## Python การใช้อาเรย์ (Array)

## อาเรย์(array) คือการเก็บข้อมูลเป็นชุดของข้อมูลที่มีลำดับของข้อมูล โดยจะประกอบด้วยข้อมูลหลายๆข้อมูลและมีชนิดของข้อมูลแบบเดียวกัน เช่น อาเรย์ของเลขจำนวนเต็ม(int), อาเรย์ของเลขทศนิยม(float), อาเรย์ของข้อความ(string) ซึ่งโดยปกติจะมีความแตกต่างจากชุดข้อมูลชนิดลิสต์ที่สามารถเก็บข้อมูลได้หลายชนิดภายในลิสต์เดียวกัน

ใน python นั้นโดยปกติแล้วจะไม่มีชุดข้อมูลชนิดอาเรย์(array)มาให้ แต่หากต้องการใช้งานในรูปแบบอาเรย์ก็จะใช้การเก็บชุดข้อมูลชนิดลิสต์(list) แต่จะทำการเก็บข้อมูลเพียงชนิดเดียวภายในลิสต์ดังกล่าวแทน เพื่อนำไปใช้งานในรูปแบบของอาเรย์(array)

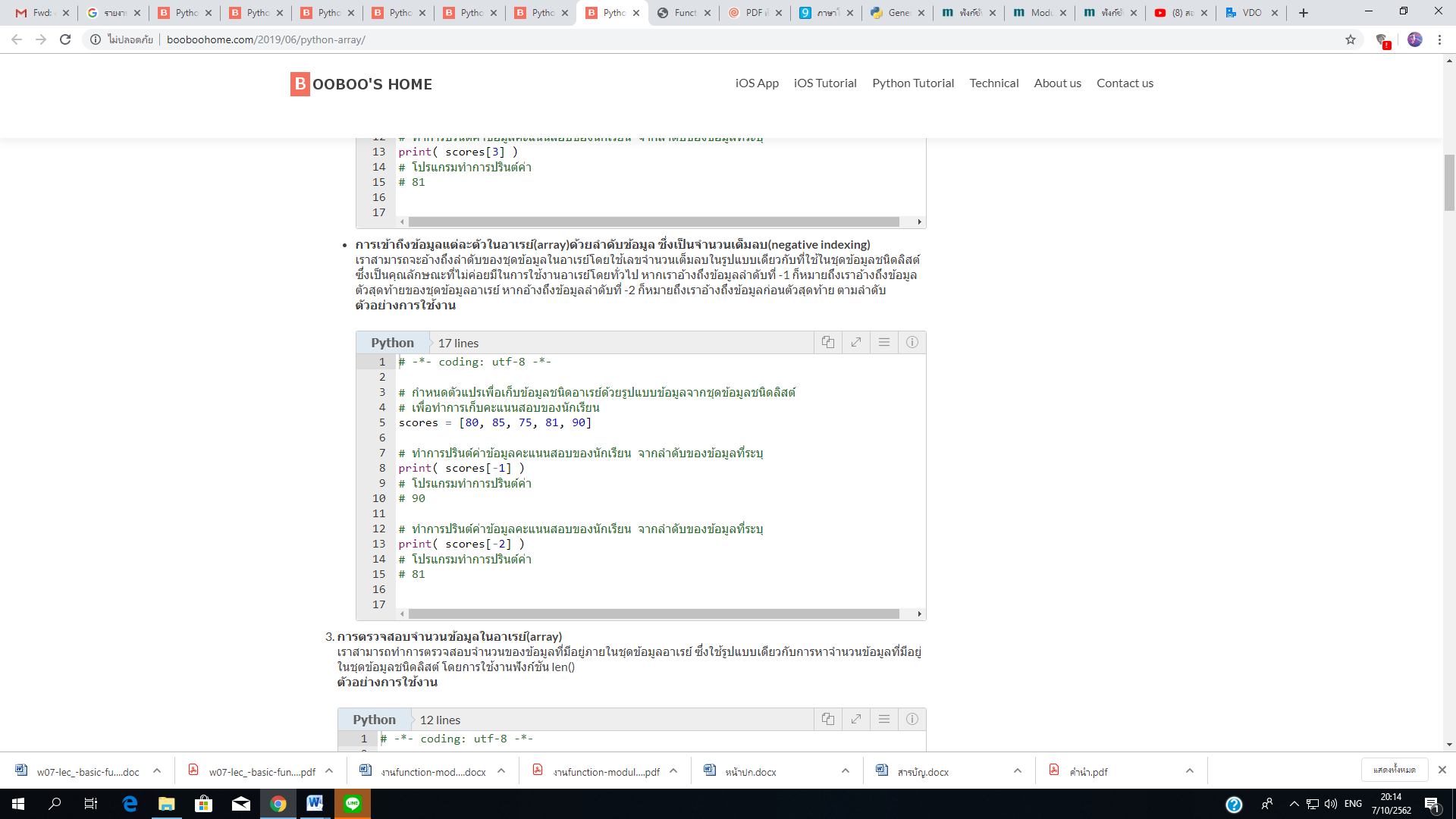
ในกรณีที่จำเป็นต้องใช้งานรูปแบบการเก็บข้อมูลชนิดอาเรย์โดยเฉพาะและมีประสิทธิภาพสูง ก็จำเป็นต้องใช้ไลบรารี่(library)เพิ่มเติม เช่น ชุดข้อมูลชนิดอาเรย์จากไลบรารี่ numpy แทน ซึ่งหากต้องการดำเนินการทางคณิตศาสตร์เฉพาะเจาะจงกับชุดข้อมูลชนิดอาเรย์นั้น การเลือกใช้งานอาเรย์จากไลบรารี่ numpy แทนการใช้การเก็บข้อมูลในรูปแบบลิสต์จะทำให้มีประสิทธิภาพมากกว่า

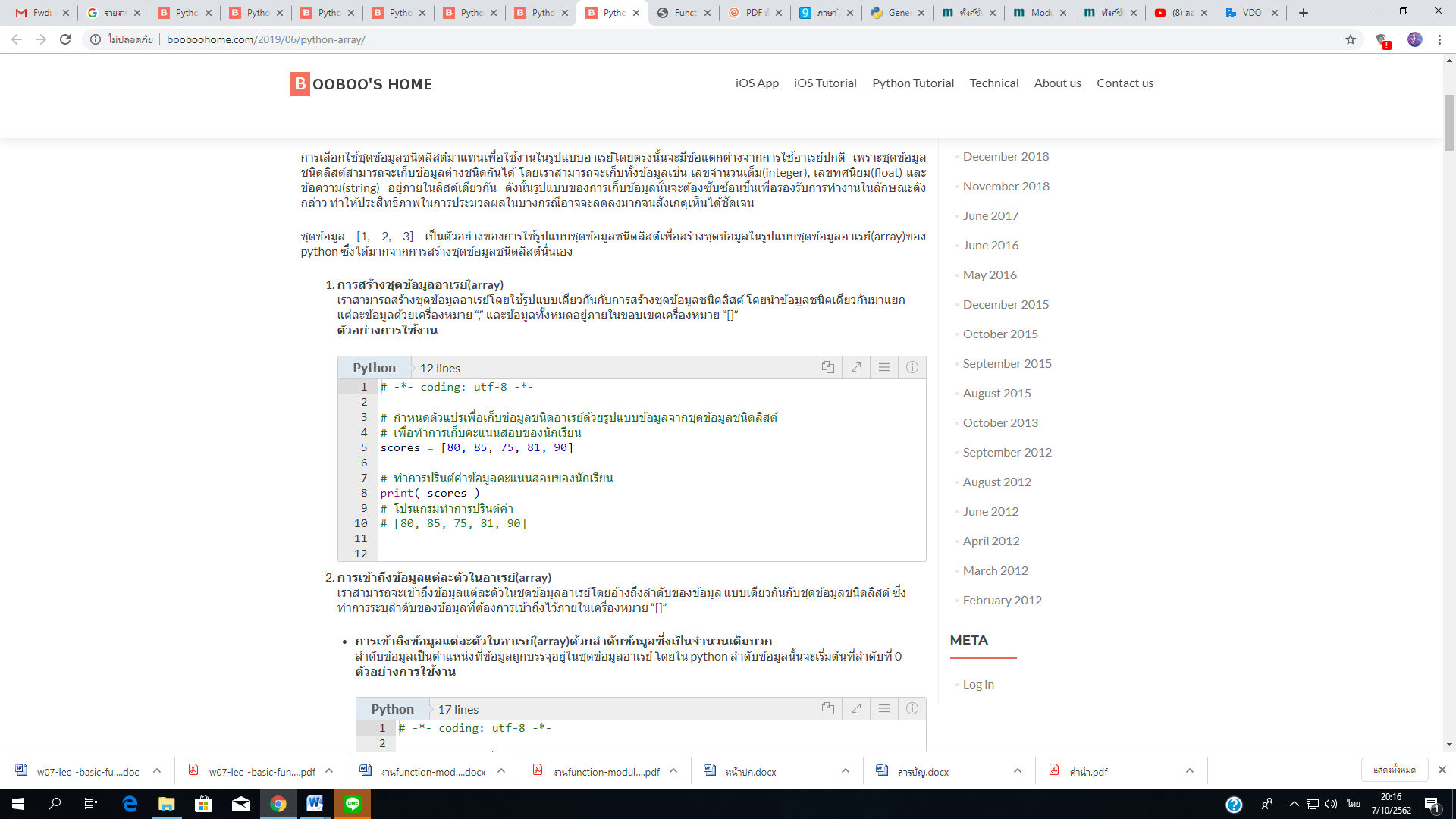
การเลือกใช้ชุดข้อมูลชนิดลิสต์มาแทนเพื่อใช้งานในรูปแบบอาเรย์โดยตรงนั้นจะมีข้อแตกต่างจากการใช้อาเรย์ปกติ เพราะชุดข้อมูลชนิดลิสต์สามารถจะเก็บข้อมูลต่างชนิดกันได้ โดยเราสามารถจะเก็บทั้งข้อมูลเช่น เลขจำนวนเต็ม(integer), เลขทศนิยม(float) และ ข้อความ(string) อยู่ภายในลิสต์เดียวกัน ดังนั้นรูปแบบของการเก็บข้อมูลนั้นจะต้องซับซ้อนขึ้นเพื่อรองรับการทำงานในลักษณะดังกล่าว ทำให้ประสิทธิภาพในการประมวลผลในบางกรณีอาจจะลดลงมากจนสังเกตุเห็นได้ชัดเจน

ชุดข้อมูล [1, 2, 3] เป็นตัวอย่างของการใช้รูปแบบชุดข้อมูลชนิดลิสต์เพื่อสร้างชุดข้อมูลในรูปแบบชุดข้อมูลอาเรย์(array)ของ python ซึ่งได้มากจากการสร้างชุดข้อมูลชนิดลิสต์นั่นเอง

**1. การสร้างชุดข้อมูลอาเรย์(array)**

เราสามารถสร้างชุดข้อมูลอาเรย์โดยใช้รูปแบบเดียวกันกับการสร้างชุดข้อมูลชนิดลิสต์ โดยนำข้อมูลชนิดเดียวกันมาแยกแต่ละข้อมูลด้วยเครื่องหมาย “,” และข้อมูลทั้งหมดอยู่ภายในขอบเขตเครื่องหมาย “[ ]”

**ตัวอย่างการใช้งาน** 



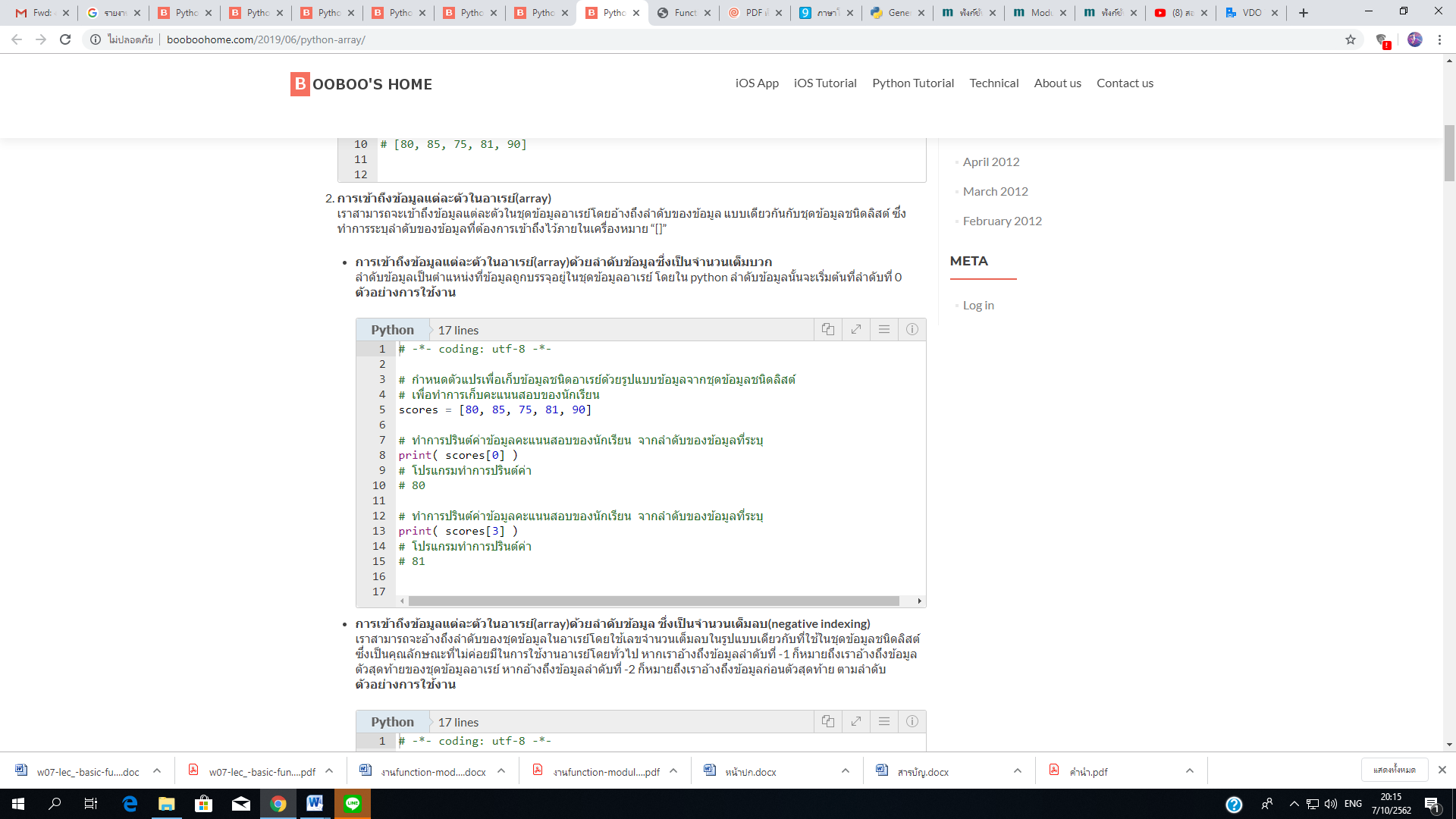
**2. การเข้าถึงข้อมูลแต่ละตัวในอาเรย์(array)**

เราสามารถจะเข้าถึงข้อมูลแต่ละตัวในชุดข้อมูลอาเรย์โดยอ้างถึงลำดับของข้อมูล แบบเดียวกันกับชุดข้อมูลชนิดลิสต์ ซึ่งทำการระบุลำดับของข้อมูลที่ต้องการเข้าถึงไว้ภายในเครื่องหมาย “[]”

**การเข้าถึงข้อมูลแต่ละตัวในอาเรย์(array)ด้วยลำดับข้อมูลซึ่งเป็นจำนวนเต็มบวก**

ลำดับข้อมูลเป็นตำแหน่งที่ข้อมูลถูกบรรจุอยู่ในชุดข้อมูลอาเรย์ โดยใน python ลำดับข้อมูลนั้นจะเริ่มต้นที่ลำดับที่ 0

**ตัวอย่างการใช้งาน**

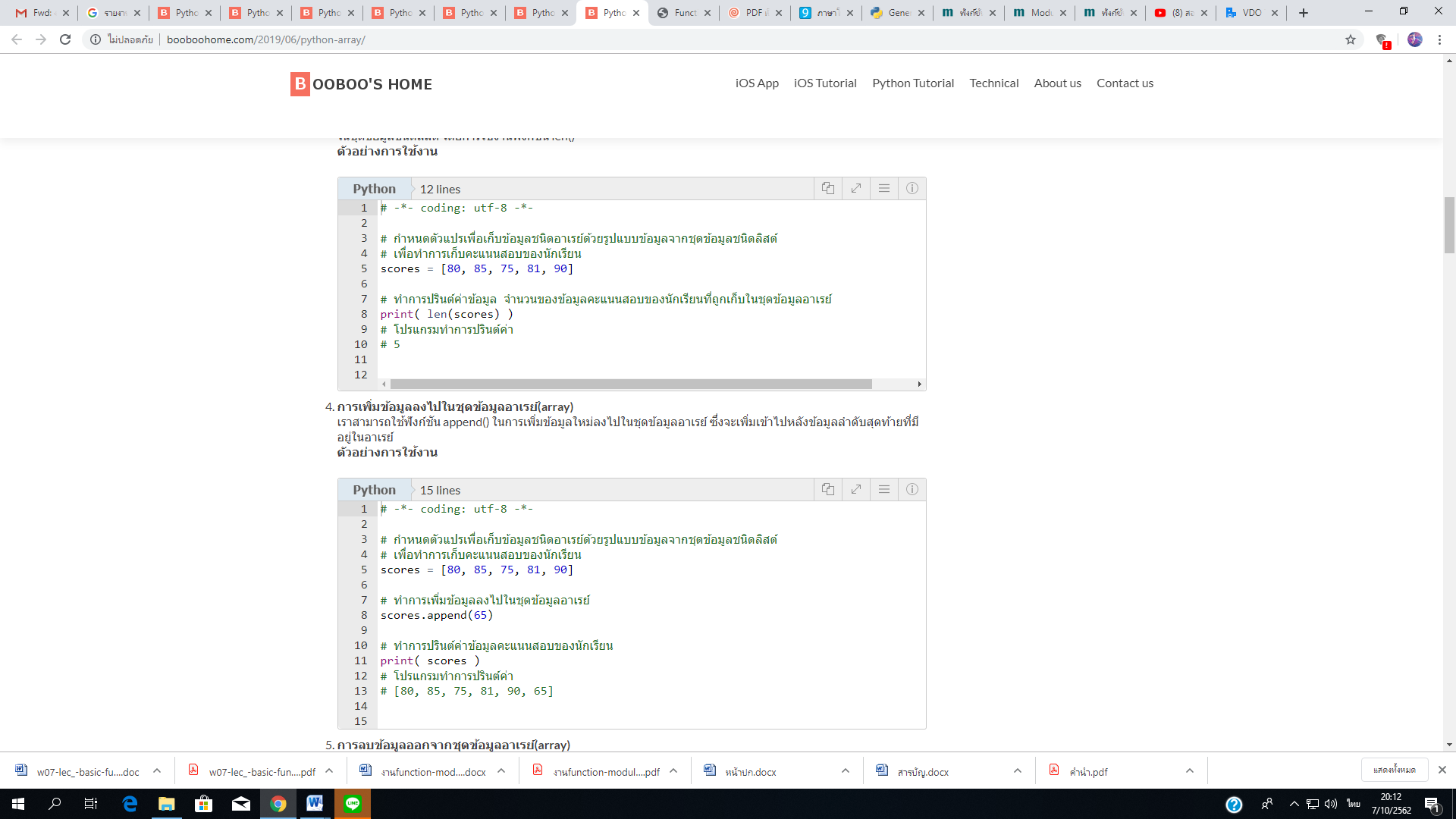
การเข้าถึงข้อมูลแต่ละตัวในอาเรย์(array)ด้วยลำดับข้อมูล ซึ่งเป็นจำนวนเต็มลบ(negative indexing)

เราสามารถจะอ้างถึงลำดับของชุดข้อมูลในอาเรย์โดยใช้เลขจำนวนเต็มลบในรูปแบบเดียวกับที่ใช้ในชุดข้อมูลชนิดลิสต์ ซึ่งเป็นคุณลักษณะที่ไม่ค่อยมีในการใช้งานอาเรย์โดยทั่วไป หากเราอ้างถึงข้อมูลลำดับที่ -1 ก็หมายถึงเราอ้างถึงข้อมูลตัวสุดท้ายของชุดข้อมูลอาเรย์ หากอ้างถึงข้อมูลลำดับที่ -2 ก็หมายถึงเราอ้างถึงข้อมูลก่อนตัวสุดท้าย ตามลำดับ

**3.การตรวจสอบจำนวนข้อมูลในอาเรย์(array)**

เราสามารถทำการตรวจสอบจำนวนของข้อมูลที่มีอยู่ภายในชุดข้อมูลอาเรย์ ซึ่งใช้รูปแบบเดียวกับการหาจำนวนข้อมูลที่มีอยู่ในชุดข้อมูลชนิดลิสต์ โดยการใช้งานฟังก์ชัน len()

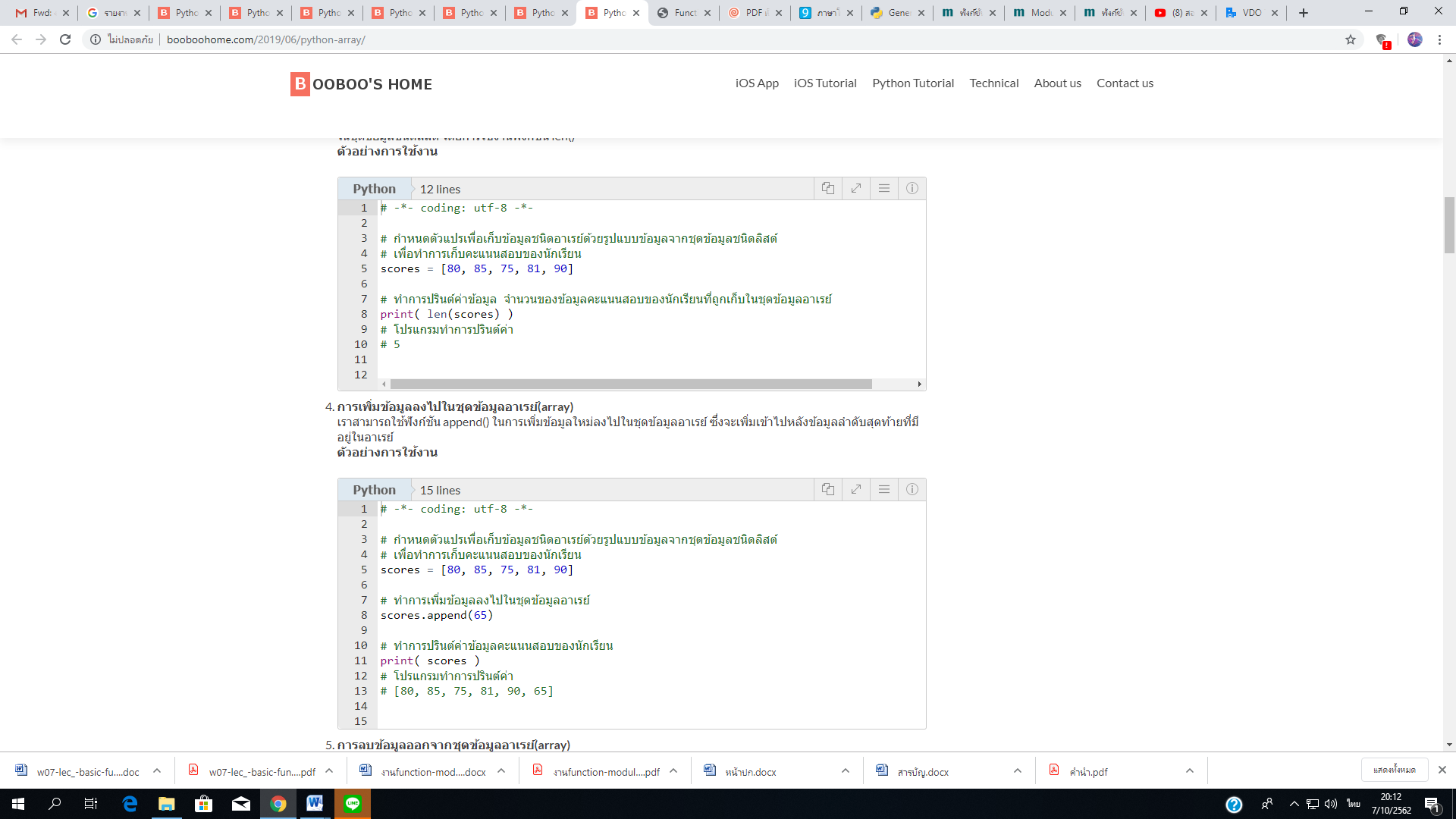
**ตัวอย่างการใช้งาน**



**4. การเพิ่มข้อมูลลงไปในชุดข้อมูลอาเรย์(array)**

เราสามารถใช้ฟังก์ชัน append() ในการเพิ่มข้อมูลใหม่ลงไปในชุดข้อมูลอาเรย์ ซึ่งจะเพิ่มเข้าไปหลังข้อมูลลำดับสุดท้ายที่มีอยู่ในอาเรย์

**ตัวอย่างการใช้งาน**

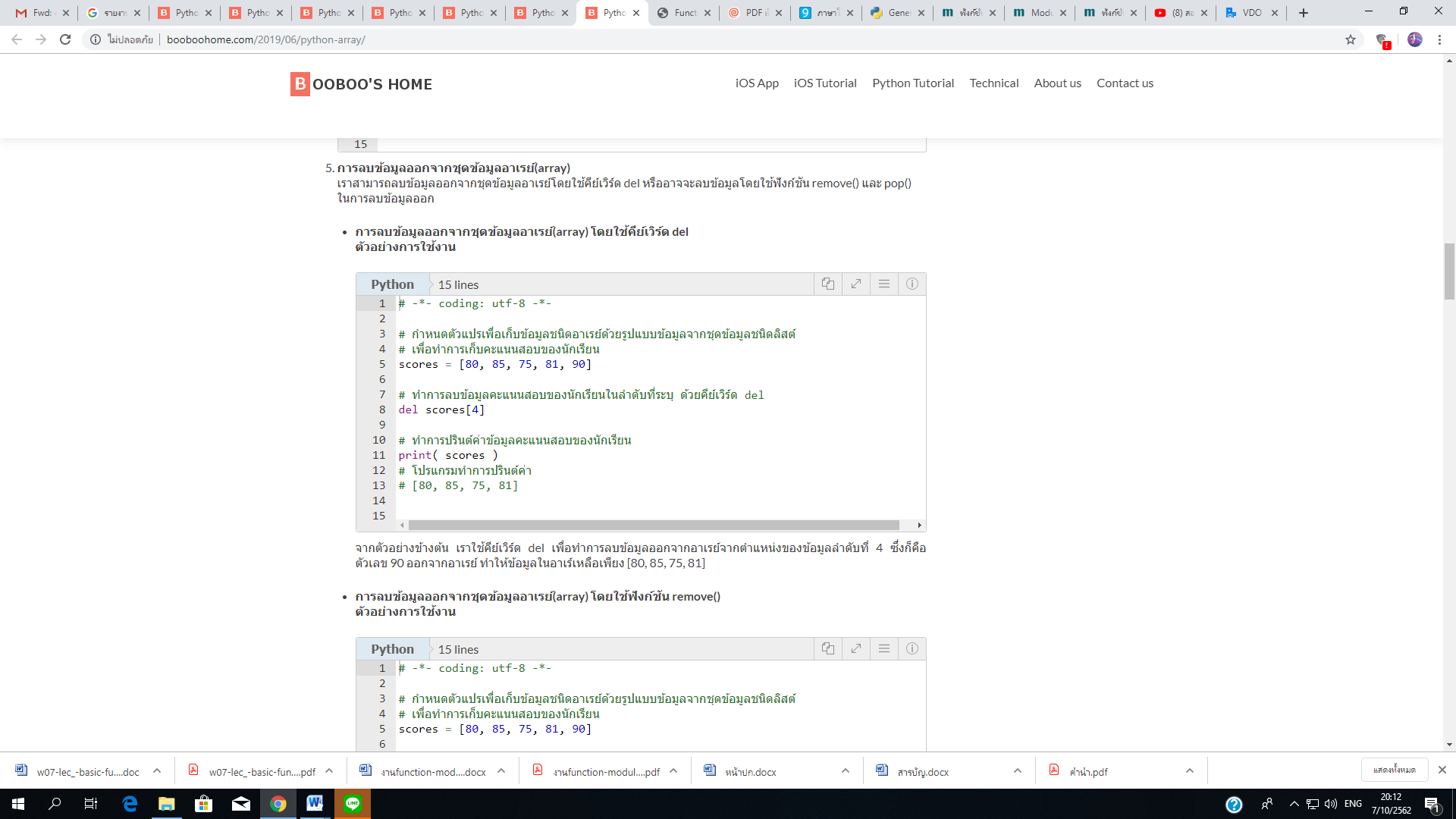


**5. การลบข้อมูลออกจากชุดข้อมูลอาเรย์(array)**

เราสามารถลบข้อมูลออกจากชุดข้อมูลอาเรย์โดยใช้คีย์เวิร์ด del หรืออาจจะลบข้อมูลโดยใช้ฟังก์ชัน remove() และ pop() ในการลบข้อมูลออก

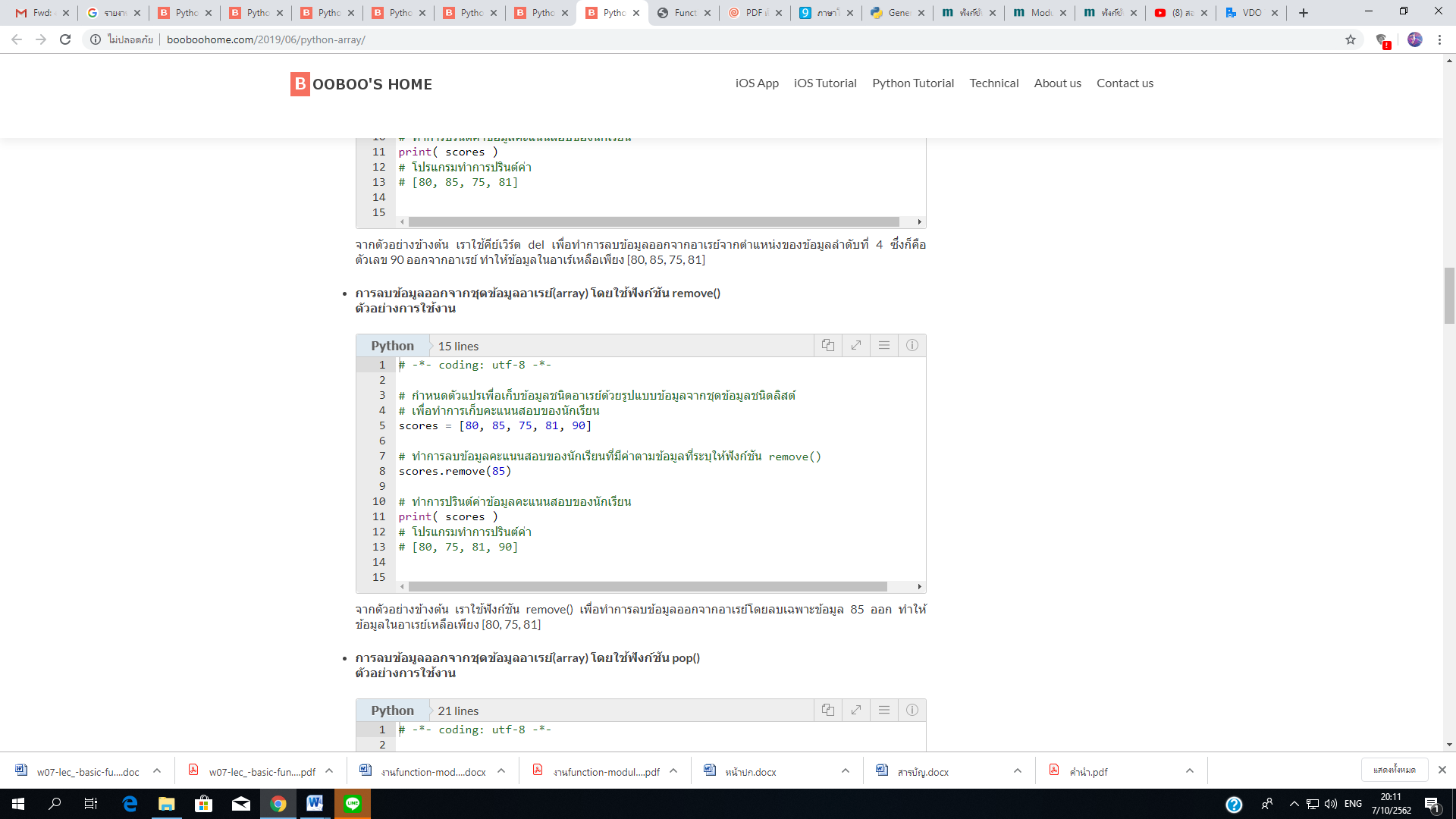
**การลบข้อมูลออกจากชุดข้อมูลอาเรย์(array) โดยใช้คีย์เวิร์ด del**

**ตัวอย่างการใช้งาน**

จากตัวอย่างข้างต้น เราใช้คีย์เวิร์ด del เพื่อทำการลบข้อมูลออกจากอาเรย์จากตำแหน่งของข้อมูลลำดับที่ 4 ซึ่งก็คือตัวเลข 90 ออกจากอาเรย์ ทำให้ข้อมูลในอาเร์เหลือเพียง [80, 85, 75, 81]

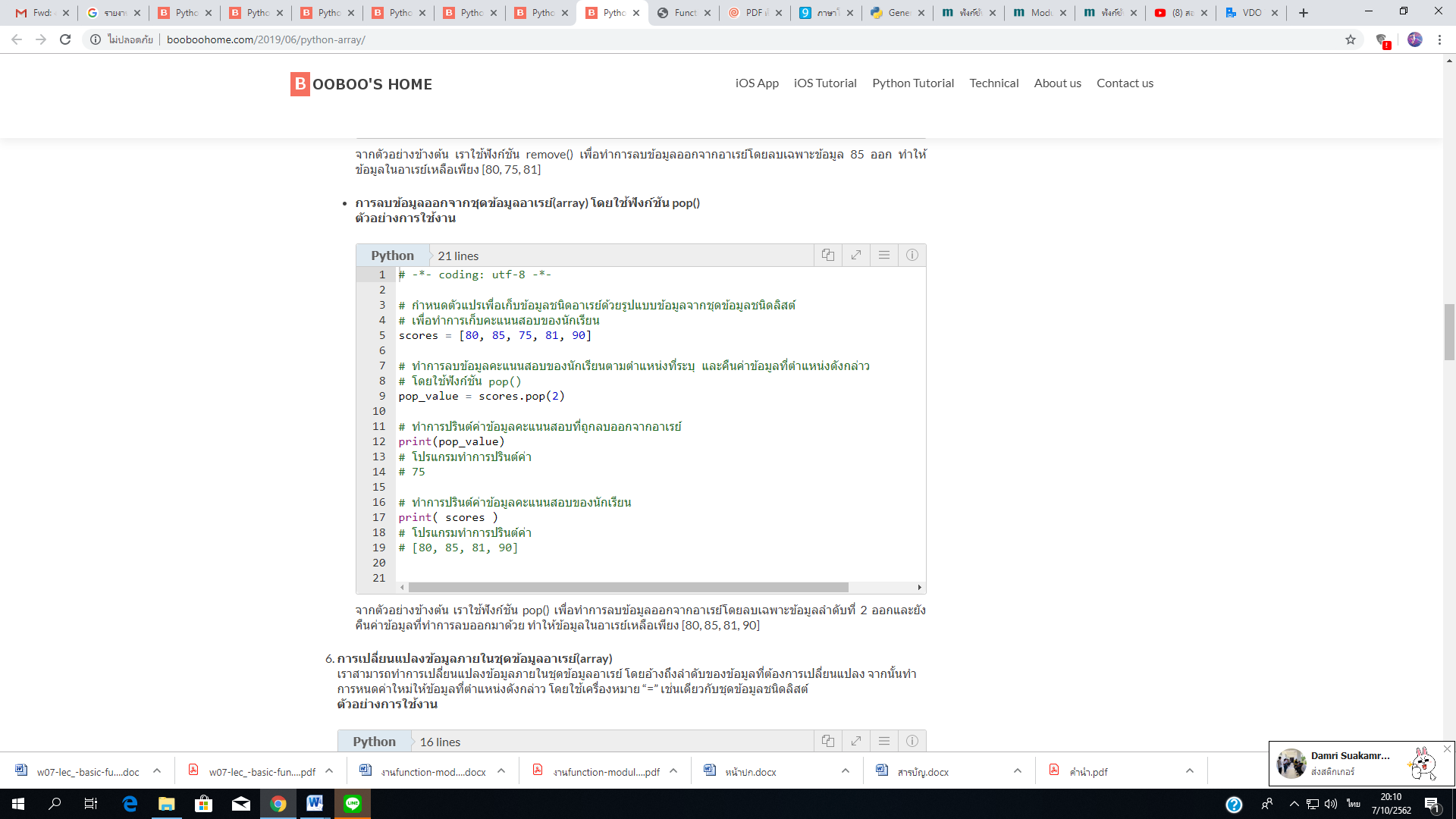
**การลบข้อมูลออกจากชุดข้อมูลอาเรย์(array) โดยใช้ฟังก์ชัน remove()**

**ตัวอย่างการใช้งาน**

**จากตัวอย่างข้างต้น** เราใช้ฟังก์ชัน remove() เพื่อทำการลบข้อมูลออกจากอาเรย์โดยลบเฉพาะข้อมูล 85 ออก ทำให้ข้อมูลในอาเรย์เหลือเพียง [80, 75, 81]

**การลบข้อมูลออกจากชุดข้อมูลอาเรย์(array) โดยใช้ฟังก์ชัน pop()**

**ตัวอย่างการใช้งาน**

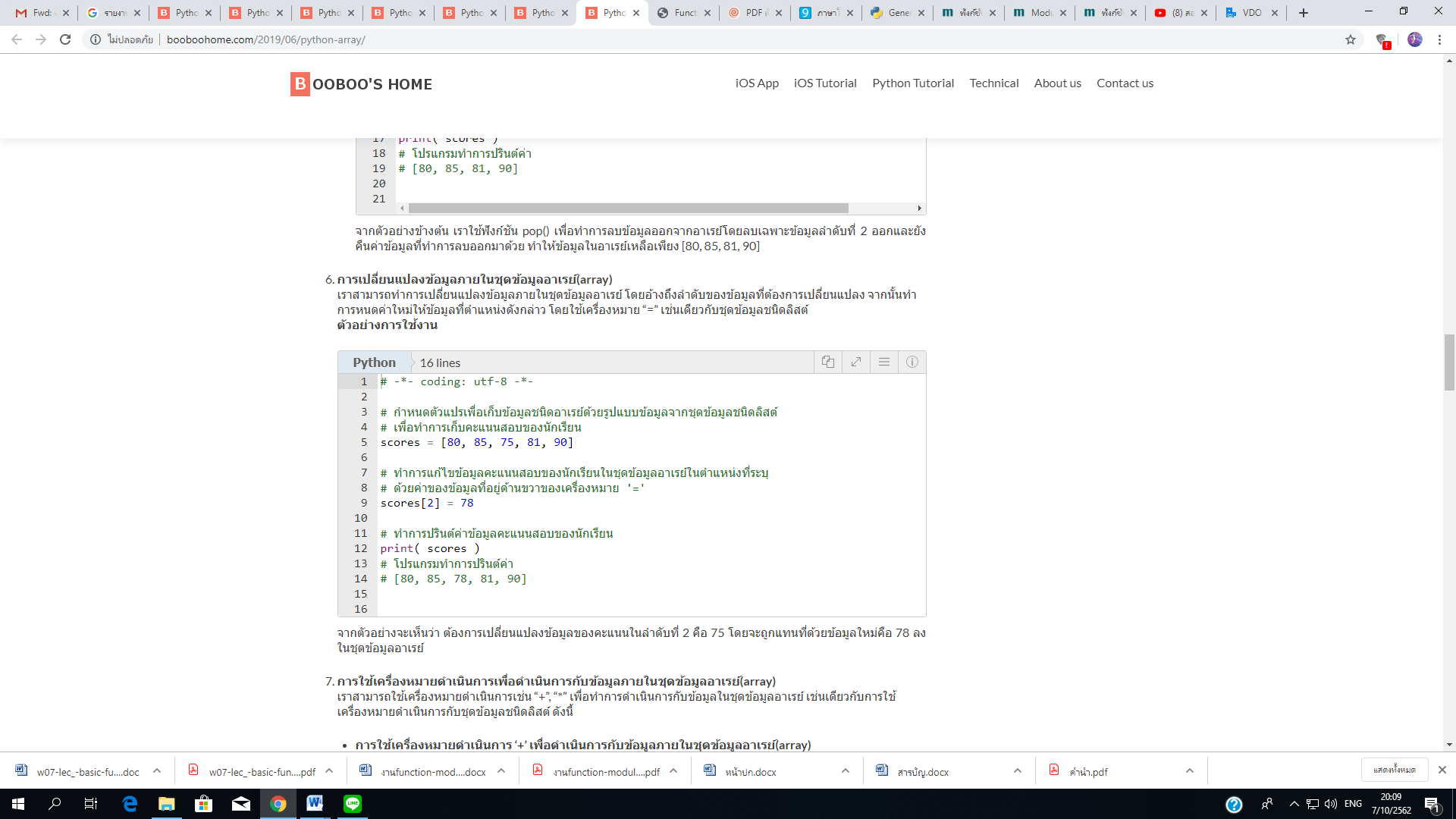


จากตัวอย่างข้างต้น เราใช้ฟังก์ชัน pop() เพื่อทำการลบข้อมูลออกจากอาเรย์โดยลบเฉพาะข้อมูลลำดับที่ 2 ออกและยังคืนค่าข้อมูลที่ทำการลบออกมาด้วย ทำให้ข้อมูลในอาเรย์เหลือเพียง [80, 85, 81, 90]

**6. การเปลี่ยนแปลงข้อมูลภายในชุดข้อมูลอาเรย์(array)**

เราสามารถทำการเปลี่ยนแปลงข้อมูลภายในชุดข้อมูลอาเรย์ โดยอ้างถึงลำดับของข้อมูลที่ต้องการเปลี่ยนแปลง จากนั้นทำการหนดค่าใหม่ให้ข้อมูลที่ตำแหน่งดังกล่าว โดยใช้เครื่องหมาย “=” เช่นเดียวกับชุดข้อมูลชนิดลิสต์

ตัวอย่างการใช้งาน



จากตัวอย่างจะเห็นว่า ต้องการเปลี่ยนแปลงข้อมูลของคะแนนในลำดับที่ 2 คือ 75 โดยจะถูกแทนที่ด้วยข้อมูลใหม่คือ 78 ลงในชุดข้อมูลอาเรย์

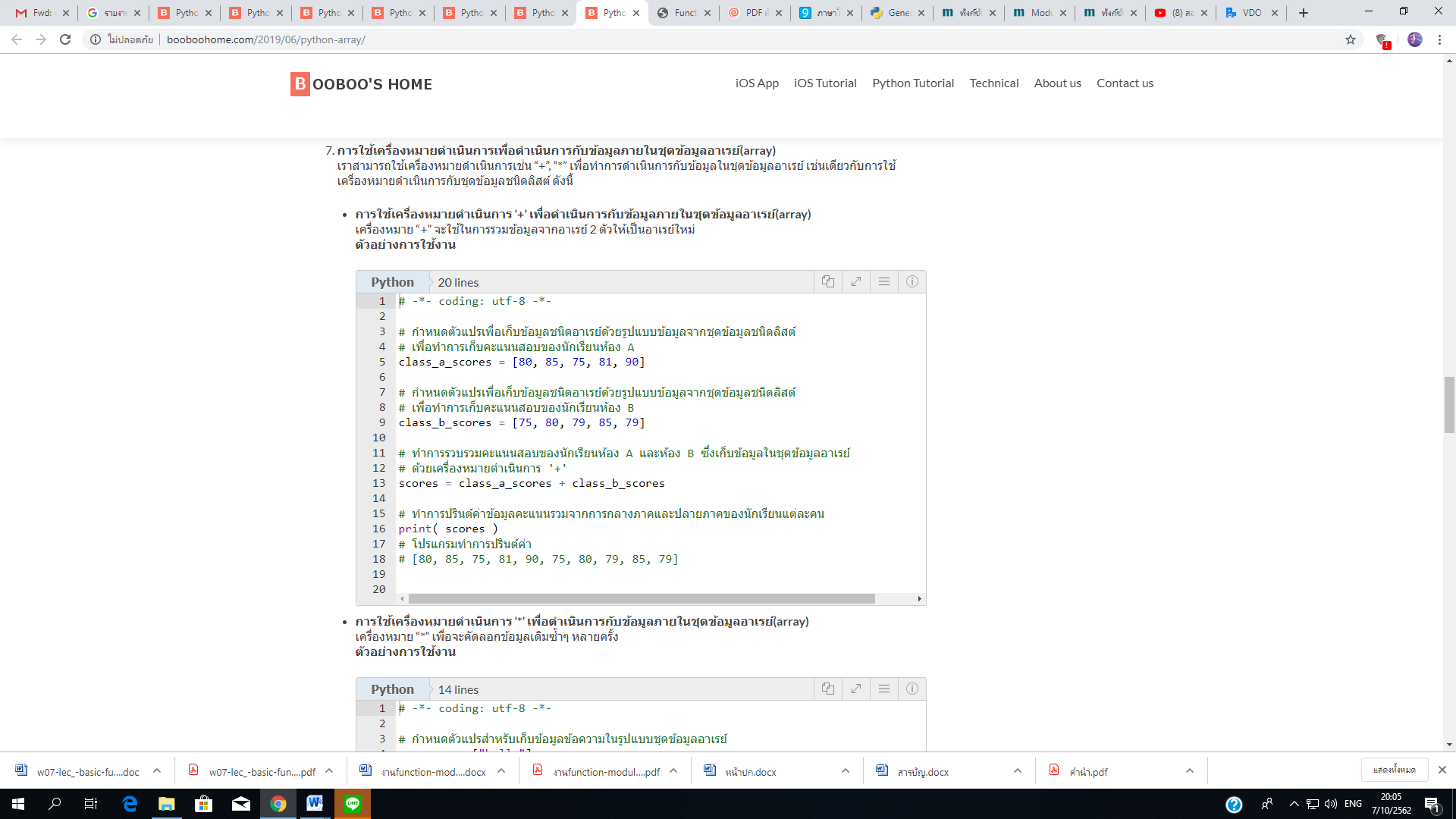
**7. การใช้เครื่องหมายดำเนินการเพื่อดำเนินการกับข้อมูลภายในชุดข้อมูลอาเรย์(array)**

เราสามารถใช้เครื่องหมายดำเนินการเช่น “+”, “\*” เพื่อทำการดำเนินการกับข้อมูลในชุดข้อมูลอาเรย์ เช่นเดียวกับการใช้เครื่องหมายดำเนินการกับชุดข้อมูลชนิดลิสต์ ดังนี้

**การใช้เครื่องหมายดำเนินการ ‘+’ เพื่อดำเนินการกับข้อมูลภายในชุดข้อมูลอาเรย์(array)**

เครื่องหมาย “+” จะใช้ในการรวมข้อมูลจากอาเรย์ 2 ตัวให้เป็นอาเรย์ใหม่

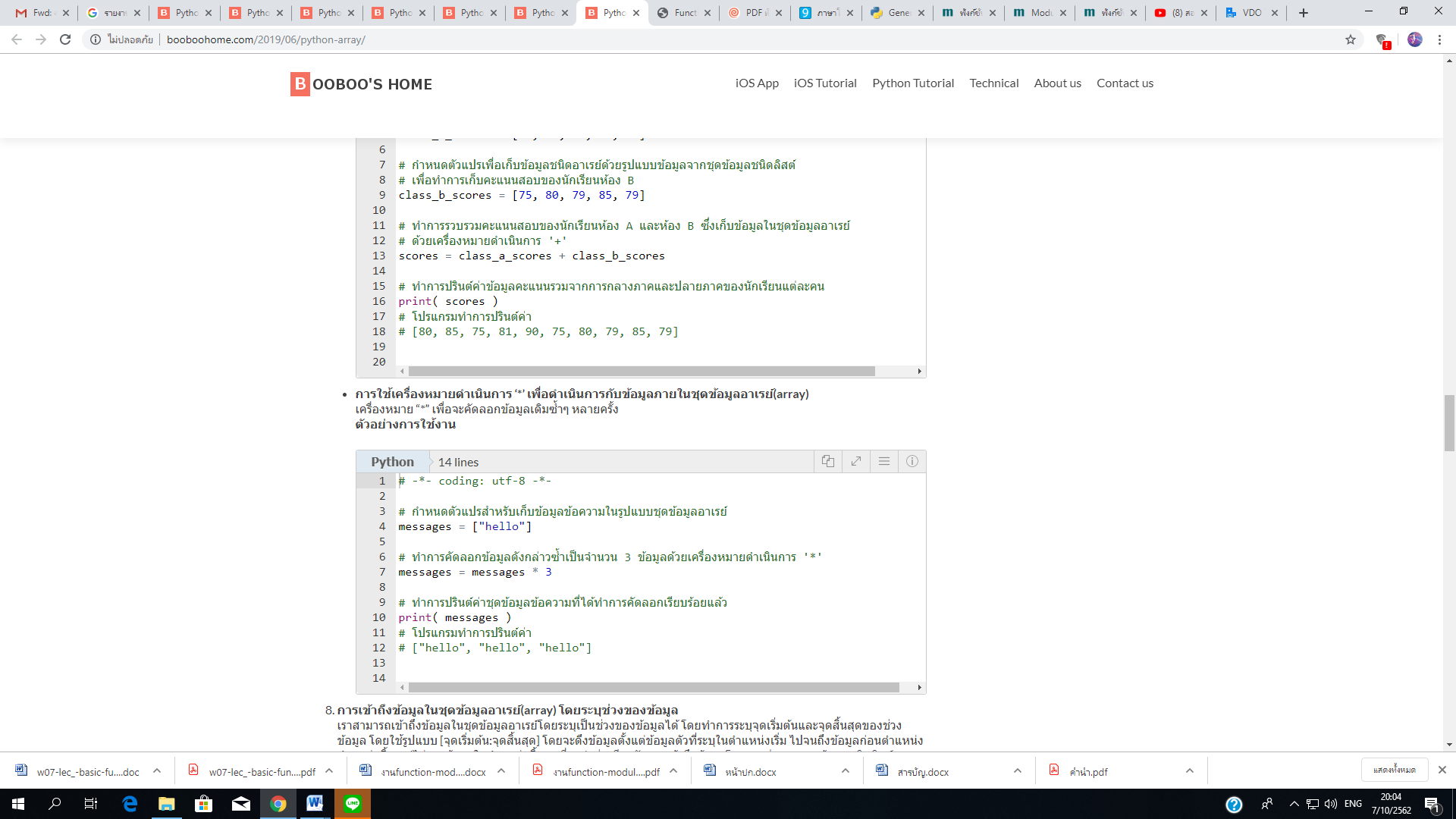
ตัวอย่างการใช้งาน



**การใช้เครื่องหมายดำเนินการ ‘\*’ เพื่อดำเนินการกับข้อมูลภายในชุดข้อมูลอาเรย์(array)**

เครื่องหมาย “\*” เพื่อจะคัดลอกข้อมูลเดิมซ้ำๆ หลายครั้ง

**ตัวอย่างการใช้งาน**



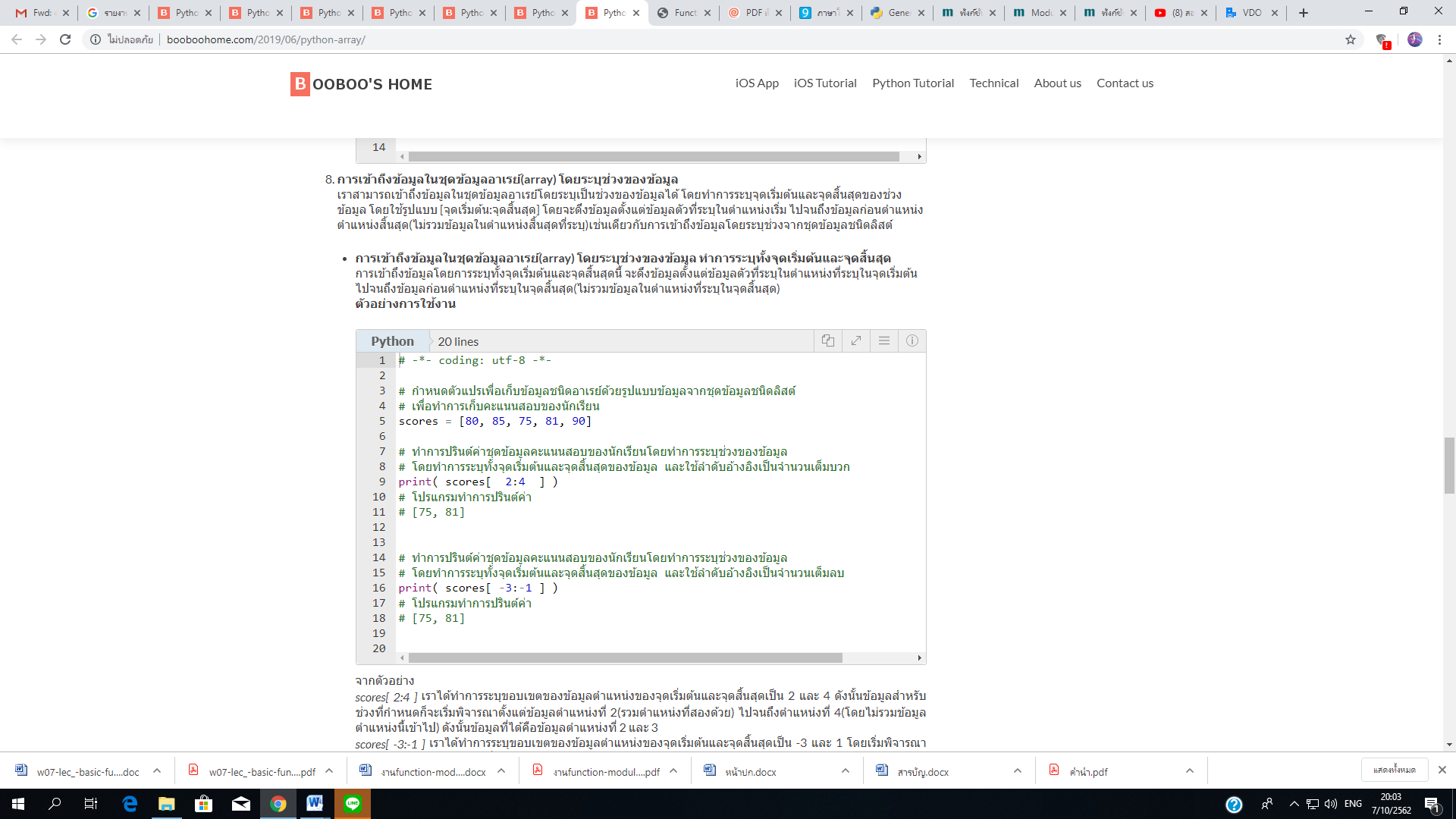
**การเข้าถึงข้อมูลในชุดข้อมูลอาเรย์(array) โดยระบุช่วงของข้อมูล**

เราสามารถเข้าถึงข้อมูลในชุดข้อมูลอาเรย์โดยระบุเป็นช่วงของข้อมูลได้ โดยทำการระบุจุดเริ่มต้นและจุดสิ้นสุดของช่วงข้อมูล โดยใช้รูปแบบ [จุดเริ่มต้น:จุดสิ้นสุด] โดยจะดึงข้อมูลตั้งแต่ข้อมูลตัวที่ระบุในตำแหน่งเริ่ม ไปจนถึงข้อมูลก่อนตำแหน่งตำแหน่งสิ้นสุด(ไม่รวมข้อมูลในตำแหน่งสิ้นสุดที่ระบุ)เช่นเดียวกับการเข้าถึงข้อมูลโดยระบุช่วงจากชุดข้อมูลชนิดลิสต์

**การเข้าถึงข้อมูลในชุดข้อมูลอาเรย์(array) โดยระบุช่วงของข้อมูล ทำการระบุทั้งจุดเริ่มต้นและจุดสิ้นสุด**

การเข้าถึงข้อมูลโดยการระบุทั้งจุดเริ่มต้นและจุดสิ้นสุดนี้ จะดึงข้อมูลตั้งแต่ข้อมูลตัวที่ระบุในตำแหน่งที่ระบุในจุดเริ่มต้น ไปจนถึงข้อมูลก่อนตำแหน่งที่ระบุในจุดสิ้นสุด(ไม่รวมข้อมูลในตำแหน่งที่ระบุในจุดสิ้นสุด)

**ตัวอย่างการใช้งาน**

**จากตัวอย่าง**

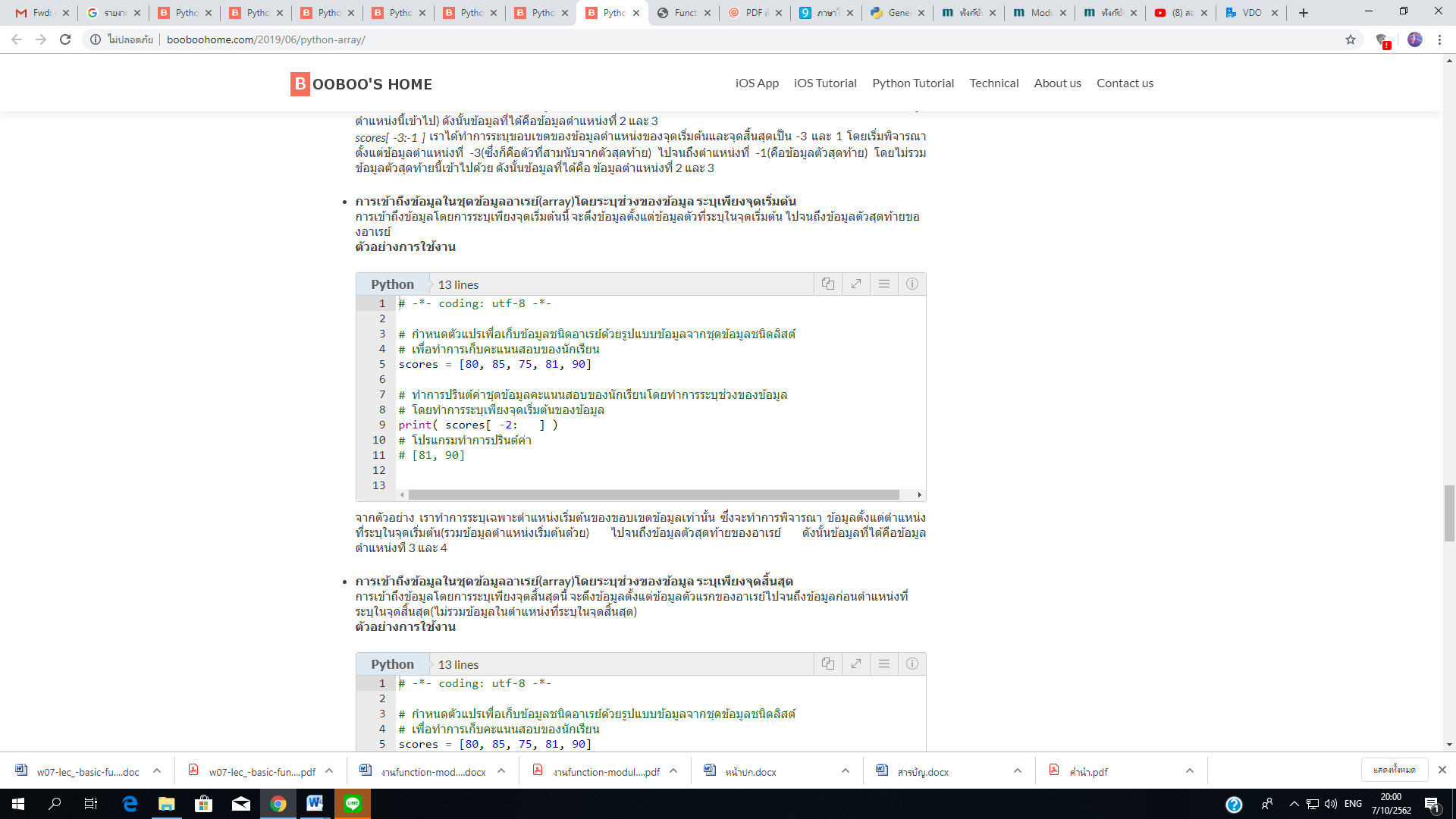
scores[ 2:4 ] เราได้ทำการระบุขอบเขตของข้อมูลตำแหน่งของจุดเริ่มต้นและจุดสิ้นสุดเป็น 2 และ 4 ดังนั้นข้อมูลสำหรับช่วงที่กำหนดก็จะเริ่มพิจารณาตั้งแต่ข้อมูลตำแหน่งที่ 2(รวมตำแหน่งที่สองด้วย) ไปจนถึงตำแหน่งที่ 4(โดยไม่รวมข้อมูลตำแหน่งนี้เข้าไป) ดังนั้นข้อมูลที่ได้คือข้อมูลตำแหน่งที่ 2 และ 3

scores[ -3:-1 ] เราได้ทำการระบุขอบเขตของข้อมูลตำแหน่งของจุดเริ่มต้นและจุดสิ้นสุดเป็น -3 และ 1 โดยเริ่มพิจารณาตั้งแต่ข้อมูลตำแหน่งที่ -3(ซึ่งก็คือตัวที่สามนับจากตัวสุดท้าย) ไปจนถึงตำแหน่งที่ -1(คือข้อมูลตัวสุดท้าย) โดยไม่รวมข้อมูลตัวสุดท้ายนี้เข้าไปด้วย ดังนั้นข้อมูลที่ได้คือ ข้อมูลตำแหน่งที่ 2 และ 3

**8.การเข้าถึงข้อมูลในชุดข้อมูลอาเรย์(array)โดยระบุช่วงของข้อมูล ระบุเพียงจุดเริ่มต้น**

การเข้าถึงข้อมูลโดยการระบุเพียงจุดเริ่มต้นนี้ จะดึงข้อมูลตั้งแต่ข้อมูลตัวที่ระบุในจุดเริ่มต้น ไปจนถึงข้อมูลตัวสุดท้ายของอาเรย์

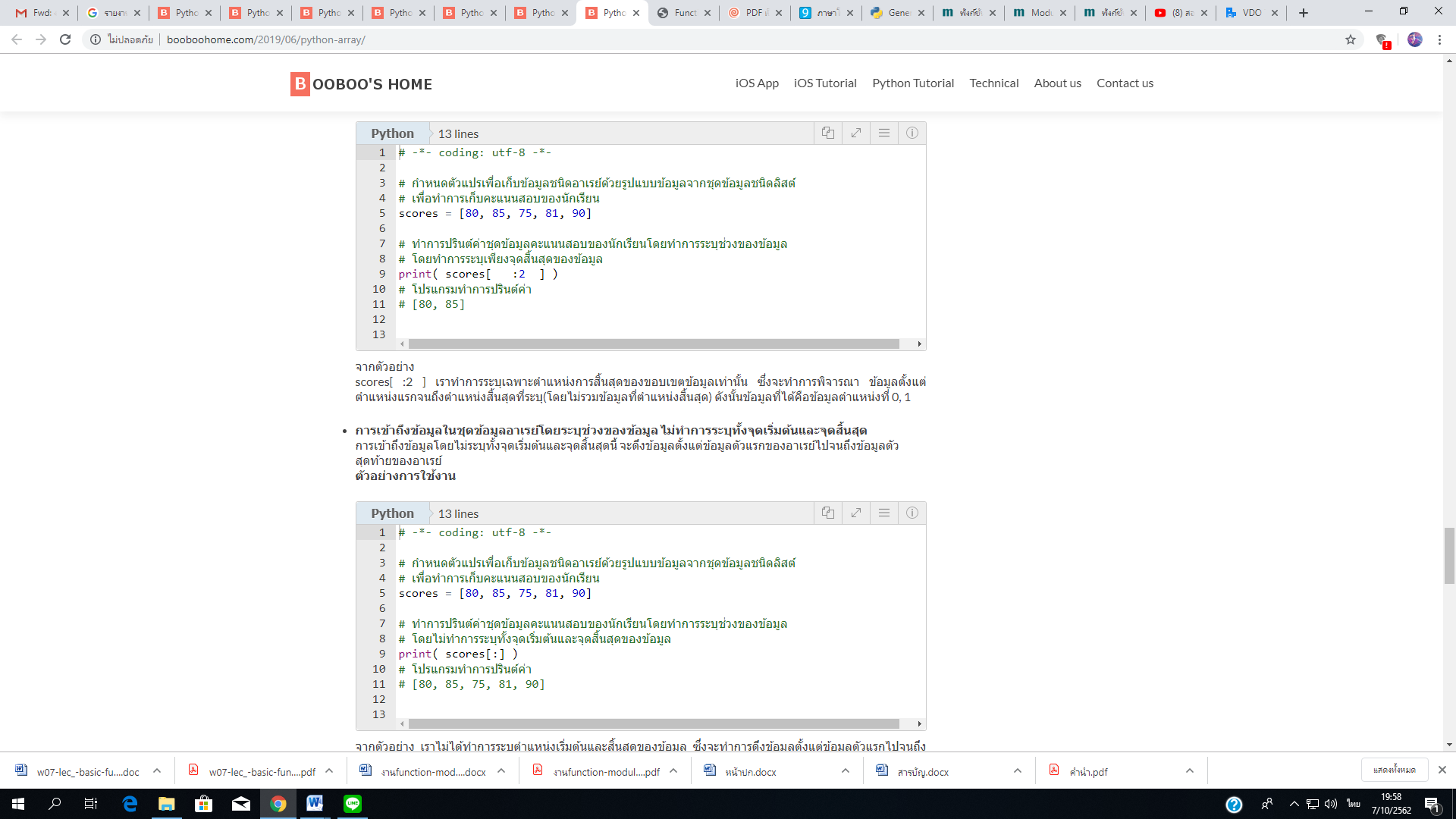
**ตัวอย่างการใช้งาน**

  
  
จากตัวอย่าง เราทำการระบุเฉพาะตำแหน่งเริ่มต้นของขอบเขตข้อมูลเท่านั้น ซึ่งจะทำการพิจารณา ข้อมูลตั้งแต่ตำแหน่งที่ระบุในจุดเริ่มต้น(รวมข้อมูลตำแหน่งเริ่มต้นด้วย) ไปจนถึงข้อมูลตัวสุดท้ายของอาเรย์ ดังนั้นข้อมูลที่ได้คือข้อมูลตำแหน่งที 3 และ 4

**การเข้าถึงข้อมูลในชุดข้อมูลอาเรย์(array)โดยระบุช่วงของข้อมูล ระบุเพียงจุดสิ้นสุด**

การเข้าถึงข้อมูลโดยการระบุเพียงจุดสิ้นสุดนี้ จะดึงข้อมูลตั้งแต่ข้อมูลตัวแรกของอาเรย์ไปจนถึงข้อมูลก่อนตำแหน่งที่ระบุในจุดสิ้นสุด(ไม่รวมข้อมูลในตำแหน่งที่ระบุในจุดสิ้นสุด)

**ตัวอย่างการใช้งาน**



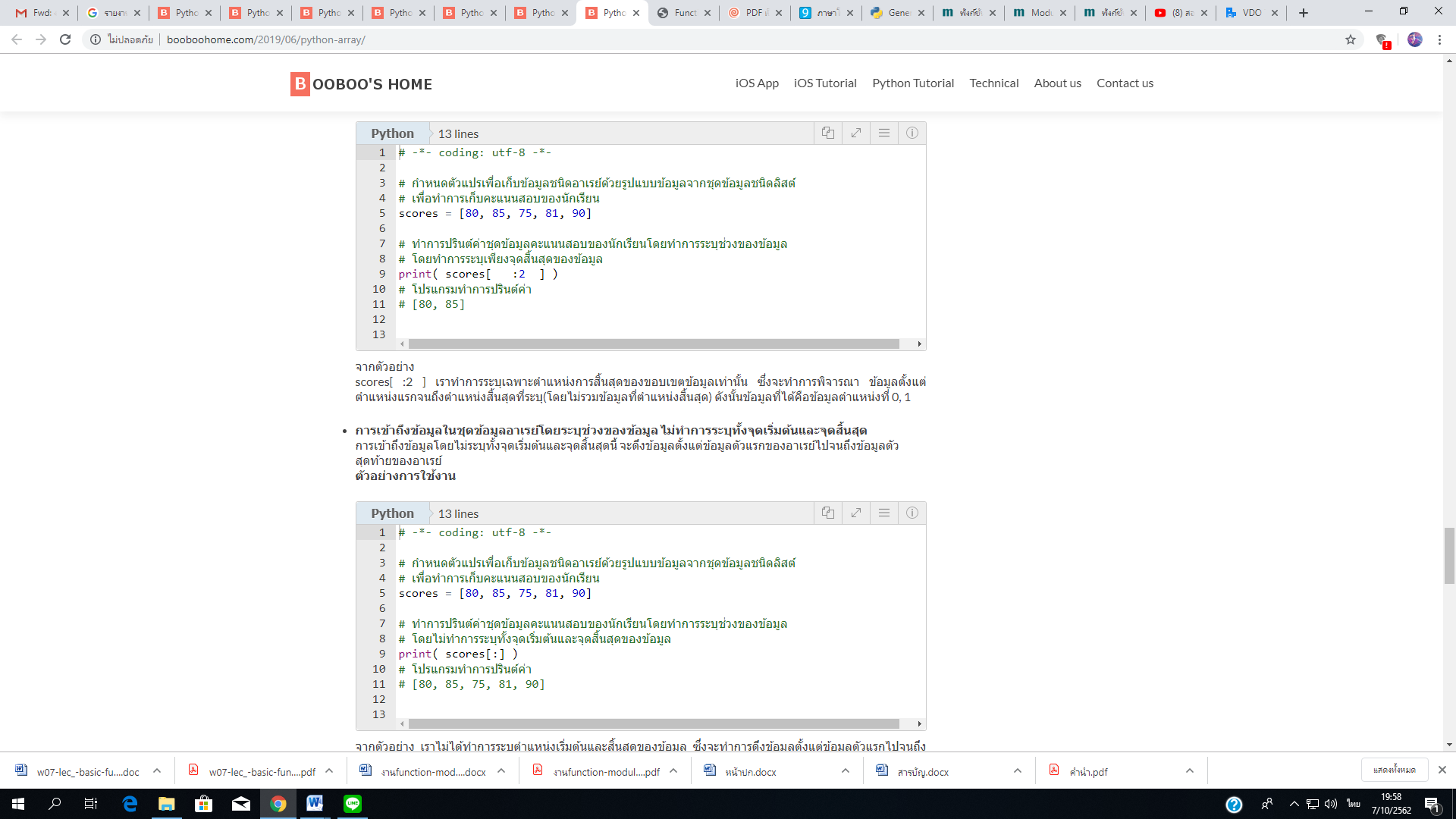
จากตัวอย่าง

scores[ :2 ] เราทำการระบุเฉพาะตำแหน่งการสิ้นสุดของขอบเขตข้อมูลเท่านั้น ซึ่งจะทำการพิจารณา ข้อมูลตั้งแต่ตำแหน่งแรกจนถึงตำแหน่งสิ้นสุดที่ระบุ(โดยไม่รวมข้อมูลที่ตำแหน่งสิ้นสุด) ดังนั้นข้อมูลที่ได้คือข้อมูลตำแหน่งที่ 0, 1

**การเข้าถึงข้อมูลในชุดข้อมูลอาเรย์โดยระบุช่วงของข้อมูล ไม่ทำการระบุทั้งจุดเริ่มต้นและจุดสิ้นสุด**

การเข้าถึงข้อมูลโดยไม่ระบุทั้งจุดเริ่มต้นและจุดสิ้นสุดนี้ จะดึงข้อมูลตั้งแต่ข้อมูลตัวแรกของอาเรย์ไปจนถึงข้อมูลตัวสุดท้ายของอาเรย์

**ตัวอย่างการใช้งาน**

จากตัวอย่าง เราไม่ได้ทำการระบุตำแหน่งเริ่มต้นและสิ้นสุดของข้อมูล ซึ่งจะทำการดึงข้อมูลตั้งแต่ข้อมูลตัวแรกไปจนถึงตัวสุดท้ายจากชุดข้อมูลอาเรย์

9. ฟังก์ชันที่สามารถใช้งานของอาเรย์(เช่นเดียวกับชุดข้อมูลชนิดลิสต์)

ฟังก์ชัน รายละเอียด

append() เพิ่มข้อมูลเข้าไปต่อจากตำแหน่งสุดท้ายของชุดข้อมูล

extend() ขยาย(extend)ข้อมูลทุกตัวในอาเรย์ไปไว้ยังอาเรย์อื่น

insert() เพิ่มข้อมูลเข้าไปตรงตำแหน่งที่ระบุของชุดข้อมูล

remove() ลบข้อมูลที่ระบุออกจากอาเรย์

pop() ลบข้อมูลและคืนค่าข้อมูลดังกล่าวจากตำแหน่งของข้อมูลที่ระบุในอาเรย์

clear() ลบข้อมูลทุกตัวออกจากอาเรย์

index() คืนค่าของตำแหน่งแรกที่พบข้อมูลที่ระบุ

count() นับจำนวนของข้อมูลทั้งหมดจากตัวแปรที่ส่งเข้าไป

sort() เรียงลำดับข้อมูลในชุดข้อมูล(ค่าตั้งต้นคือเรียงจากน้อยไปหามาก)

reverse() สลับลำดับของการเรียงข้อมูลจากหน้าไปหลัง

copy() คืนค่าลิสต์ใหม่ซึ่งคัดลอกมาจากข้อมูลในอาเรย์เดิม

การสร้างอาเรย์หลายมิติ(multidimensional arrays)

อาเรย์หลายมิติก็คือ การมีข้อมูลอาเรย์ซ้อนอยู่ภายในข้อมูลอาเรย์อีกชั้นหนึ่ง ซึ่งอาจจะมีอาเรย์ย่อยๆ เป็นส่วนประกอบภายใน

**10. การสร้างอาเรย์หลายมิติ**

**ตัวอย่างการใช้งาน**

