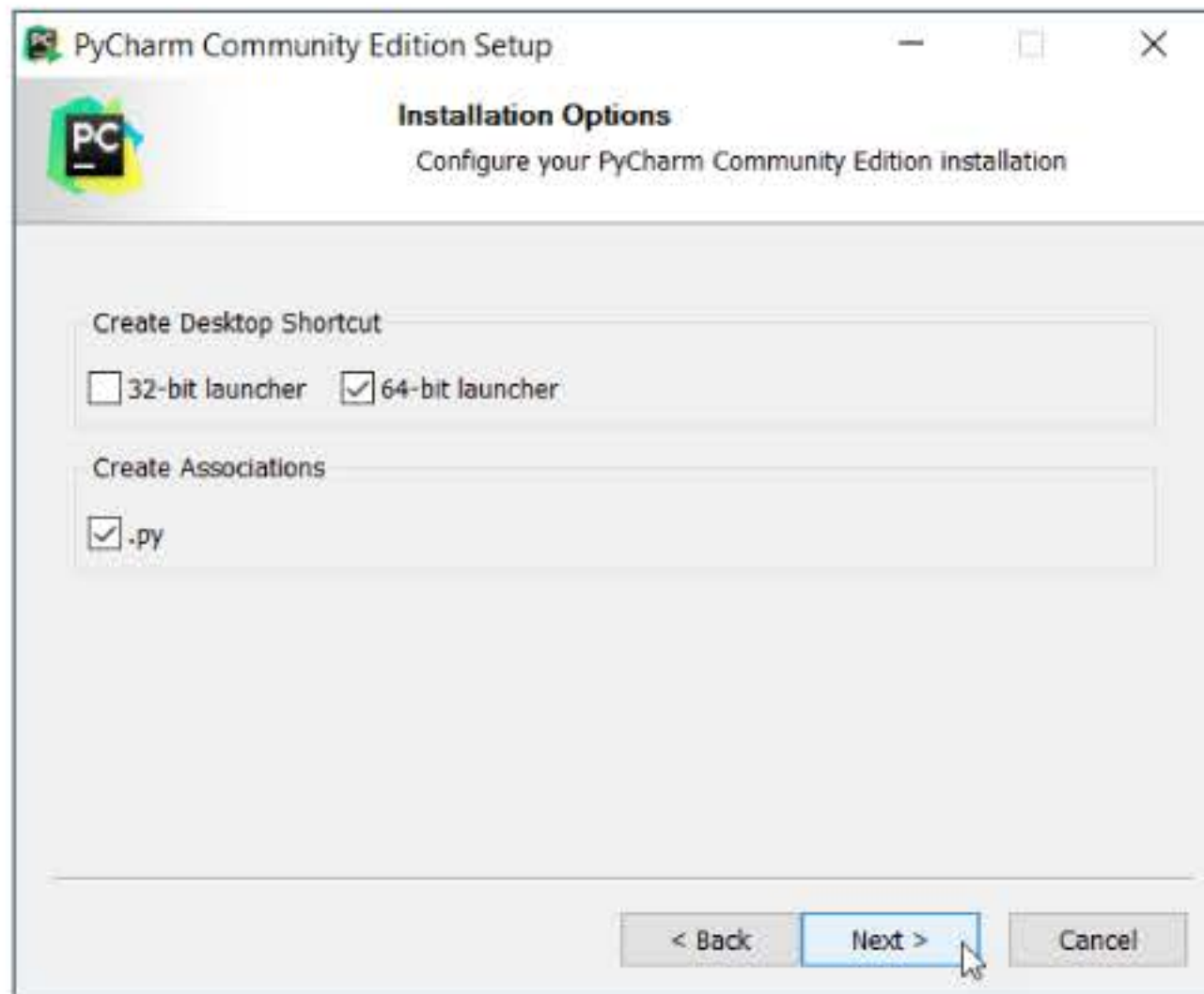
ถ้าเราจะนำ PyCharm มาเป็นเครื่องมือในการเขียนไพธอน ก็มีขั้นตอนการติดตั้งโดยสังเขปดังต่อไปนี้

1. สามารถดาวน์โหลด PyCharm ได้จากเว็บไซต์ <https://www.jetbrains.com/pycharm/download/> โดยหากต้องการใช้งานฟรีก็เลือกรุ่น **Community Edition**



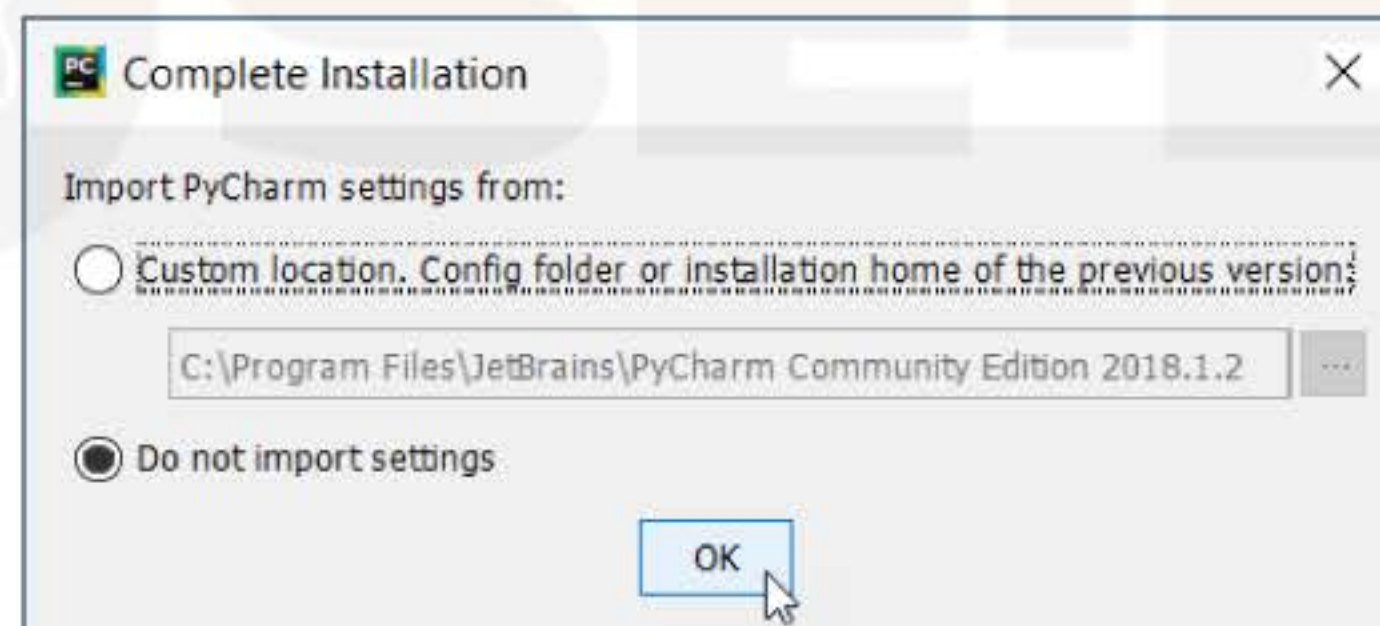
2. การติดตั้ง PyCharm ก็ทำเหมือนกับโปรแกรมอื่นๆ ไป โดยดับเบิลคลิกไฟล์ติดตั้งที่ดาวน์โหลดมา ซึ่งข้อกำหนดในบางขั้นตอน มีดังภาพต่อไปนี้



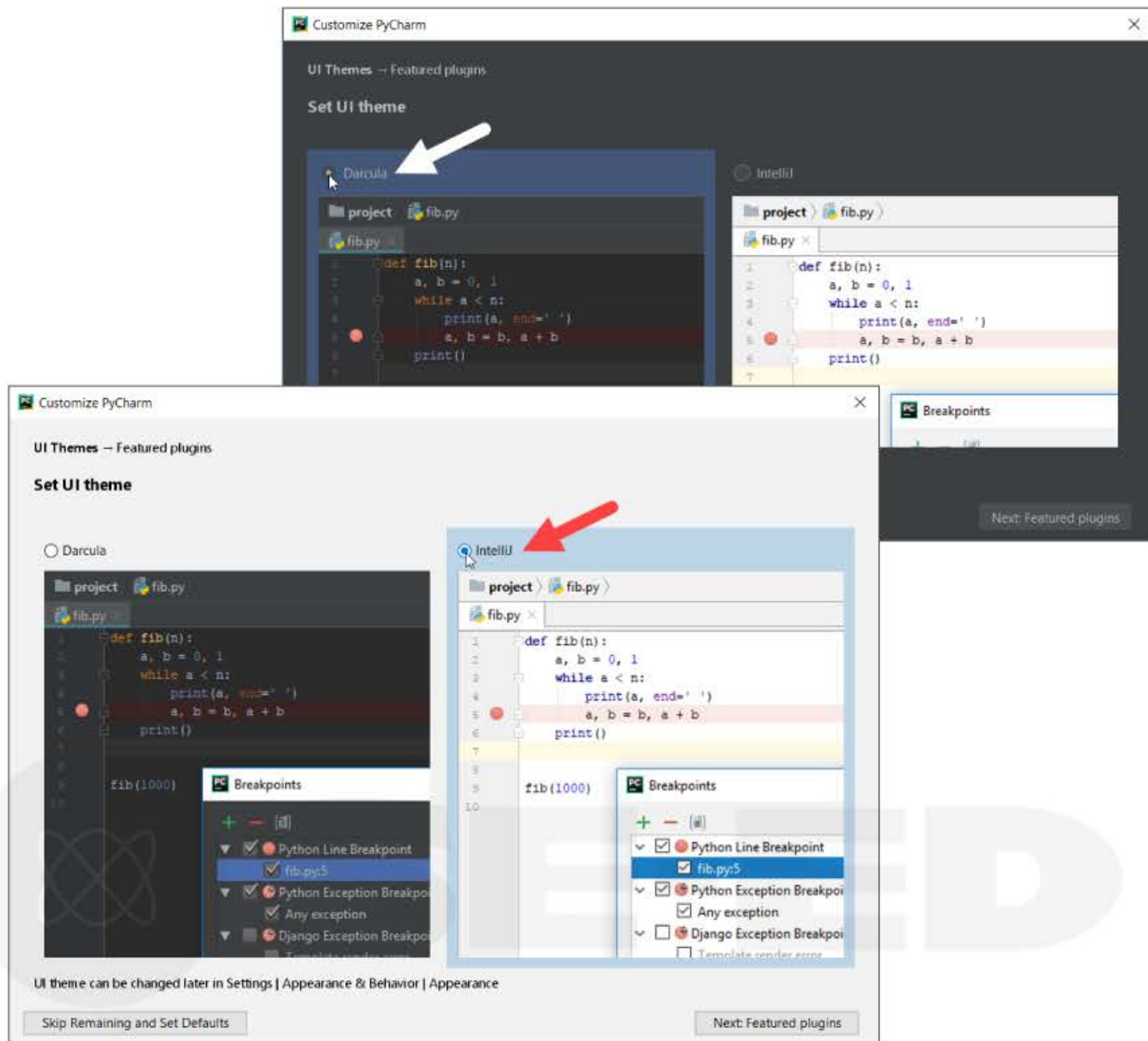


พื้นฐานการใช้ PyCharm

การเขียนโปรแกรมด้วย PyCharm จะต้องเริ่มที่การสร้างโปรเจกต์ (Project) ใหม่เสมอ โดยให้เปิดเข้าสู่ PyCharm แล้วทำตามขั้นตอนดังต่อไปนี้



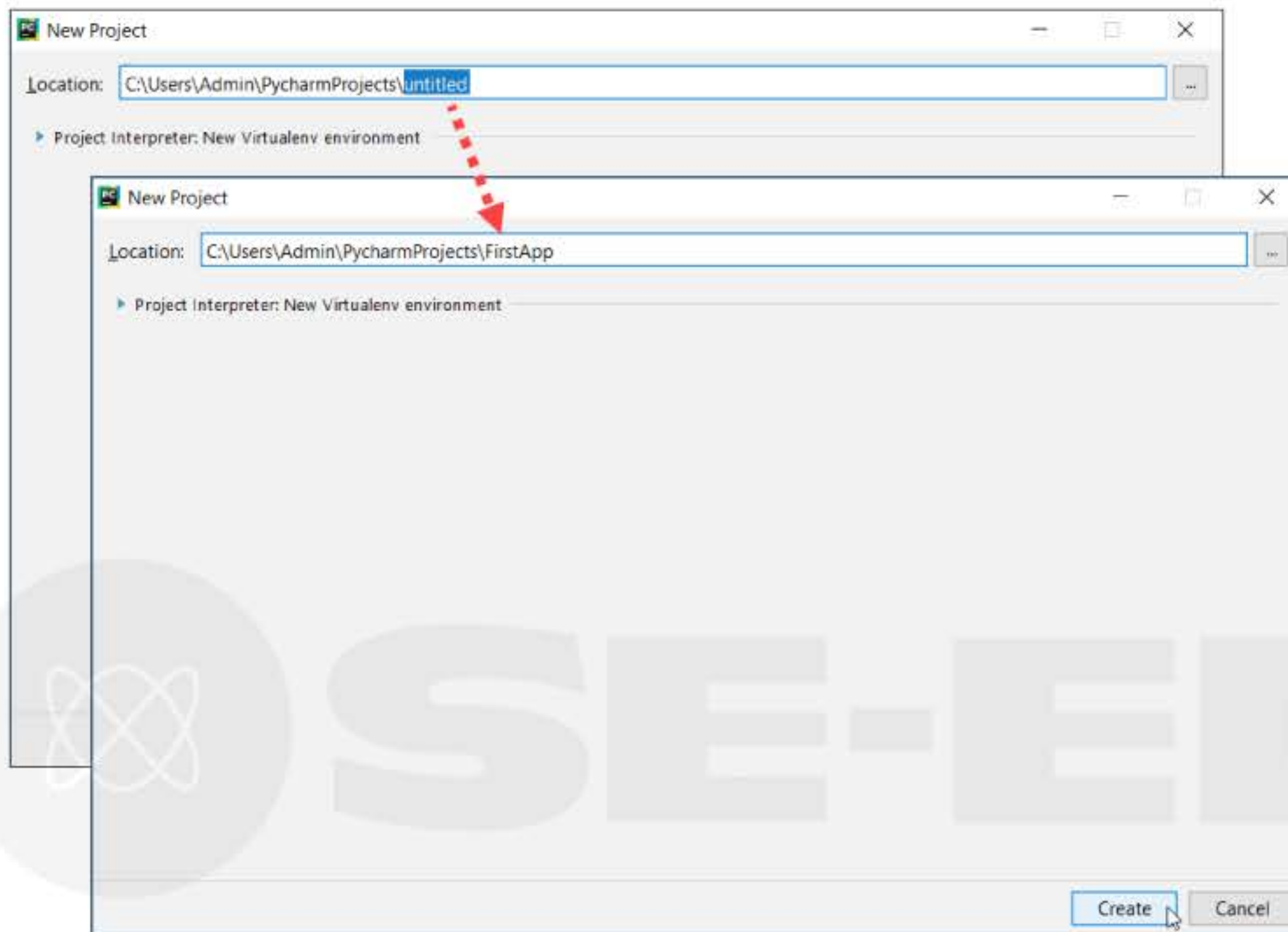
1. ถ้าเป็นการเปิดเข้าใช้งาน PyCharm ครั้งแรก จะมีหน้าจอขึ้นมาสอบถามว่าจะนำเข้าการตั้งค่าจากเวอร์ชันก่อนหรือไม่ ทั้งนี้หากเราไม่เคยติดตั้งมาก่อน ก็ให้เลือก Do not import settings แต่ถ้าเคยใช้เวอร์ชันเก่ามาแล้ว และยังไม่ได้ลบการตั้งค่า ก็นำมาใช้งานต่อโดยเลือกที่ Custom location... แล้วก็ไปเลือกตำแหน่งของไฟล์นั้น
2. ถ้าเป็นการเปิดเข้าใช้งาน PyCharm ครั้งแรก ก็จะมีหน้าจอให้เราเลือกธีม (Theme) หรือลักษณะสีสันทันของ PyCharm ซึ่งมี 2 แบบคือ **Darcula** (สีดำ) และ **IntelliJ** (สีเทาอ่อน) ให้เลือกแบบที่ต้องการ แล้วคลิกปุ่ม **Skip Remaining and Set Defaults** (ในหนังสือเล่มนี้ผู้เขียนใช้ธีม IntelliJ)



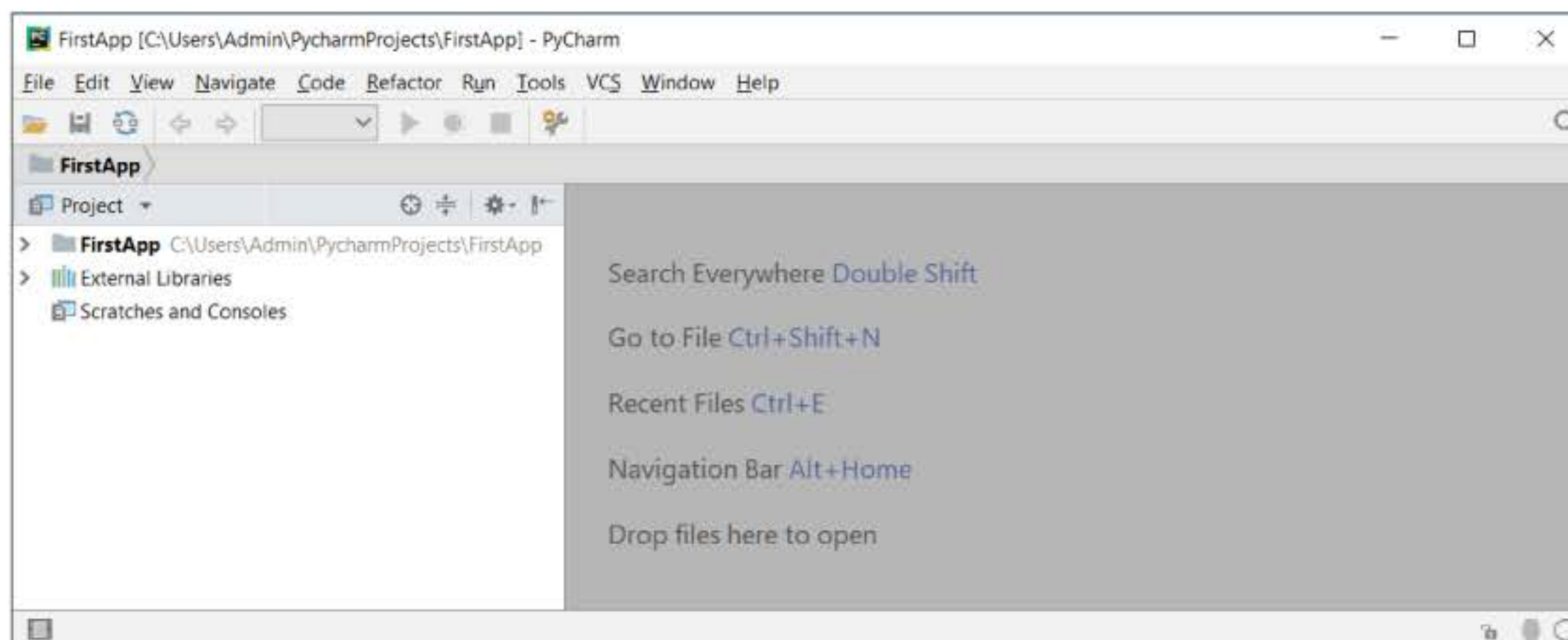
- เมื่อปรากฏหน้าจอ Welcome to PyCharm เลือกที่เมนู **Create New Project** (เมื่อเปิดเข้าสู่ PyCharm ครั้งต่อๆ ไปที่ไม่ใช่ครั้งแรก จะมาเริ่มที่ขั้นตอนนี้)



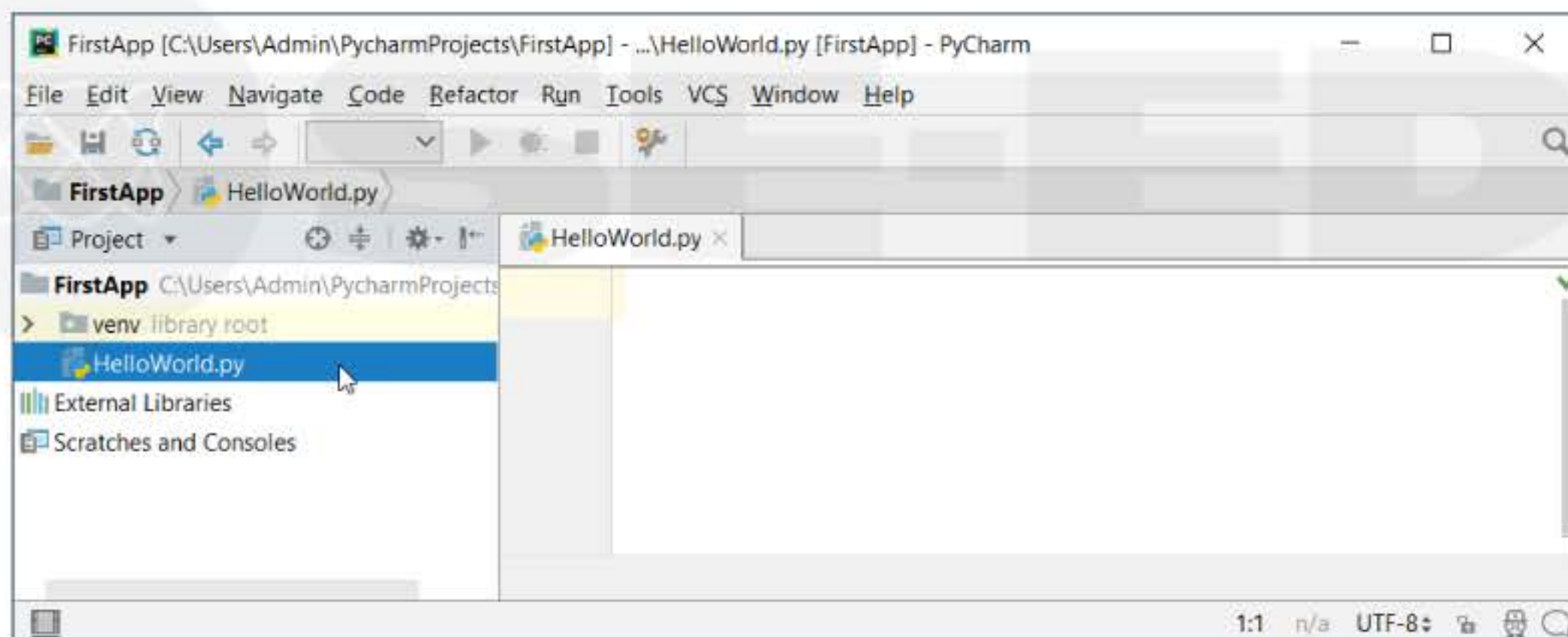
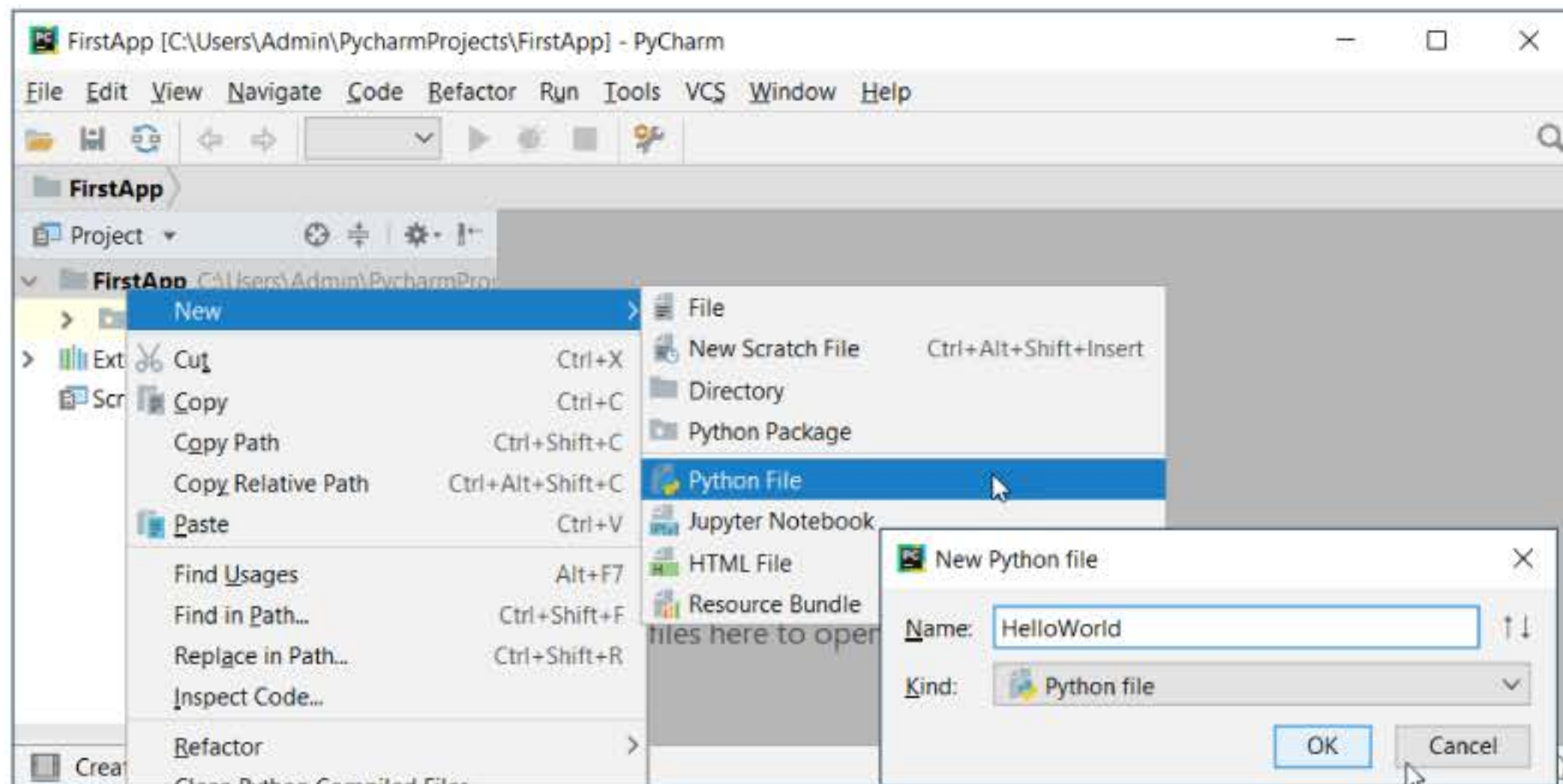
- ตามปกติโปรเจกต์ของ PyCharm จะถูกจัดเก็บไว้ที่ C:/Users/ชื่อผู้ใช้/PyCharmProjects ซึ่งเมื่อเราติดตั้ง PyCharm โฟลเดอร์ดังกล่าวจะถูกสร้างขึ้นมาให้โดยอัตโนมัติ แม้เราสามารถเปลี่ยนแปลงตำแหน่งการจัดเก็บเป็นอื่นได้ แต่ควรใช้ตำแหน่งตามที่โปรแกรมกำหนด เพื่อให้เป็นแนวทางเดียวกัน โดยในขั้นตอนนี้เราต้องตั้งชื่อโฟลเดอร์สำหรับจัดเก็บโปรเจกต์ที่กำลังจะสร้างขึ้น เช่น ในที่นี้กำหนดชื่อเป็น FirstApp



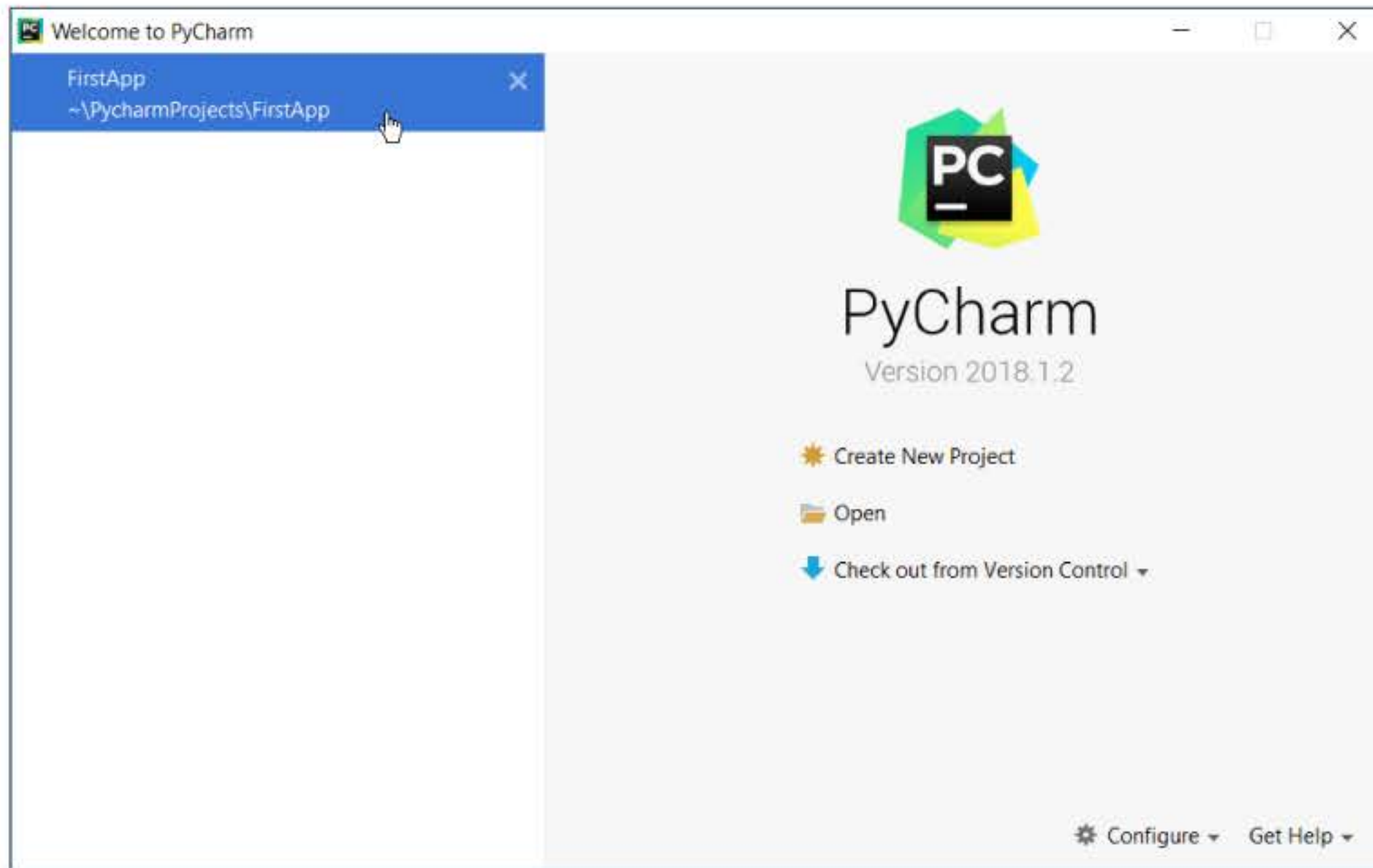
- เมื่อเข้าสู่หน้าจอหลักของ PyCharm ก็จะมีลักษณะดังนี้ (ถ้าทูลบาร์ไม่ปรากฏดังในภาพกดไป ให้เลือกที่เมนู **View > Toolbar**)



6. ถ้าเราจะเริ่มต้นการเขียนโปรแกรม ต้องสร้างไฟล์เพื่อจัดเก็บโค้ดไพธอนขึ้นมาก่อน โดยคลิกขวาที่โฟลเดอร์ชื่อโปรเจกต์ (ในที่นี้คือ FirstApp) > **New > Python File** จากนั้นกำหนดชื่อไฟล์ลงไป แล้วไฟล์ก็จะถูกสร้างขึ้น



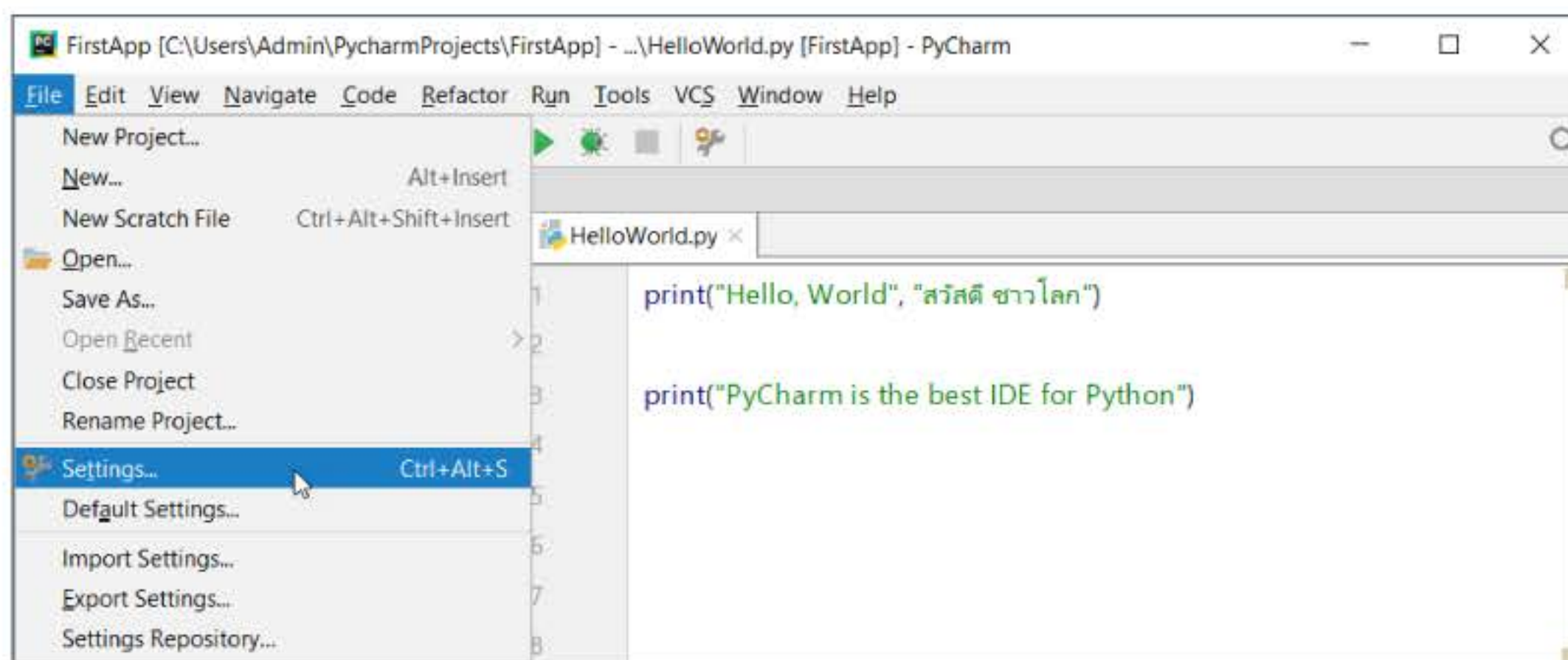
7. ถ้าต้องการปิดโปรเจกต์ ให้เลือกที่เมนู **File > Close Project** ซึ่งก็จะกลับเข้าสู่หน้าจอ Welcome แต่ถ้าต้องการปิดทั้งโปรเจกต์และปิดโปรแกรม PyCharm ด้วย ให้เลือกที่เมนู **File > Exit**
8. เมื่อเราเปิดเข้าสู่ PyCharm ในครั้งต่อไป จะปรากฏรายชื่อโปรเจกต์ที่เราเคยสร้างเอาไว้ที่ช่องด้านซ้ายมือของหน้าจอ Welcome แต่ถ้าไม่มีรายชื่อปรากฏในช่องนี้ ให้คลิกเมนู **Open** แล้วเลือกตำแหน่งที่เราจัดเก็บโปรเจกต์เอาไว้ ซึ่งถ้าใช้ตามค่าดีฟอลต์ก็จะอยู่ที่ **C:\Users\ชื่อผู้ใช้\PyCharmProjects**



การกำหนดฟอนต์และธีม

ฟอนต์ของโค้ดและส่วนผลลัพธ์ที่ PyCharm กำหนดมาให้ล่วงหน้า อาจเกิดปัญหาในการแสดงอักขระภาษาไทย หรือมีรูปแบบที่ไม่ค่อยถูกใจเรานัก ซึ่งก็สามารถเปลี่ยนแปลงใหม่ได้ดังนี้

1. เลือกที่เมนู **File > Settings...**

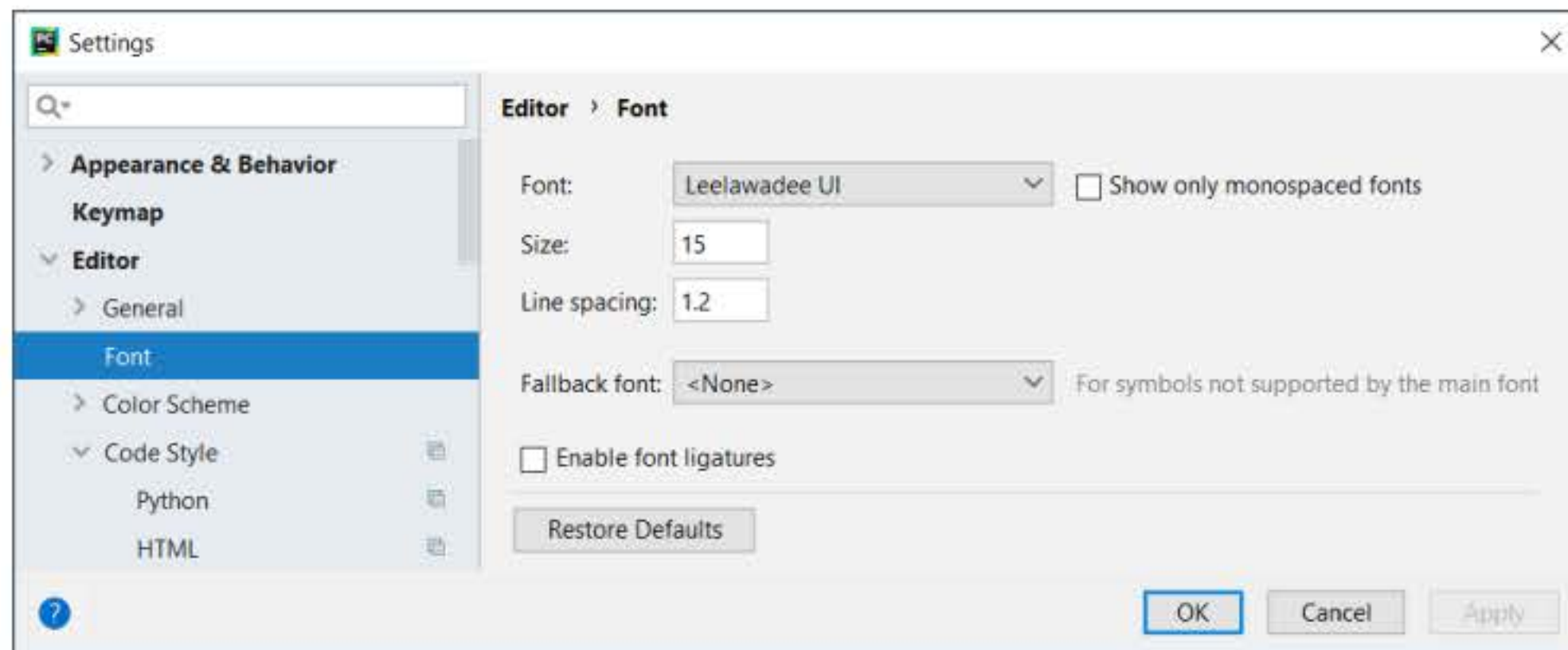


2. ในหน้าจอถัดไป ให้ทำตามขั้นตอนดังนี้

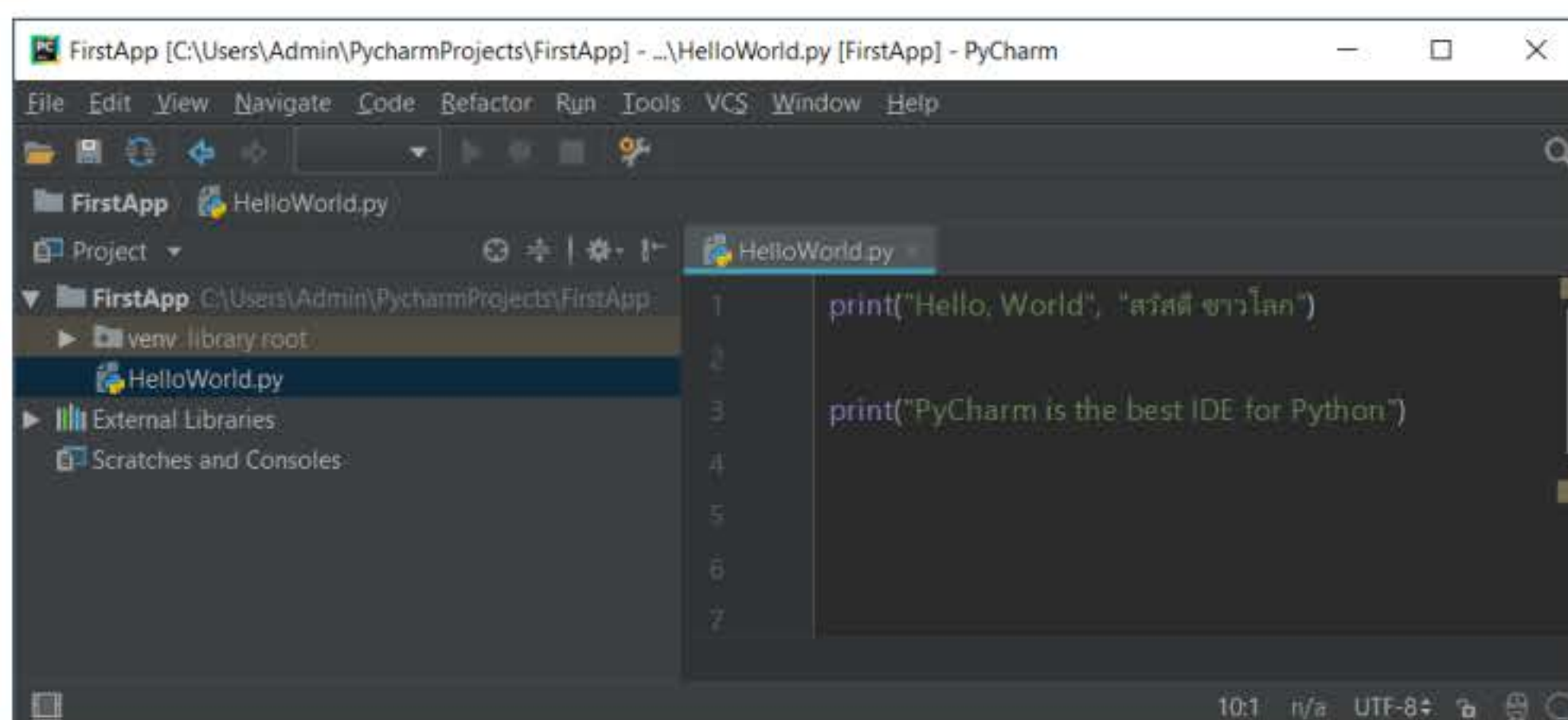
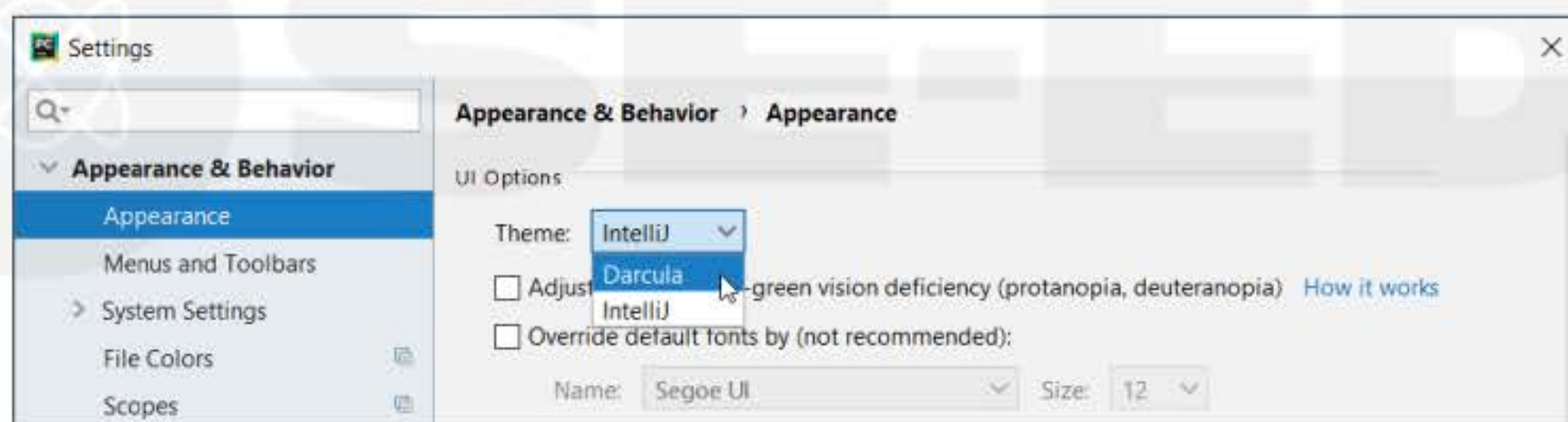
1) เลือกที่ **Editor > Font**

2) เอาเครื่องหมาย **Show only monospaced fonts** ออก

- 3) เลือกชนิดฟอนต์และกำหนดขนาดตามต้องการ โดยในกรณีของผู้เขียน ได้เลือกชนิด Leelawadee UI ขนาด 15 และ Line spacing: 1.2



3. ตามปกติ PyCharm จะมีธีม (Theme) ให้เลือก 2 แบบคือ Darcula (สีดำน) และ IntelliJ (สีเทาอ่อน) หากเราต้องการเปลี่ยนธีม ก็ให้เลือกที่เมนู **File > Settings...** ตามเดิม แล้วในหน้าจอถัดไป ให้เลือก **Appearance & Behavior > Appearance** จากนั้นเลือก Theme ดังภาพ



สำหรับในบทนี้ เราจะเรียนรู้แค่การเตรียมเครื่องมือสำหรับเขียนโค้ดดังที่กล่าวมาแล้ว ซึ่งผู้อ่านควรฝึกใช้งานให้เกิดความคุ้นเคยในระดับหนึ่งก่อน มิฉะนั้นอาจเกิดปัญหาต่อการใช้งานจริงในบทต่อๆ ไปได้

การเขียนโปรแกรมด้วย Python สำหรับผู้เริ่มต้น

ไพธอน (Python) เป็นหนึ่งในภาษาคอมพิวเตอร์ที่มีจำนวนผู้ใช้งานสูงสุดในปัจจุบัน เนื่องจากลักษณะโครงสร้างที่ไม่ซับซ้อน เรียนรู้ง่าย และความสามารถอันโดดเด่นอีกหลายๆ ด้าน จึงเหมาะอย่างยิ่งสำหรับผู้เริ่มต้นศึกษาด้านการเขียนโปรแกรม โดยหนังสือเล่มนี้ได้รวบรวมเนื้อหาที่จำเป็นต้องรู้ทั้งหมดในขั้นพื้นฐาน พร้อมทั้งตัวอย่างและแบบฝึกหัดท้ายบทที่หลากหลาย รวมถึงเวิร์คช็อปอีกบางส่วน โดยใช้เครื่องมือยอดนิยมอย่าง Python IDLE และ PyCharm ซึ่งสามารถเรียนรู้ด้วยตนเอง และเป็นแนวทางสำหรับศึกษาเพิ่มเติมจากแหล่งข้อมูลอื่นๆ ต่อไปในอนาคตได้

เนื้อหาในหนังสือเล่มนี้ประกอบด้วย :

บทที่ 1 : Python และเครื่องมือเขียนโค้ด
บทที่ 2 : พื้นฐานการเขียนภาษา Python
บทที่ 3 : โอเปอเรเตอร์และข้อมูลตัวเลข
บทที่ 4 : การเปรียบเทียบและกำหนดเงื่อนไข
บทที่ 5 : การทำซ้ำแบบวนรอบ
บทที่ 6 : ข้อมูลแบบลำดับรายการ
บทที่ 7 : การใช้สตริงเพิ่มเติมและวันเวลา
บทที่ 8 : ฟังก์ชันและโมดูล

บทที่ 9 : การตรวจสอบข้อผิดพลาด
บทที่ 10 : การอ่านและเขียนไฟล์
บทที่ 11 : การสร้างคลาสในเบื้องต้น
บทที่ 12 : การสร้าง GUI ด้วย Tkinter (ตอนที่ 1)
บทที่ 13 : การสร้าง GUI ด้วย Tkinter (ตอนที่ 2)
บทที่ 14 : จัดการฐานข้อมูลด้วย Python
บทที่ 15 : เวิร์คช็อป 1: เกม Craps
บทที่ 16 : เวิร์คช็อป 2: Address Book



www.se-ed.com



sbc.fans



คอมพิวเตอร์/การเขียนโปรแกรม