



สร้าง GUI Application  
ด้วยภาษา Python

# สิ่งที่จะได้เรียนในคอร์ส

คอร์สนี้ผู้เรียนจะได้เรียนรู้การสร้าง GUI Application ด้วยภาษา Python ตั้งแต่พื้นฐานจนนำไปประยุกต์ใช้งาน เพื่อสร้างแอปพลิเคชันด้านต่างๆได้ตามความต้องการ โดยเนื้อหาในคอร์สจะแบ่งออกเป็น 6 ส่วน

# สิ่งที่จะได้เรียนในคอร์ส

1. พื้นฐานการสร้าง GUI ด้วยภาษา Python
2. สร้างแอปจัดการงาน (CheckList Application)
3. สร้างแอปแปลงอุณหภูมิ (Temperature Converter)
4. สร้างแอปเครื่องคิดเลข (Calculator Application)
5. สร้างแอปจดบันทึก (Note Application)
6. นำแอปไปใช้งานจริง (Build To Exe)

# ปูพื้นฐานการสร้าง GUI

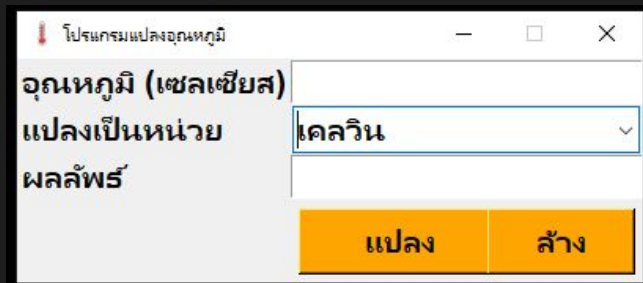
- เรียนรู้การสร้างหน้าจอโปรแกรม
- การสร้าง Widget พื้นฐาน (ปุ่ม , ข้อความ)
- เรียนรู้การจัดการ Widget ด้วย Layout
- กระบวนการทำงานในโปรแกรมพื้นฐาน  
เช่น ปิดโปรแกรม , ข้อความแจ้งเตือน เป็นต้น

# แอปจัดการงาน (CheckList)

- ออกแบบหน้าจอโปรแกรม
- ระบบบันทึกข้อมูล
- ระบบแสดงรายการด้วย ListBox
- ระบบลบข้อมูล
- ระบบปิดโปรแกรม



# แอปแปลงอุณหภูมิ (Temperature Converter)

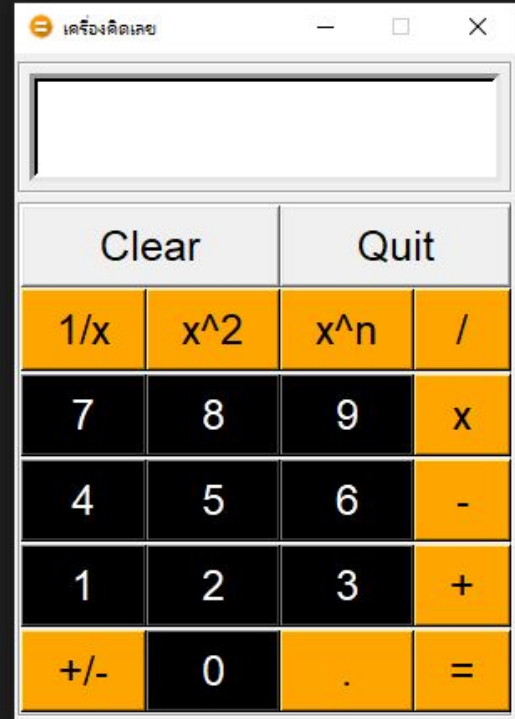


The screenshot shows a window titled "โปรแกรมแปลงอุณหภูมิ" (Temperature Converter Program). It contains a text input field labeled "อุณหภูมิ (เซลเซียส)" (Temperature (Celsius)), a dropdown menu labeled "แปลงเป็นหน่วย" (Convert to unit) with "เคลวิน" (Kelvin) selected, and an empty output field labeled "ผลลัพธ์" (Result). At the bottom are two orange buttons: "แปลง" (Convert) and "ล้าง" (Clear).

- ระบบตัวเลือกหน่วยองศา (เคลวิน,ฟาเรนไฮน์)
- ระบบคำนวณแปลงองศา

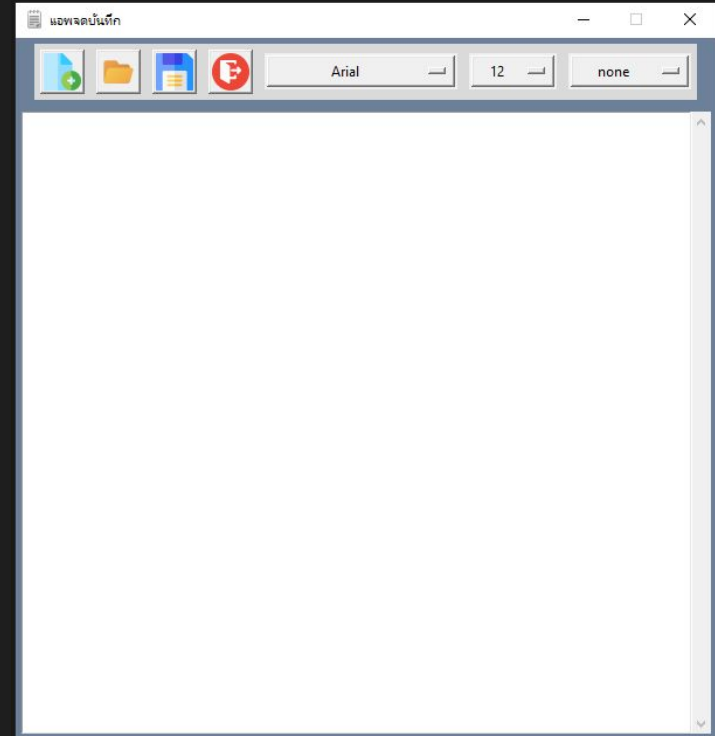
# แอปเครื่องคิดเลข (Calculator)

- จัดการตัวเลข
- คำนวณคณิตศาสตร์พื้นฐาน  
(บวก , ลบ คูณ , หาร)
- คำนวณเลขยกกำลัง
- คำนวณตัวเลขที่มีจุดทศนิยม



# แอปจดบันทึก (Note Application)

- จัดการฟอนต์
- จัดการขนาดข้อความ
- จัดการรูปแบบข้อความ
- ระบบเขียนไฟล์
- ระบบอ่านไฟล์





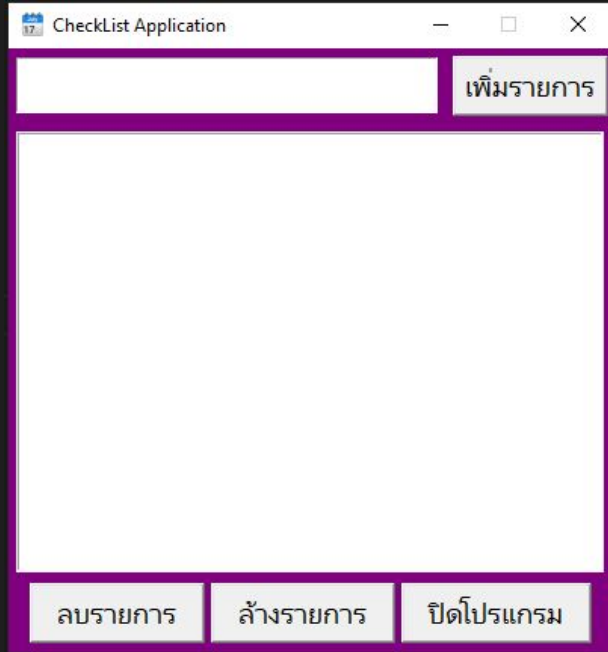
ต้องมีพื้นฐานอะไรบ้าง

# ต้องมีพื้นฐานอะไรบ้าง

- พื้นฐานการเขียนโปรแกรมภาษา Python
- เข้าใจโครงสร้างข้อมูลในภาษา Python (List , Dictionary)
- พื้นฐานการเขียนโปรแกรมเชิงวัตถุด้วยภาษา Python (Class , Method , Object)

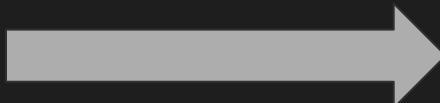
# CheckList Application

# ออกแบบโปรแกรม (User Interface)

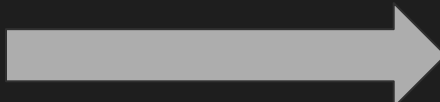


# ออกแบบโปรแกรม (User Interface)

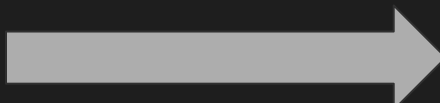
The screenshot shows a window titled "CheckList Application". It features a purple border and a white background. At the top left is a text input field. To its right is a button labeled "เพิ่มรายการ" (Add Item). The main area is a large empty rectangle. At the bottom, there are three buttons labeled "ลบรายการ" (Delete Item), "ล้างรายการ" (Clear Item), and "ปิดโปรแกรม" (Close Program).



Input Frame



Output Frame



Button Frame

# ListBox Widget

คือ Widget ที่มีลักษณะเป็นกลุ่มข้อมูล (List)  
ภายใน ListBox จะบรรจุข้อมูลสมาชิกที่มีลักษณะเป็น  
รายการมากกว่า 1 รายการ โดยผู้ใช้งานโปรแกรมนั้น  
สามารถเลือกรายการจาก ListBox มาใช้งานได้

# ListBox Method

- insert คือ เมธอดสำหรับแทรกรายการในตำแหน่งที่ต้องการ
- delete คือ เมธอดสำหรับลบรายการออกจาก ListBox

# Temperature Converter



# ออกแบบโปรแกรม (User Interface)

โปรแกรมแปลงอุณหภูมิ

อุณหภูมิ (เซลเซียส)

แปลงเป็นหน่วย

เคลวิน

ผลลัพธ์

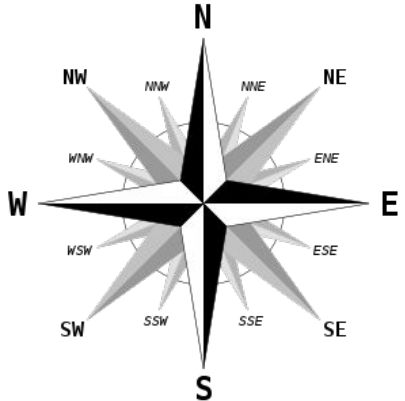
แปลง

ล้าง

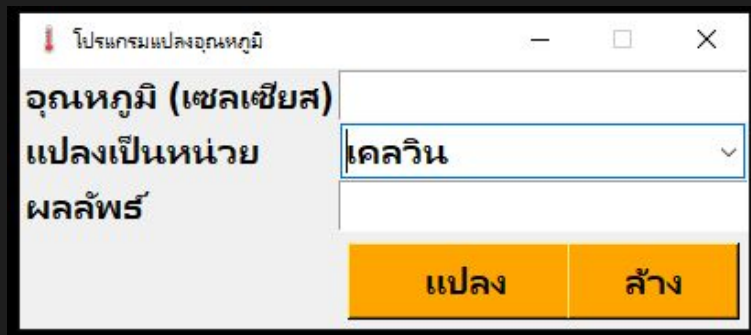
# Grid (Layout)

คือ รูปแบบการจัดวางตำแหน่งของ Widget ในพื้นที่ แบบตารางที่มีลักษณะเป็นแบบแถว (แนวนอน) และ แบบคอลัมน์ (แนวตั้ง) โดยวาง Widget ลงไปในแต่ละช่องซึ่งมีเลขลำดับช่องเริ่มต้นที่ 0

# ลักษณะการจัดวาง (Options)

Options	ความหมาย
 <p>Sticky</p>	<p>จัดตำแหน่งการวาง Widget ในทิศทางที่ต้องการ (ค่าเริ่มต้นอยู่ตรงกลาง) โดยกำหนดทิศผ่านตัวอักษร</p> <ul style="list-style-type: none"><li>• N (ทิศเหนือ)</li><li>• E (ตะวันออก)</li><li>• S (ใต้)</li><li>• W (ตะวันตก)</li><li>• NE (ทิศตะวันออกเฉียงเหนือ)</li><li>• NW (ทิศตะวันตกเฉียงเหนือ)</li><li>• SE (ทิศตะวันออกเฉียงใต้)</li><li>• SW (ทิศตะวันตกเฉียงใต้)</li></ul>

# ออกแบบโปรแกรม (User Interface)



โปรแกรมแปลงอุณหภูมิ

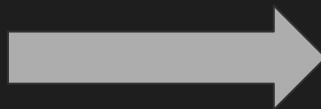
อุณหภูมิ (เซลเซียส)

แปลงเป็นหน่วย

ผลลัพธ์

เคลวิน

แปลง ล้าง



Combobox

# สูตรแปลงอุณหภูมิ °C

แปลงเป็นฟาเรนไฮน์

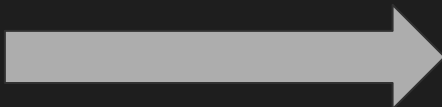
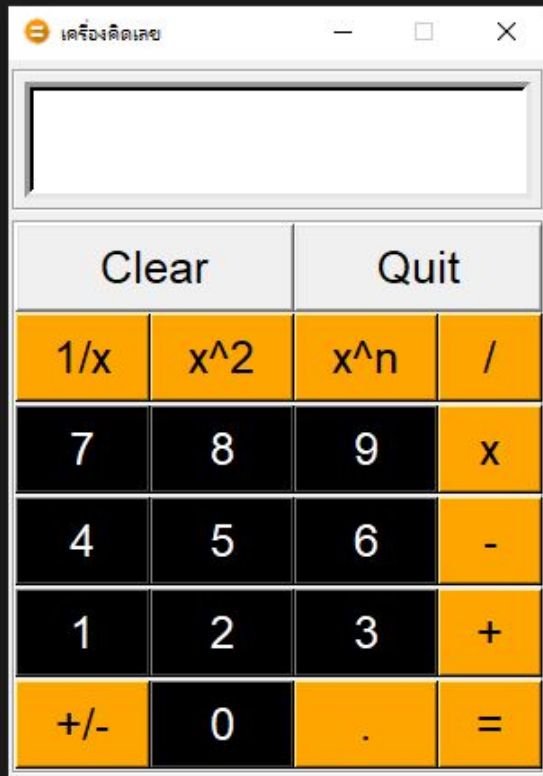
$$F = C \times 1.80 + 32$$

แปลงเป็นเคลวิน

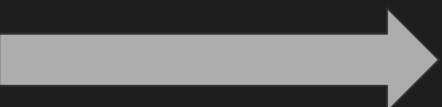
$$K = C + 273$$

เครื่องคิดเลข

# ออกแบบโปรแกรม



Display Frame



Button Frame

# Grid (Layout)

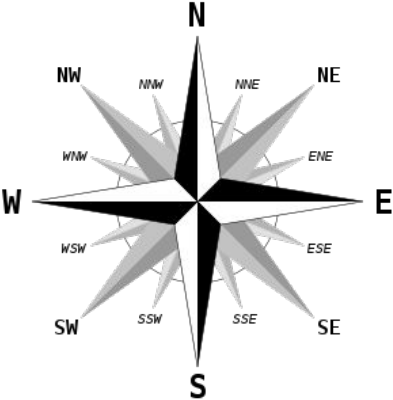
คือ รูปแบบการจัดวางตำแหน่งของ Widget ในพื้นที่แบบตารางที่มีลักษณะเป็นแบบแถว (แนวนอน) และ แบบคอลัมน์ (แนวตั้ง) โดยวาง Widget ลงไปในแต่ละช่องซึ่งมีเลขลำดับช่องเริ่มต้นที่ 0



# ลักษณะการจัดวาง (Options)

Options	ความหมาย
<b>Columnspan</b>	จำนวนคอลัมน์ที่ต้องการใช้ ค่าเริ่มต้นคือ 1 คอลัมน์

# ลักษณะการจัดวาง (Options)

Options	ความหมาย
 <p>Sticky</p>	<p>จัดตำแหน่งการวาง Widget ในทิศทางที่ต้องการ (ค่าเริ่มต้นอยู่ตรงกลาง) โดยกำหนดทิศผ่านตัวอักษร</p> <ul style="list-style-type: none"><li>• N (ทิศเหนือ)</li><li>• E (ตะวันออก)</li><li>• S (ใต้)</li><li>• W (ตะวันตก)</li><li>• NE (ทิศตะวันออกเฉียงเหนือ)</li><li>• NW (ทิศตะวันตกเฉียงเหนือ)</li><li>• SE (ทิศตะวันออกเฉียงใต้)</li><li>• SW (ทิศตะวันตกเฉียงใต้)</li></ul>

# ตัวดำเนินการ (Operator)

Operator	ความหมาย
Inverse	$1/x$ ( $x$ คือตัวเลขที่ไม่ใช่ 0)
Square	$x^2$ (เลขที่ยกกำลัง 2)
Exponent	$x^n$ (เลขยกกำลัง $n$ )
Divide	/ (การหาร)
Multiply	$\times$ (การคูณ)

# ตัวดำเนินการ (Operator)

Operator	ความหมาย
Add	+ (การบวก)
Subtract	- (การลบ)
Equal	= (เท่ากับ)
Decimal	. (จุดทศนิยม)
Negate	+/- (เครื่องหมายตรงข้าม)

# State Widget

คือ การกำหนดให้ Widget นั้นสามารถทำงานได้หรือไม่

- DISABLED คือ ไม่สามารถทำงานได้
- NORMAL คือ ทำงานปกติ

# Equal Function (คำนวณทางคณิตศาสตร์)

โครงสร้างคำสั่ง

`firstNumber operator secondNumber`

# Equal Function (คำนวณทางคณิตศาสตร์)

โครงสร้างคำสั่ง

firstNumber operator secondNumber



50.2

+

30.5

# Equal Function (คำนวณทางคณิตศาสตร์)

โครงสร้างคำสั่ง

firstNumber operator secondNumber



100

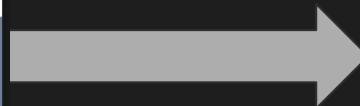
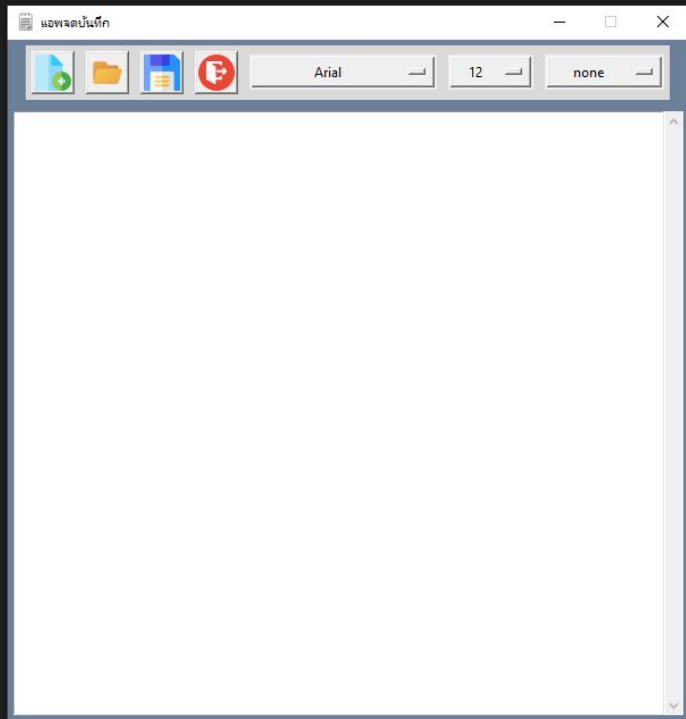
-

50

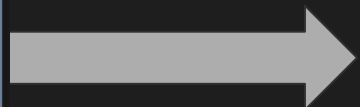


แอปจดบันทึก

# ออกแบบโปรแกรม



Menu Frame



Text Frame