



โครงการวิศวะไฟฟ้าสร้างสรรค์บริการวิชาการ

การตรวจจับข้อความในภาพ

รุ่นที่

โดยใช้ Python ร่วมกับ

Tesseract และ EasyOCR

1

วิทยากรโดย อ.ดร.โสภณ ผู้มีจรรยา และ อ.พรชัย เปี่ยมมทรพย์



วันอาทิตย์ที่ 21 เมษายน 2567

เวลา 09.30 - 16.30 น.



ภาควิชาวิศวกรรมไฟฟ้า

คณะวิศวกรรมศาสตร์และเทคโนโลยีอุตสาหกรรม

มหาวิทยาลัยศิลปากร

OCR : Optical Character Recognition



Tesseract OCR



ព័ត៌មាន ៖ Logo Python, Smartphone, tablet : <https://www.canva.com/> (PRO)

ព័ត៌មាន ៖ Logo EasyOCR : https://www.jaided.ai/static/img/svg_icon/EasyOCR_OSS3.svg

ព័ត៌មាន ៖ Logo Tesseract OCR : https://upload.wikimedia.org/wikipedia/commons/7/78/Tesseract_OCR_logo_%28Google%29.png?20190813175819

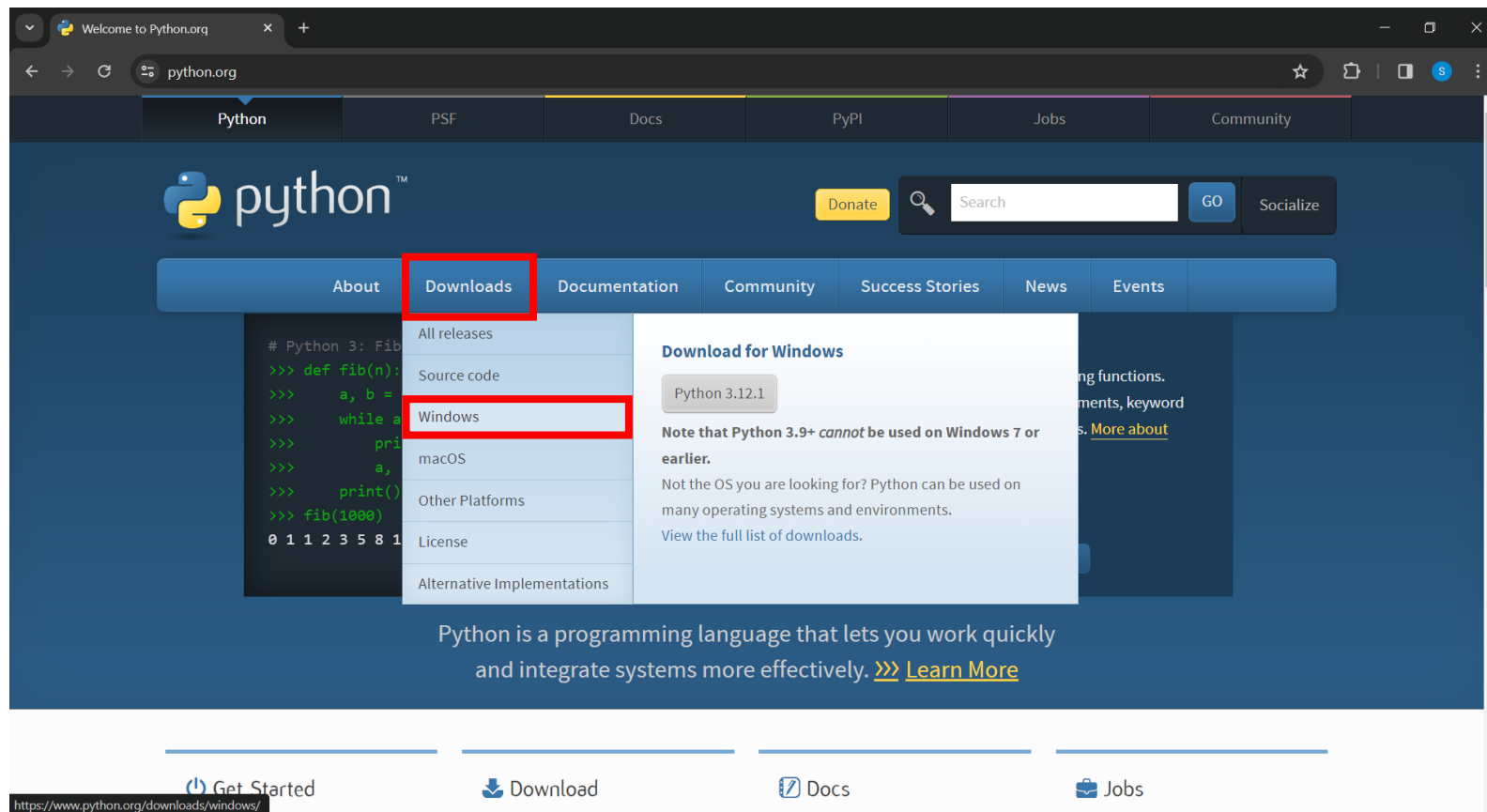
ព័ត៌មាន ៖ Logo Gemini : https://storage.googleapis.com/gweb-uniblog-publish-prod/images/final_keyword_header.width-1200.format-webp.webp

แนวทางการนำ OCR ไปประยุกต์ใช้งาน

- การตรวจจับข้อความในภาพ
- การค้นหาข้อมูลจากข้อความในภาพ
- การแปลภาษา
- การกรอกข้อมูลอัตโนมัติ
- การลบข้อความออกจากภาพ
- การสกัดข้อมูลจากเอกสาร Scan ใบเสร็จ หนังสือ
- การอ่านป้ายทะเบียนรถ
- การอ่านเลขมิเตอร์น้ำ ไฟ
- การอ่านป้ายบอกทาง ป้ายแจ้งเตือน ป้ายโฆษณา
- การอ่านข้อความในภาพเป็นเสียงพูด
- ฯลฯ

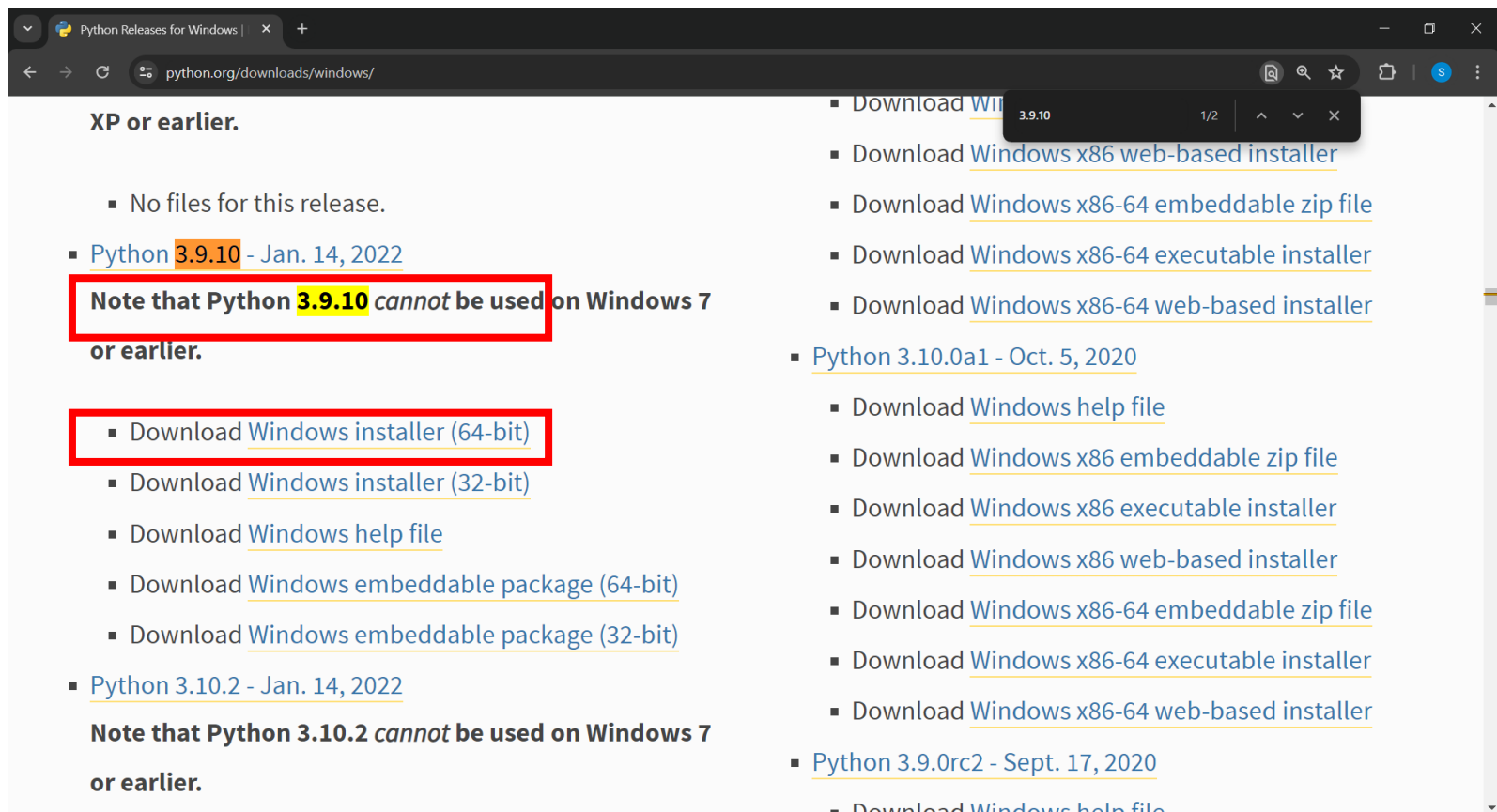
การติดตั้งโปรแกรม Python 3.9

เข้าไปดาวน์โหลดไฟล์ติดตั้งที่เว็บไซต์ <https://www.python.org>
โดยให้กดปุ่ม “Downloads” จากนั้น กดปุ่ม “Windows”



การติดตั้งโปรแกรม Python

จากนั้น ให้เลือกไฟล์ติดตั้ง โดยในตัวอย่างนี้เลือก **version 3.9.10** สำหรับ Windows 64 bit โดยคลิกที่ **Download Windows installer (64-bit)**



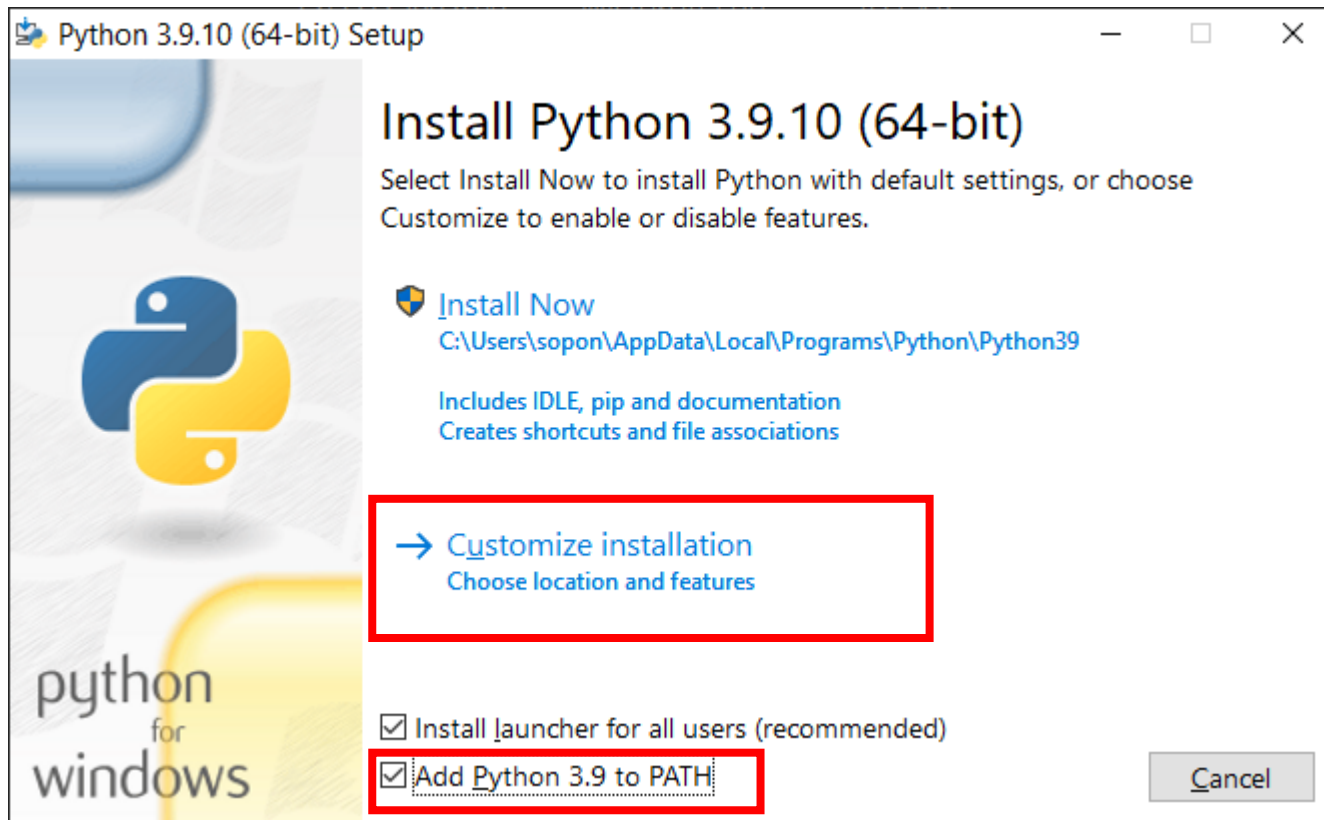
The screenshot shows the Python Releases for Windows page. The browser address bar displays 'python.org/downloads/windows/'. The page content includes a list of download links for various Python versions. The 'Python 3.9.10' section is highlighted, and the 'Download Windows installer (64-bit)' link is circled in red. A red box highlights a note: 'Note that Python 3.9.10 cannot be used on Windows 7 or earlier.' The 'Python 3.10.0a1' section is also visible, listing several download options.

- XP or earlier.
 - No files for this release.
 - Python 3.9.10 - Jan. 14, 2022
 - Note that Python 3.9.10 cannot be used on Windows 7 or earlier.
 - Download [Windows installer \(64-bit\)](#)
 - Download [Windows installer \(32-bit\)](#)
 - Download [Windows help file](#)
 - Download [Windows embeddable package \(64-bit\)](#)
 - Download [Windows embeddable package \(32-bit\)](#)
 - Python 3.10.2 - Jan. 14, 2022
 - Note that Python 3.10.2 cannot be used on Windows 7 or earlier.
- Python 3.9.10
 - Download [Windows help file](#)
 - Download [Windows x86 web-based installer](#)
 - Download [Windows x86-64 embeddable zip file](#)
 - Download [Windows x86-64 executable installer](#)
 - Download [Windows x86-64 web-based installer](#)
- Python 3.10.0a1 - Oct. 5, 2020
 - Download [Windows help file](#)
 - Download [Windows x86 embeddable zip file](#)
 - Download [Windows x86 executable installer](#)
 - Download [Windows x86 web-based installer](#)
 - Download [Windows x86-64 embeddable zip file](#)
 - Download [Windows x86-64 executable installer](#)
 - Download [Windows x86-64 web-based installer](#)
- Python 3.9.0rc2 - Sept. 17, 2020
 - Download [Windows help file](#)

การติดตั้งโปรแกรม Python

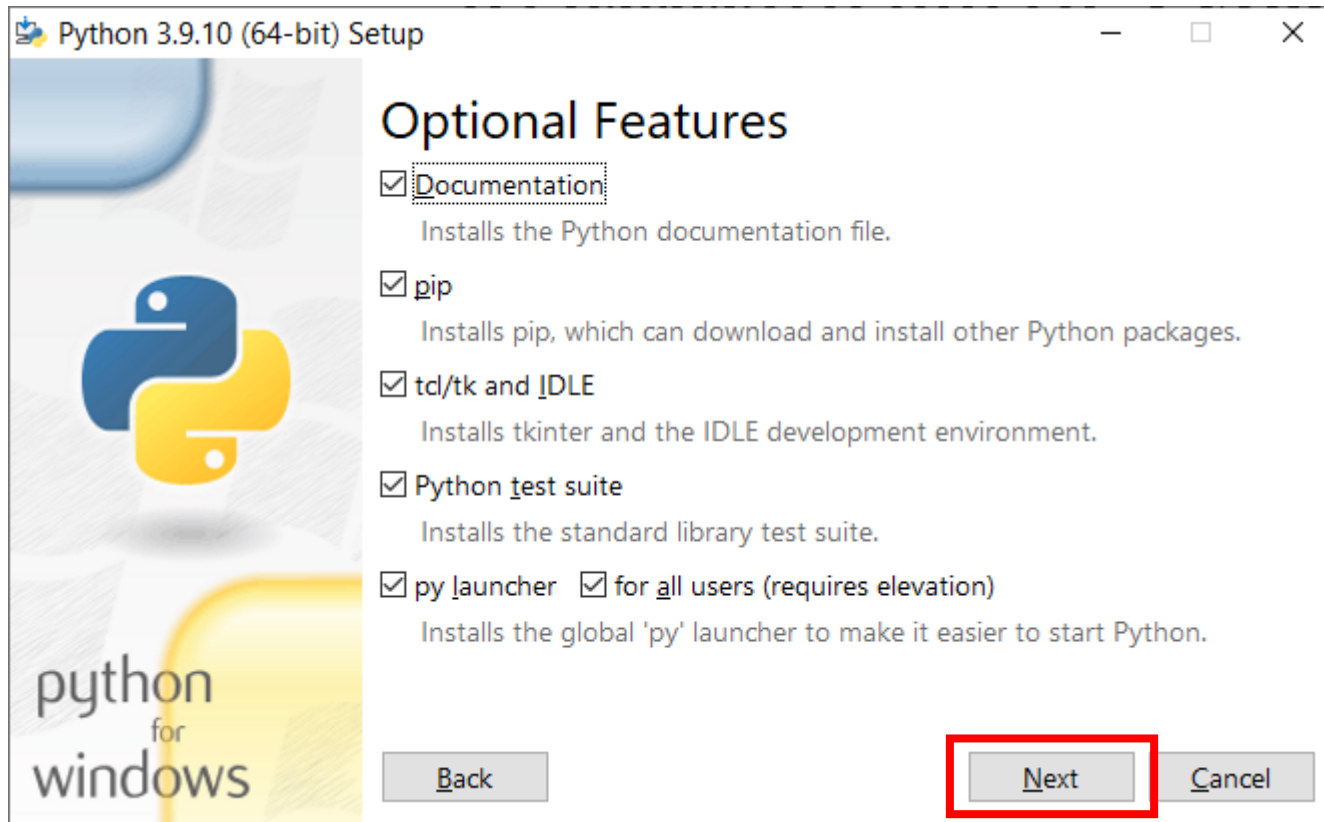
เมื่อดาวน์โหลดไฟล์ติดตั้งเสร็จแล้ว ให้กดดับเบิลคลิกที่ไฟล์ติดตั้ง
ที่ได้ดาวน์โหลดมา python-3.9.10-amd64.exe

จากนั้น ให้คลิกเลือก “Add Python 3.9 to PATH” แล้วคลิกที่ “Customize installation”



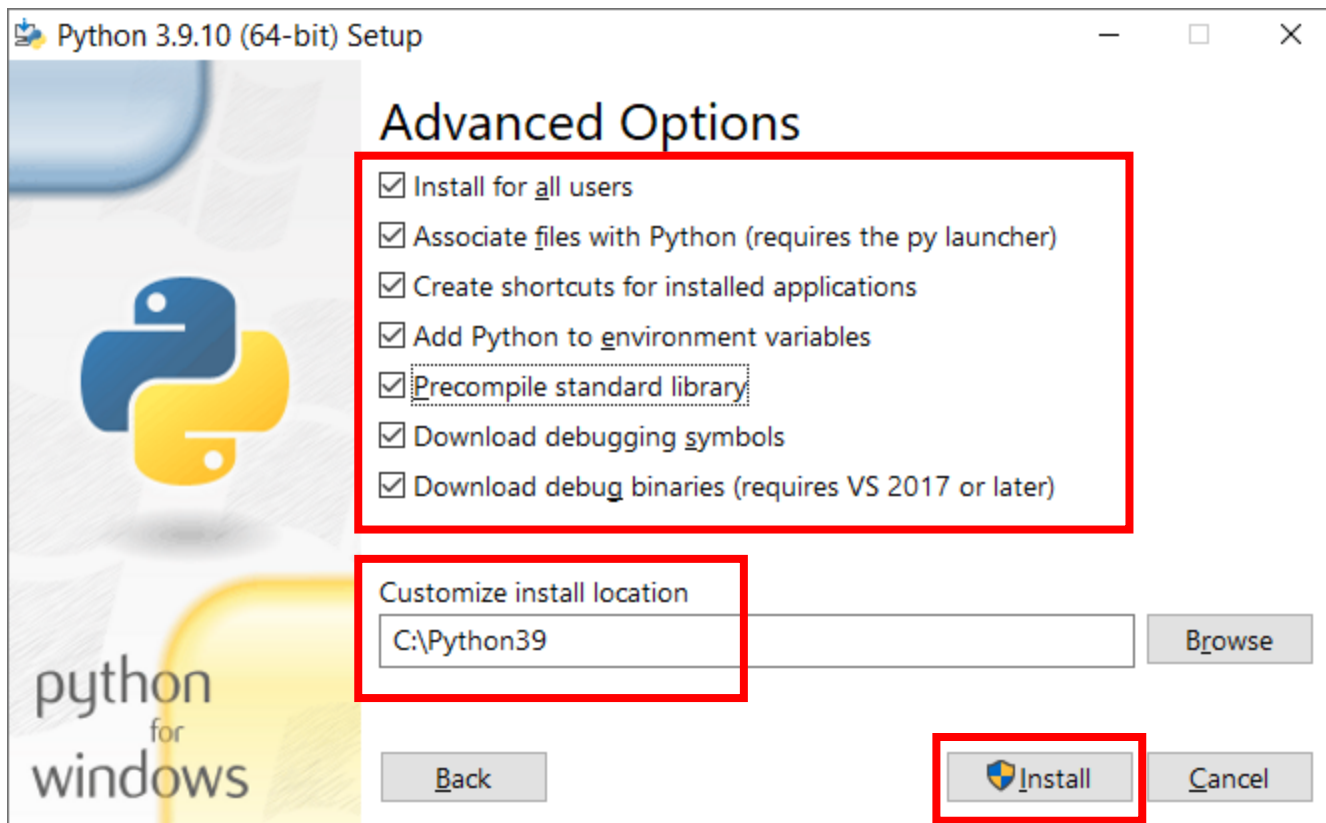
การติดตั้งโปรแกรม Python

จากนั้น ให้กดปุ่ม “Next”



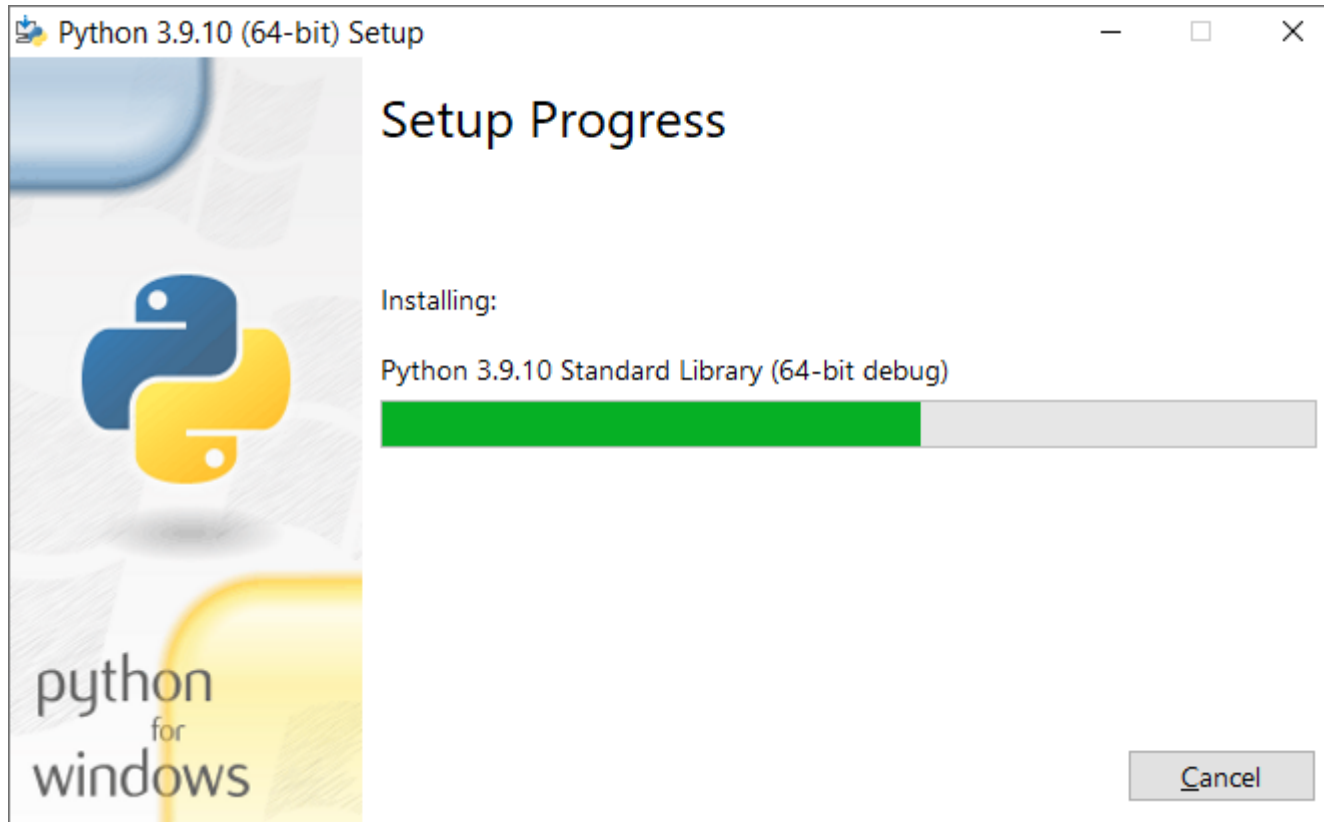
การติดตั้งโปรแกรม Python

จากนั้น ให้ติ๊กเลือกทุก Option แล้วให้กำหนด location เป็น C:\Python39 แล้วให้คลิก “Install”



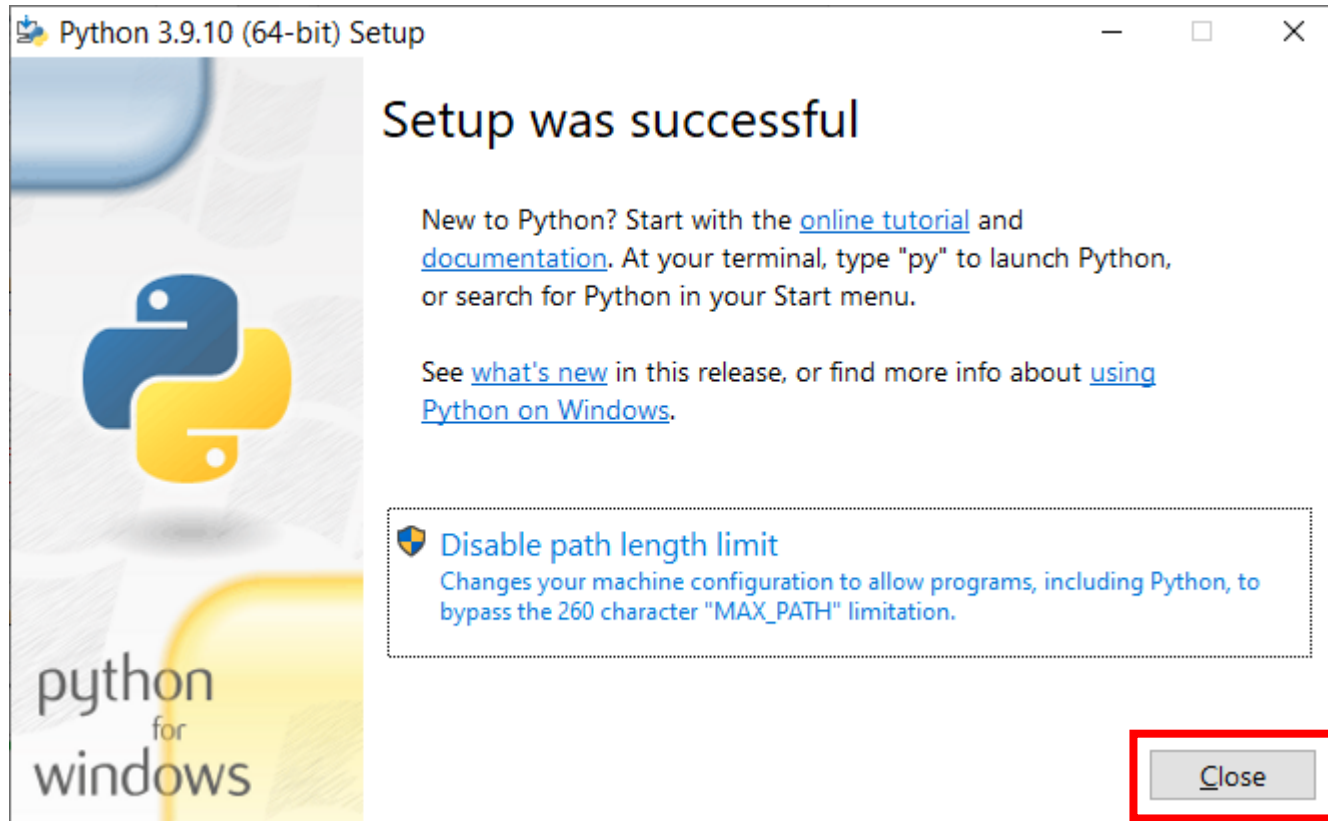
การติดตั้งโปรแกรม Python

จากนั้น ให้รอกันกว่าจะติดตั้งเสร็จ



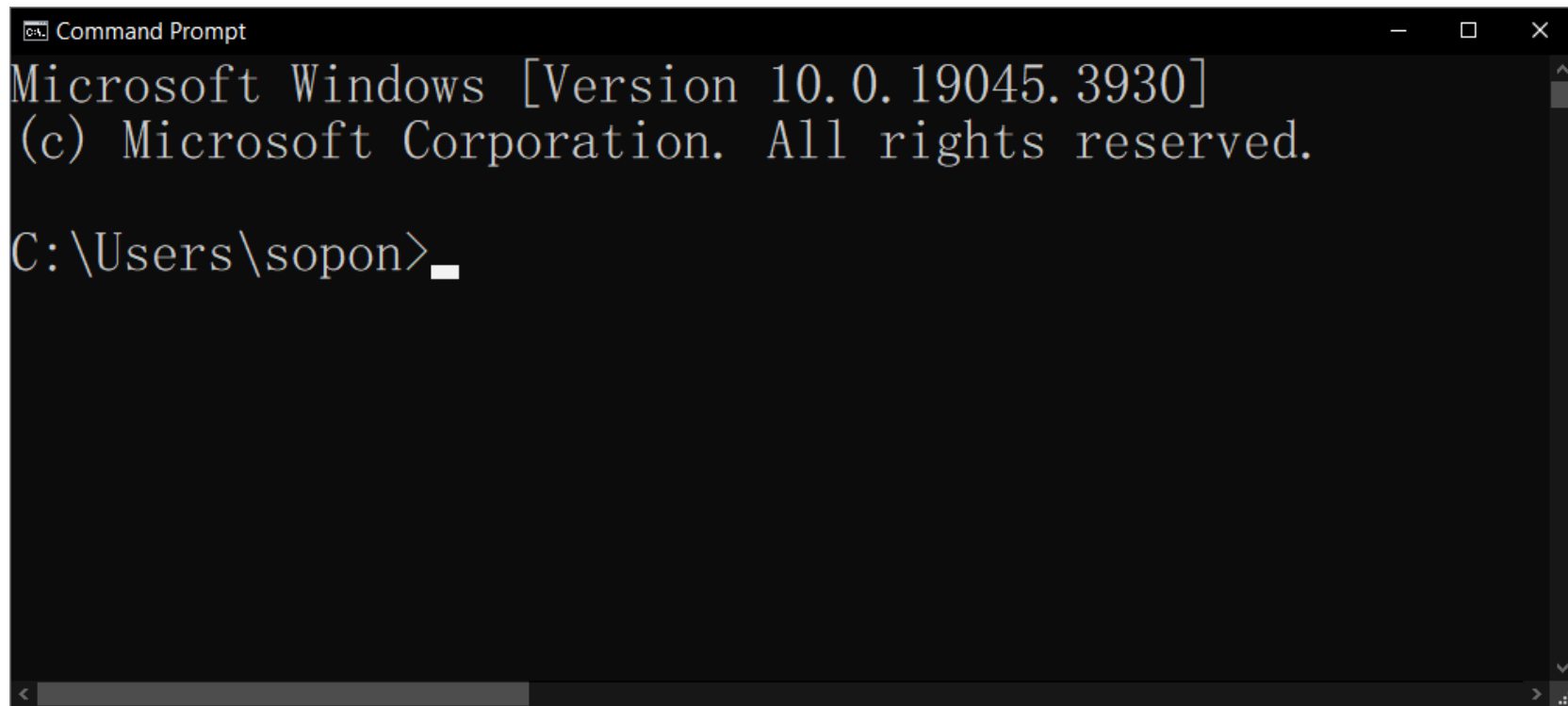
การติดตั้งโปรแกรม Python

เมื่อติดตั้งเสร็จสิ้นแล้ว ให้กดปุ่ม “Close” เป็นอันเสร็จสิ้นขั้นตอนการติดตั้ง Python



การติดตั้งไลบรารีโดยใช้คำสั่ง pip

ให้เข้าไปที่ `cmd` โดยการกดที่ปุ่ม `start` แล้วพิมพ์ `cmd` ในช่องค้นหา แล้วกด `Enter` จะปรากฏหน้าต่าง `cmd` ดังรูป

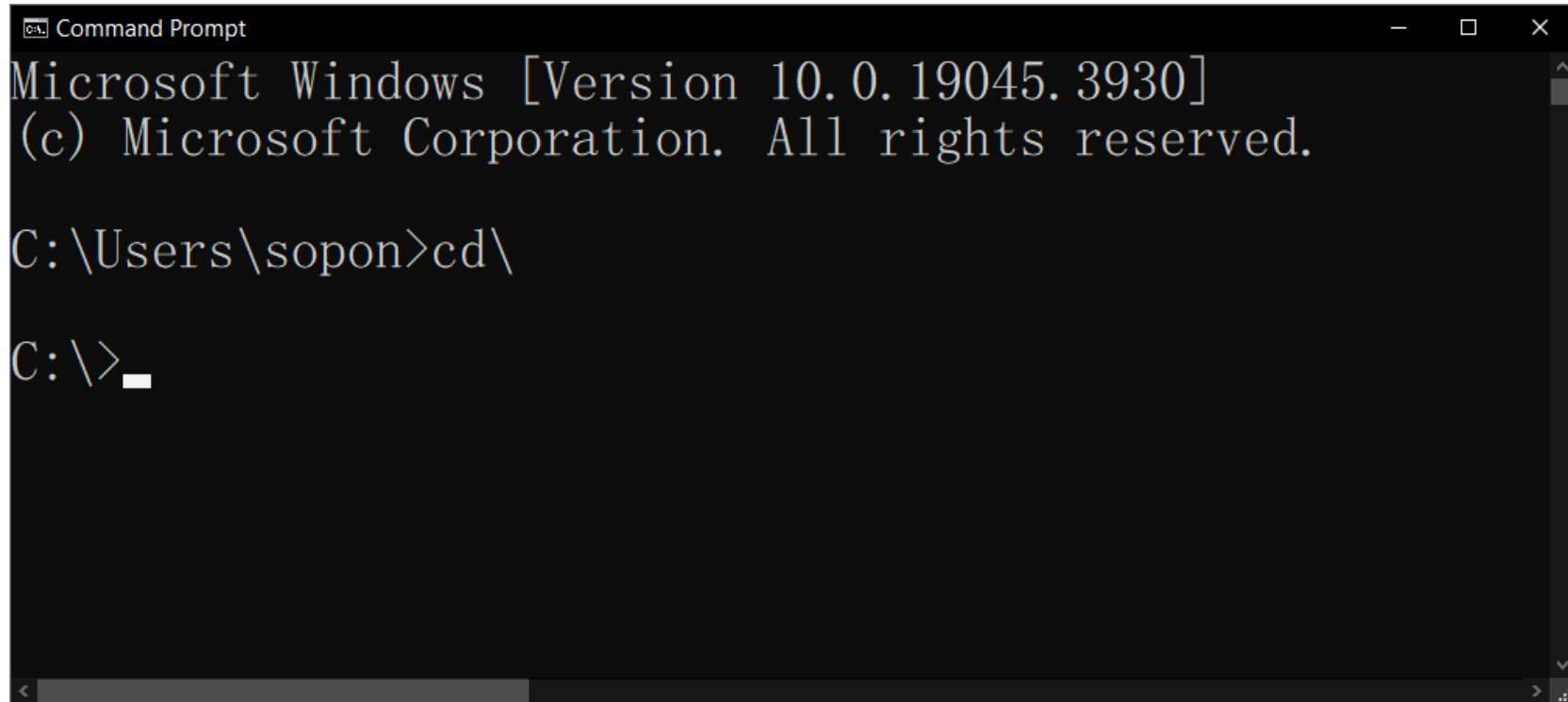


```
Command Prompt
Microsoft Windows [Version 10.0.19045.3930]
(c) Microsoft Corporation. All rights reserved.

C:\Users\sopon>
```

การติดตั้งไลบรารีโดยใช้คำสั่ง pip

ให้พิมพ์คำสั่ง `cd\` เพื่อออกมาที่ `C:\>`



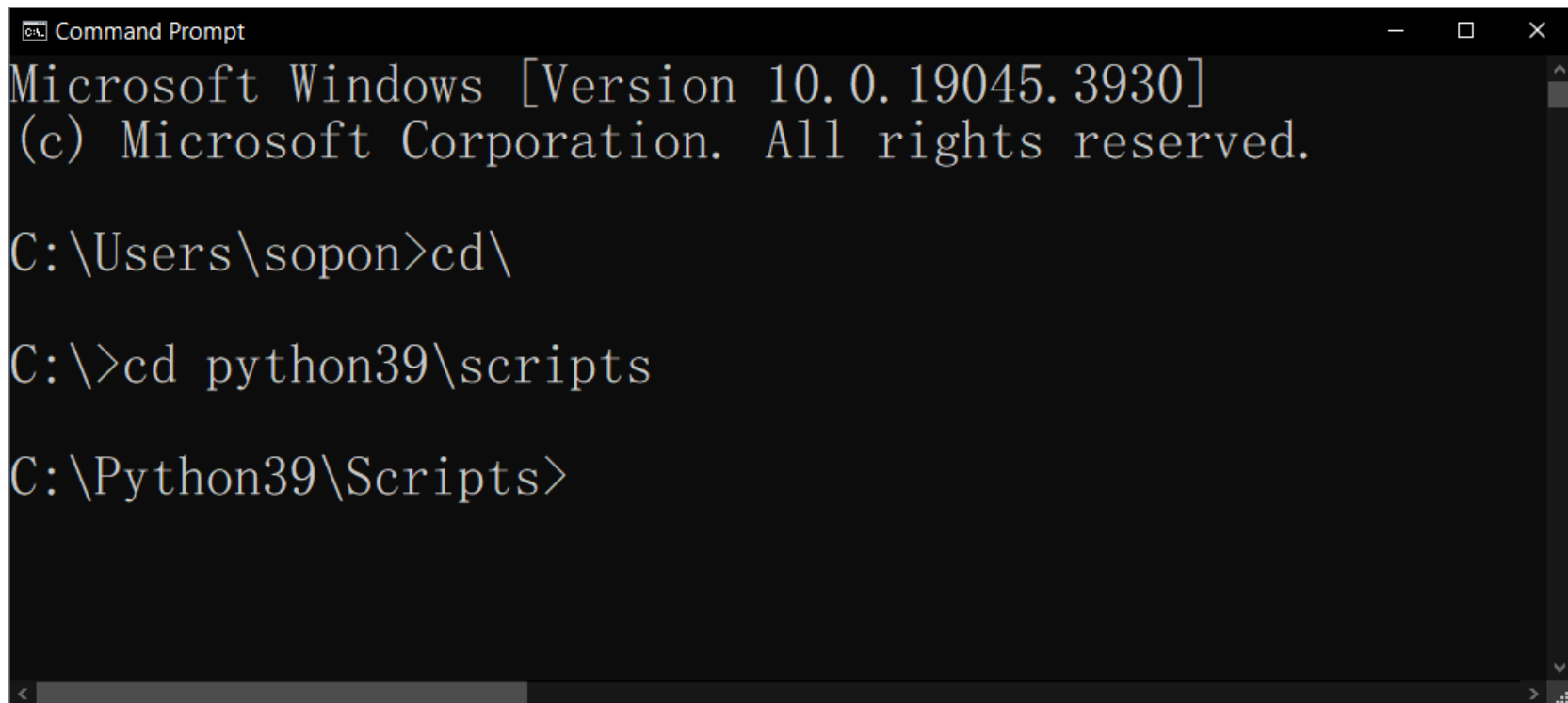
```
Command Prompt
Microsoft Windows [Version 10.0.19045.3930]
(c) Microsoft Corporation. All rights reserved.

C:\Users\sopon>cd\

C:\>_
```

การติดตั้งไลบรารีโดยใช้คำสั่ง pip

จากนั้น ให้พิมพ์คำสั่ง `cd python39\scripts` เพื่อเข้าไปที่ `C:\Python39\Scripts` ซึ่งเป็นที่อยู่ของคำสั่ง `pip`



```
Command Prompt
Microsoft Windows [Version 10.0.19045.3930]
(c) Microsoft Corporation. All rights reserved.

C:\Users\sopon>cd\

C:\>cd python39\scripts

C:\Python39\Scripts>
```

การติดตั้งไลบรารีที่ใช้ในการอบรม

ที่จอ **cmd** จากนั้น ให้ทำการติดตั้งไลบรารีต่าง ๆ ดังนี้

ชื่อไลบรารี	การติดตั้งโดยใช้ pip	หน้าที่การทำงาน
google.generativeai	pip install google-generativeai	เชื่อมต่อกับ Gemini
matplotlib	pip install matplotlib	พล็อตกราฟ แสดงภาพ
opencv	pip install opencv-contrib-python	ประมวลผลภาพ
pytesseract	pip install pytesseract	Tesseract OCR แปลงภาพเป็นข้อความ
easyocr	pip install easyocr	EasyOCR แปลงภาพเป็นข้อความ
pdf2image	pip install pdf2image	แปลงไฟล์ pdf เป็นภาพ
pygame	pip install pygame	สร้างเกม เล่นไฟล์เสียง

การติดตั้งไลบรารีที่ใช้ในการอบรม (ต่อ)

ชื่อไลบรารี	การติดตั้งโดยใช้ pip	หน้าที่การทำงาน
gTTS	pip install gTTS	แปลงข้อความเป็นเสียงพูด
pyaudio	##for python 3.9 pip install full path\PyAudio-0.2.11-cp39-cp39-win_amd64.whl ##for python 3.10 pip install full path\PyAudio-0.2.11-cp310-cp310-win_amd64.whl หมายเหตุ: ต้องใส่ที่อยู่ (full path ของ file .whl)	การจัดการเสียง

เมื่อติดตั้งจนครบทุกอันแล้ว สามารถใช้คำสั่ง **pip list** ในการตรวจสอบการติดตั้งไลบรารีได้

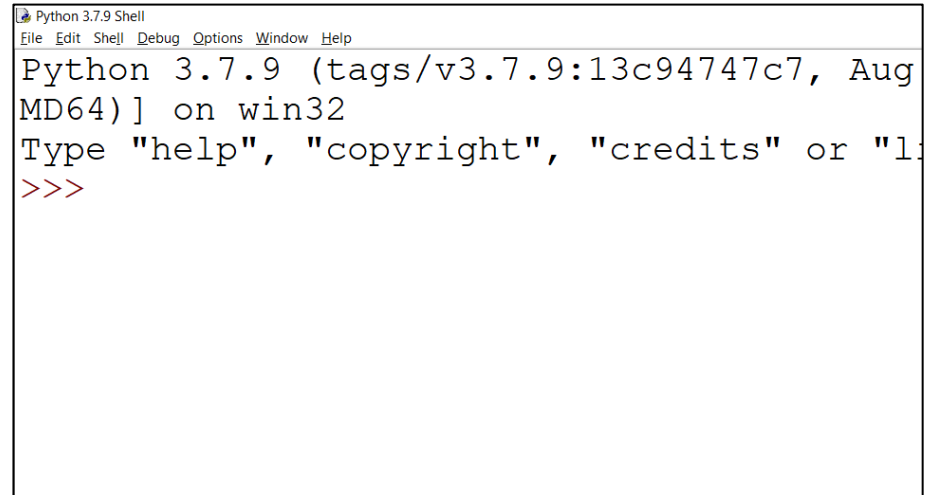
เครื่องมือในการเขียนโปรแกรมภาษา Python

- **Python IDLE**
- **Eclipse PyDev**
- **Notepad++**
- **ActiveState Komodo IDE**
- **PyCharm**
- **Jupyter Notebook**
- **Google Colab**
- **Visual Studio Code**
- **<https://repl.it/languages/python3>**

Python IDLE

Shell Window (Python Shell)

ใช้ในการเขียนโปรแกรมแบบที่ละบรรทัด
และใช้ในการแสดงผลรันโปรแกรม



```
Python 3.7.9 Shell
File Edit Shell Debug Options Window Help
Python 3.7.9 (tags/v3.7.9:13c94747c7, Aug
MD64)] on win32
Type "help", "copyright", "credits" or "license()" for more
>>>
```

Edit Window

ใช้ในการเขียนโค้ดโปรแกรมแบบยาว
แล้วรันโค้ดทีเดียวโดยการกด F5



```
*untitled*
File Edit Format Run Options Window Help
1 a = 5
2 b = 4
3 c = a + b
4 print(c)
5
```

ทบทวนการเขียนโปรแกรมภาษา Python

- **boolean**
- **integer**
- **float**
- **string**
- **list**
- **dict**
- **if**
- **for**
- **while**
- **try except**
- **import**

ข้อมูลชนิด Boolean

Code:

```
a = True
b = False
print(a)
print(b)
print(type(a)) # คำสั่ง type ใช้สำหรับดูชนิดข้อมูล
print(type(b))
```

Run:

```
True
False
<class 'bool'>
<class 'bool'>
```

ข้อมูลชนิด Boolean

Code:

```
a = 5 > 2
print(a)
print(type(a))
b = 4 < 3
print(b)
print(type(b))
```

Run:

```
True
<class 'bool'>
False
<class 'bool'>
```

ข้อมูลชนิด Integer (เลขจำนวนเต็ม)

Code:

```
x = 10
y = -3
print(x)
print(y)
print(type(x))
print(type(y))
```

Run:

```
10
-3
<class 'int'>
<class 'int'>
```

ข้อมูลชนิด Float (เลขทศนิยม)

Code:

```
x = 3.0
y = -4.56
z = 0.0
print(x)
print(y)
print(z)
print(type(x))
print(type(y))
print(type(z))
```

Run:

```
3.0
-4.56
0.0
<class 'float'>
<class 'float'>
<class 'float'>
```

ข้อมูลชนิด Float (เลขทศนิยม)

Code:

```
a = 3.14546789
print(format(a, '.0f')) # แสดงทศนิยม 0ตำแหน่ง
print(format(a, '.1f')) # แสดงทศนิยม 1ตำแหน่ง
print(format(a, '.2f')) # แสดงทศนิยม 2ตำแหน่ง
print(format(a, '.3f')) # แสดงทศนิยม 3ตำแหน่ง
print(format(a, '.4f')) # แสดงทศนิยม 4ตำแหน่ง
print(format(a, '.5f')) # แสดงทศนิยม 5ตำแหน่ง
print(format(a, '.6f')) # แสดงทศนิยม 6ตำแหน่ง
print(format(a, '.7f')) # แสดงทศนิยม 7ตำแหน่ง
print(format(a, '.8f')) # แสดงทศนิยม 8ตำแหน่ง
print(format(a, '.9f')) # แสดงทศนิยม 9ตำแหน่ง
print(format(a, '.10f')) # แสดงทศนิยม 10ตำแหน่ง
```

Run:

```
3
3.1
3.15
3.145
3.1455
3.14547
3.145468
3.1454679
3.14546789
3.145467890
3.1454678900
```

ข้อมูลชนิด String (ข้อความ)

Code:

```
x = 'A'  
y = "A"  
z = 'Hello Python'  
w = "Hello Computer"  
k = ''  
print(x)  
print(y)  
print(z)  
print(w)  
print(k)  
print(type(x))  
print(type(y))  
print(type(z))  
print(type(w))  
print(type(k))
```

Run:

```
A  
A  
Hello Python  
Hello Computer  
  
<class 'str'>  
<class 'str'>  
<class 'str'>  
<class 'str'>  
<class 'str'>
```

ข้อมูลชนิด String (ข้อความ)

Code:

```
x = "Computer"  
print(x[0])  
print(x[1])  
print(x[2])  
print(x[3])  
print(x[4])  
print(x[5])  
print(x[6])  
print(x[7])  
#print(x[8]) #string index out of range
```

Run:

```
C  
o  
m  
p  
u  
t  
e  
r
```

ข้อมูลชนิด String (ข้อความ)

Code:

```
a = "Google"  
b = "Python"  
c = a + b  
print(c)
```

Run:

```
GooglePython
```

Code:

```
a = "Hello Computer"  
b = len(a) # นับจำนวนตัวอักษร  
print(b)
```

Run:

```
14
```

ข้อมูลชนิด List []

Code:

```
x = []  
y = [1,4.5,-15]  
z = ['ant','cat','dog']  
print(x)  
print(y)  
print(z)  
print(type(x))  
print(type(y))  
print(type(z))
```

Run:

```
[]  
[1, 4.5, -15]  
['ant', 'cat', 'dog']  
<class 'list'>  
<class 'list'>  
<class 'list'>
```

ข้อมูลชนิด List []

Code:

```
x = [1,3,4,5.6]
y = ['ant', 'cat', 'dog']
print(x[0])
print(y[1])
```

Run:

```
1
cat
```

Code:

```
x = ['ant',1,3.4]
print(x)
```

Run:

```
['ant', 1, 3.4]
```

ข้อมูลชนิด Dictionary {k,v}

Code:

```
x = {0:'ant',1:'cat',2:'dog'}  
print(type(x))
```

Run:

```
<class 'dict'>
```

Code:

```
x = {4:'ant',7:'cat',9:'dog'}  
print(x[4])  
print(x[9])
```

Run:

```
ant  
dog
```

การทำตามเงื่อนไข if

Code:

```
x = 3
if (x > 2):
    print("Yes")
else:
    print("No")
```

Run:

Yes

Code:

30

```
score = 80
if (score > 70):
    print("A")
elif (score > 60):
    print("B")
elif (score > 50):
    print("C")
else:
    print("F")
```

Run:

A

การทำตามเงื่อนไข if

Code:

```
# if ซ้อน if
x = 40
if (x > 0):
    if (x < 50):
        print("One")
    else:
        print("Two")
else:
    print("Three")
```

Run:

One

การวนรอบทำซ้ำ for

- `for i in range (start,stop,step)` วน start ถึง (stop-1) ครั้งละ step
- `for i in range (stop)` จะมี start = 0, step = 1
- `for i in range (start,stop)` จะมี step = 1

Code:

```
for i in range(5): # 0 ถึง (5-1)
    print(i)
```

Run:

```
0
1
2
3
4
```

การวนรอบทำซ้ำ for

- `for i in range (start,stop,step)` วน start ถึง (stop-1) ครั้งละ step
- `for i in range (stop)` จะมี start = 0, step = 1
- `for i in range (start,stop)` จะมี step = 1

Code:

```
for i in range(2,6):  
    print(i)
```

Run:

```
2  
3  
4  
5
```

Code:

```
# วน for ใน list  
x = [1,3,5,7,100]  
for i in x:  
    print(i)
```

Run:

```
1  
3  
5  
7  
100
```

การวนรอบทำซ้ำ while

- วนลูป เมื่อเงื่อนไขเป็นจริง
- ไม่วนลูป เมื่อเงื่อนไขเป็นเท็จ
- ออกจากลูปเมื่อเจอคำสั่ง break

Code:

```
i = 0
while (i < 10):
    print(i)
    i = i + 1
print("Bye")
```

Run:

```
0
1
2
3
4
5
6
7
8
9
Bye
```

การวนรอบทำซ้ำ while

- วนลูป เมื่อเงื่อนไขเป็นจริง
- ไม่วนลูป เมื่อเงื่อนไขเป็นเท็จ
- ออกจากลูปเมื่อเจอคำสั่ง break

Code:

```
i = 0
while (i < 10):
    print(i)
    if (i==5):
        break
    i = i +1
print("Bye")
```

Run:

```
0
1
2
3
4
5
Bye
```

การจัดการข้อผิดพลาด โดยใช้ try except

Code:

```
try:  
    a = 30  
    b = a + c  
    print(b)  
except:  
    pass # ข้ามผ่านไปเลยเมื่อเกิด error ขึ้น  
  
print("hello")
```

Run:

```
hello
```

ข้อสังเกต : โปรแกรมจะรันต่อไปได้ โดยไม่สนใจ error ที่เกิดขึ้น
โปรแกรมจะไม่เกิดการหยุด run กลางคันสามารถ run ต่อไปได้อจนจบ

การจัดการข้อผิดพลาด โดยใช้ try except

Code:

```
a = 30
b = a + c
print(b)
print("hello")
```

Run:

```
Traceback (most recent call last):
  File "C:/Users/sopon/Desktop/testerror.py", line 2, in <module>
    b = a + c
NameError: name 'c' is not defined
```

ข้อสังเกต : โปรแกรมจะ error รันต่อไปไม่ได้ เนื่องจากไม่รู้จักตัวแปร c

Code:

```
try:
    a = 30
    b = a + c
    print(b)
except:
    print("ตรวจพบ error")

print("hello")
```

Run:

```
ตรวจพบ error
hello
```

ข้อสังเกต : โปรแกรมจะรันต่อไปได้จนถึง print("hello")

การจัดการข้อผิดพลาด โดยใช้ try except

Code:

```
try:  
    a = 30  
    b = a + c  
    print(b)  
except:  
    pass # ข้ามผ่านไปเลยเมื่อเกิด error ขึ้น  
  
print("hello")
```

Run:

```
hello
```

ข้อสังเกต : โปรแกรมจะรันต่อไปได้ โดยไม่สนใจ error ที่เกิดขึ้น
โปรแกรมจะไม่เกิดการหยุด run กลางคันสามารถ run ต่อไปได้จนจบ

การ import โมดูล

Code:

```
import math
# นำเข้าทุกฟังก์ชันของโมดูล math
print(dir(math))
# คำสั่ง dir ใช้ดูว่าโมดูลนี้ข้างในมีอะไรให้เรียกใช้งานได้บ้าง
```

Run:

```
['__doc__', '__loader__', '__name__', '__package__', '__spec__', 'acos', 'acosh', 'asin', 'asinh', 'atan', 'atan2', 'atanh', 'ceil', 'copysign', 'cos', 'cosh', 'degrees', 'e', 'erf', 'erfc', 'exp', 'expm1', 'fabs', 'factorial', 'floor', 'fmod', 'frexp', 'fsum', 'gamma', 'gcd', 'hypot', 'inf', 'isclose', 'isfinite', 'isinf', 'isnan', 'ldexp', 'lgamma', 'log', 'log10', 'log1p', 'log2', 'modf', 'nan', 'pi', 'pow', 'radians', 'sin', 'sinh', 'sqrt', 'tan', 'tanh', 'tau', 'trunc']
```

การ import โมดูล

Code:

```
import math
# นำเข้าทุกฟังก์ชันของโมดูล math
x = math.pow(2,3)
y = math.sqrt(4)
print(x)
print(y)
```

Run:

```
8.0
2.0
```

Code:

```
from math import pow
# นำเข้าแค่ฟังก์ชัน pow ของโมดูล math
x = pow(2,3)
print(x)
```

Run:

```
8.0
```

การ import โมดูล

Code:

```
from math import pow
x = pow(2,3)
y = sqrt(9)    # จะ error เนื่องจาก
# ไม่รู้จักฟังก์ชัน sqrt เพราะไม่ได้นำเข้ามา
print(x)
print(y)
```

Run:

```
error
```

Code:

```
from math import *    # นำเข้าทุกฟังก์ชันของโมดูล math
x = pow(2,3)
# เรียกชื่อฟังก์ชัน pow ได้เลยโดยไม่ต้องพิมพ์ math.pow
y = sqrt(81)
z = exp(2)
print(x)
print(y)
print(z)
```

Run:

```
8.0
9.0
7.38905609893065
```

การ import โมดูล

Code:

```
# การใช้ as เพื่อให้ชื่อโมดูลสั้นลง
import math as mt
x = mt.sqrt(9)
y = mt.exp(3)
print(x)
print(y)
```

Run:

```
3.0
20.085536923187668
```

Code:

```
# การใช้ as เพื่อให้ชื่อฟังก์ชันในโมดูลสั้นลง
from math import sqrt as sq
x = sq(9)
y = sq(4)
print(x)
print(y)
```

Run:

```
3.0
2.0
```

โค้ดโปรแกรมตัวอย่าง Pillow

ชื่อไฟล์	โค้ดโปรแกรม
pillow001.py	การเปิดไฟล์ภาพด้วย Pillow และแสดงภาพด้วย matplotlib
pillow002.py	การวาดรูปหลายเหลี่ยมลงบนภาพ
pillow003.py	การบันทึกภาพ

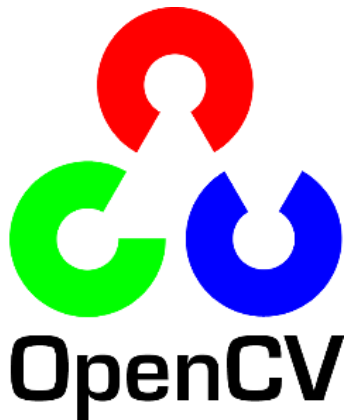


โค้ดโปรแกรมตัวอย่าง OpenCV

ชื่อไฟล์	โค้ดโปรแกรม
opencv001.py	การเปิดไฟล์ภาพด้วย OpenCV และแสดงภาพด้วย matplotlib
opencv002.py	การวาดกรอบสี่เหลี่ยมลงบนภาพ
opencv003.py	การใส่ตัวอักษรลงบนภาพ (ภาษาอังกฤษ)
opencv004.py	การบันทึกภาพ



รายชื่อ font ใน OpenCV



FONT_HERSHEY_SIMPLEX = 0

FONT_HERSHEY_PLAIN = 1

FONT_HERSHEY_DUPLEX = 2

FONT_HERSHEY_COMPLEX = 3

FONT_HERSHEY_TRIPLEX = 4

FONT_HERSHEY_COMPLEX_SMALL = 5

FONT_HERSHEY_SCRIPT_SIMPLEX = 6

FONT_HERSHEY_SCRIPT_COMPLEX = 7

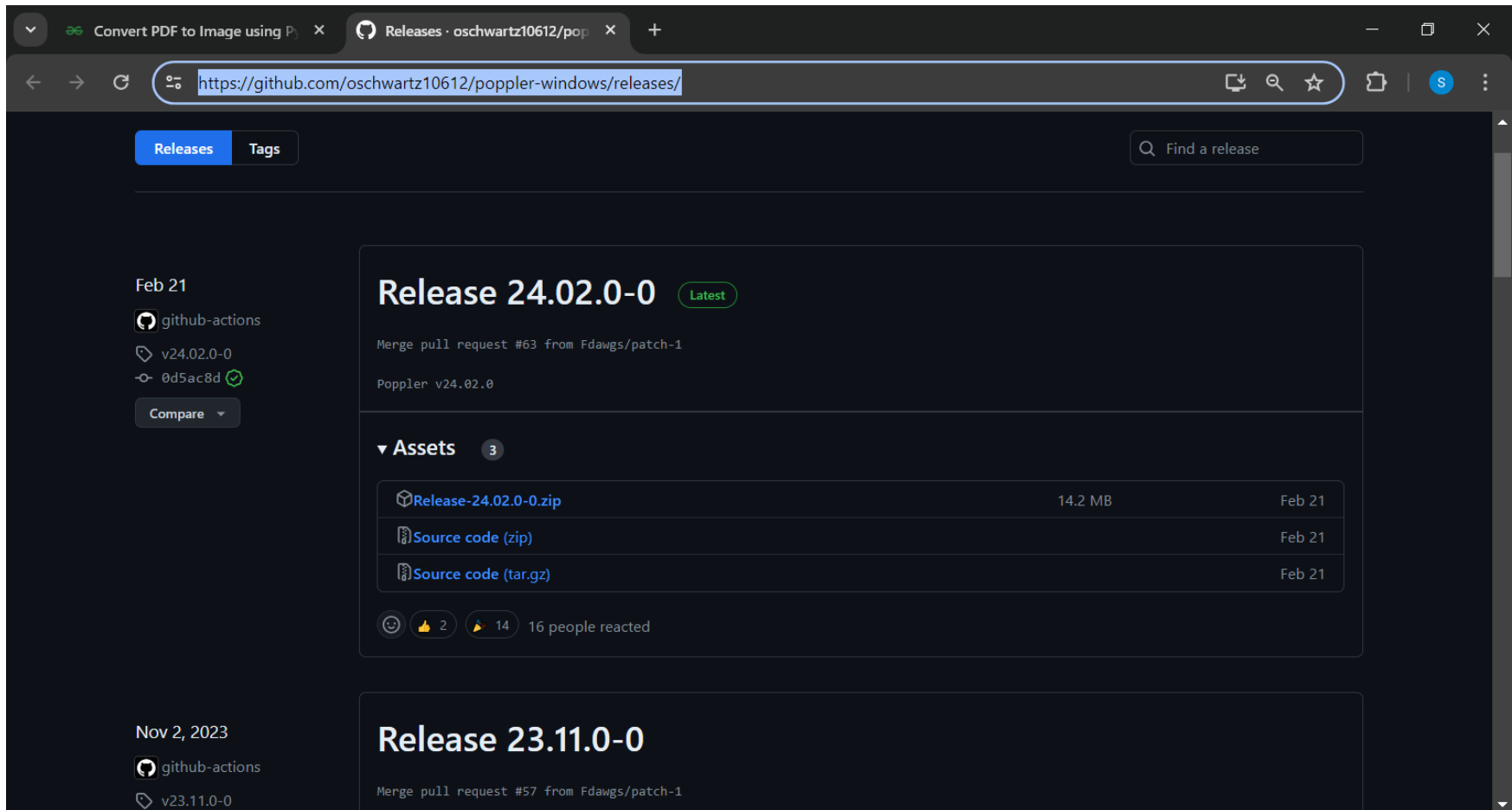
โค้ดโปรแกรมตัวอย่าง Pygame

ชื่อไฟล์	โค้ดโปรแกรม
pygame001.py	การสร้าง Display Surface
pygame002.py	การเปลี่ยนสี Background
pygame003.py	การวาดเส้นตรง วงกลม และ สีเหลี่ยม
pygame004.py	การวาดรูปหลายเหลี่ยม
pygame005.py	การใส่ตัวอักษร
pygame006.py	การใส่ภาพลงบน Display Surface
pygame007.py	การใส่ภาพลงบน Display Surface และใส่ตัวอักษร
pygame008.py	การใส่ Object



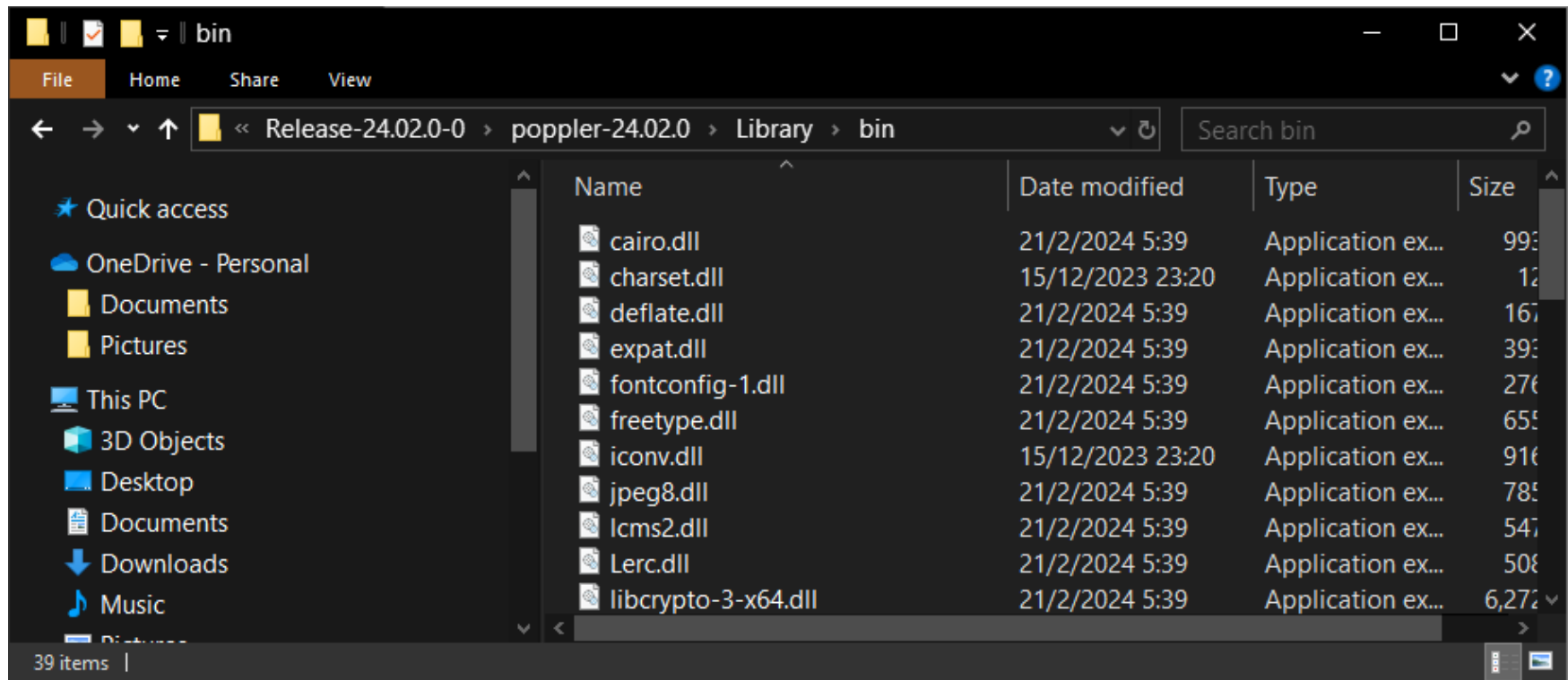
การแปลง pdf เป็น image

1. ให้ไปที่เว็บไซต์ <https://github.com/oschwartz10612/poppler-windows/releases> เพื่อดาวน์โหลดไฟล์ **Release-24.02.0-0.zip**



การแปลง pdf เป็น image

2. ให้แตกไฟล์ **Release-24.02.0-0.zip** ออกมา พร้อมทั้ง copy path เต็มของโฟลเดอร์ bin ไว้ใช้งาน เช่น **D:\Release-24.02.0-0\poppler-24.02.0\Library\bin**



การแปลง pdf เป็น image

3. นำ path เต็มของ โฟลเดอร์ bin มาใส่ในโค้ด pdf2image001.py

```
1 from pdf2image import convert_from_path
2
3 images = convert_from_path('pdf002.pdf',
4                             poppler_path = r'D:\Release-24.02.0-0\poppler-24.02.0\Library\bin')
5
6 for i in range(len(images)):
7     images[i].save('page'+ str(i) + '.jpg', 'JPEG')
8
9 print('แปลงไฟล์เรียบร้อยแล้ว')
```

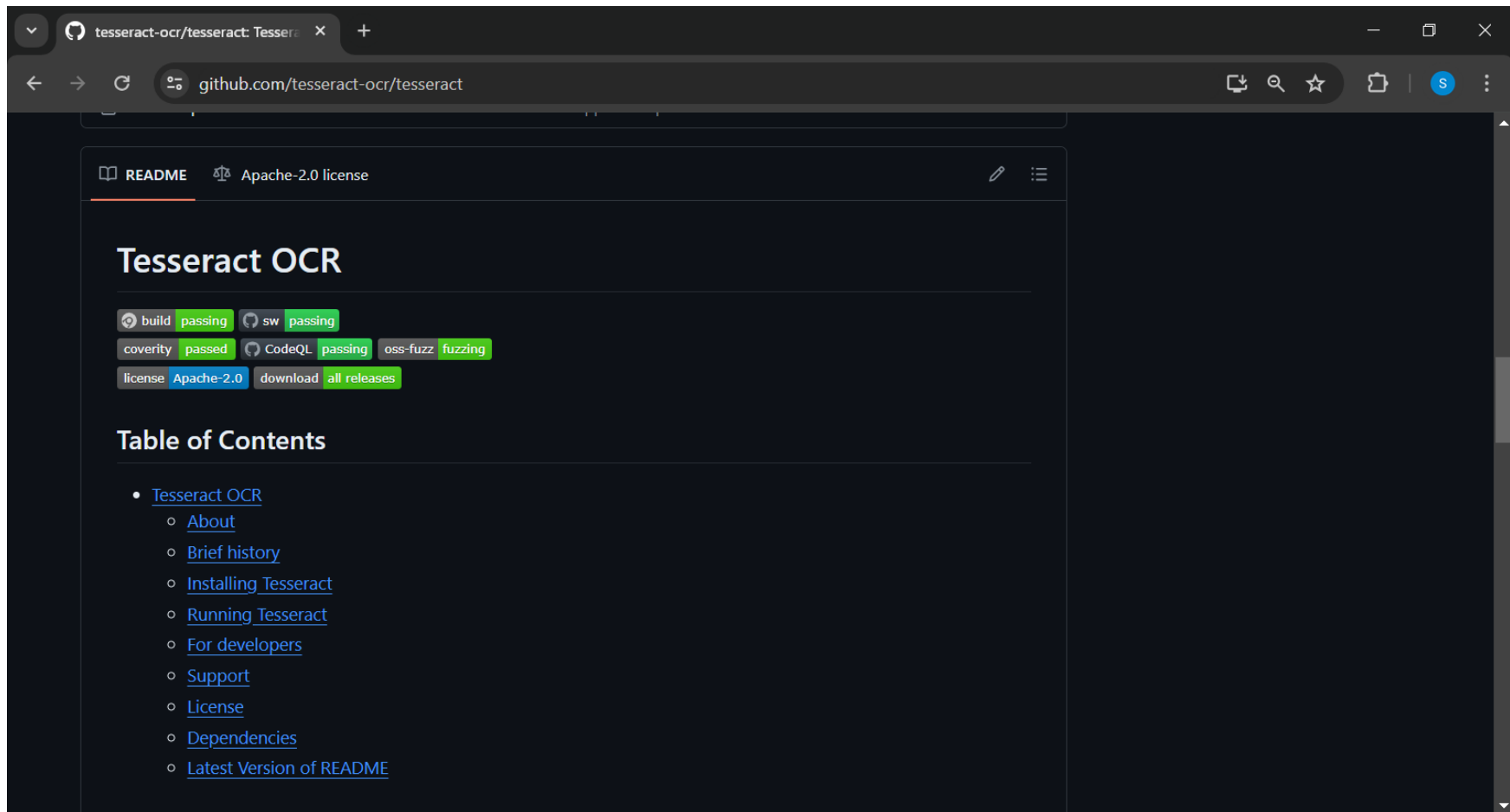
โค้ดโปรแกรมตัวอย่าง pdf2image

ชื่อไฟล์	โค้ดโปรแกรม
pdftoimage001.py	การแปลงไฟล์ pdf เป็น image
pdftoimage002.py	การแปลงไฟล์ pdf เป็น image และแสดงภาพ (หน้าแรกหน้าเดียว)

Tesseract OCR

Tesseract OCR

เว็บไซต์ Tesseract OCR → <https://github.com/tesseract-ocr/tesseract>



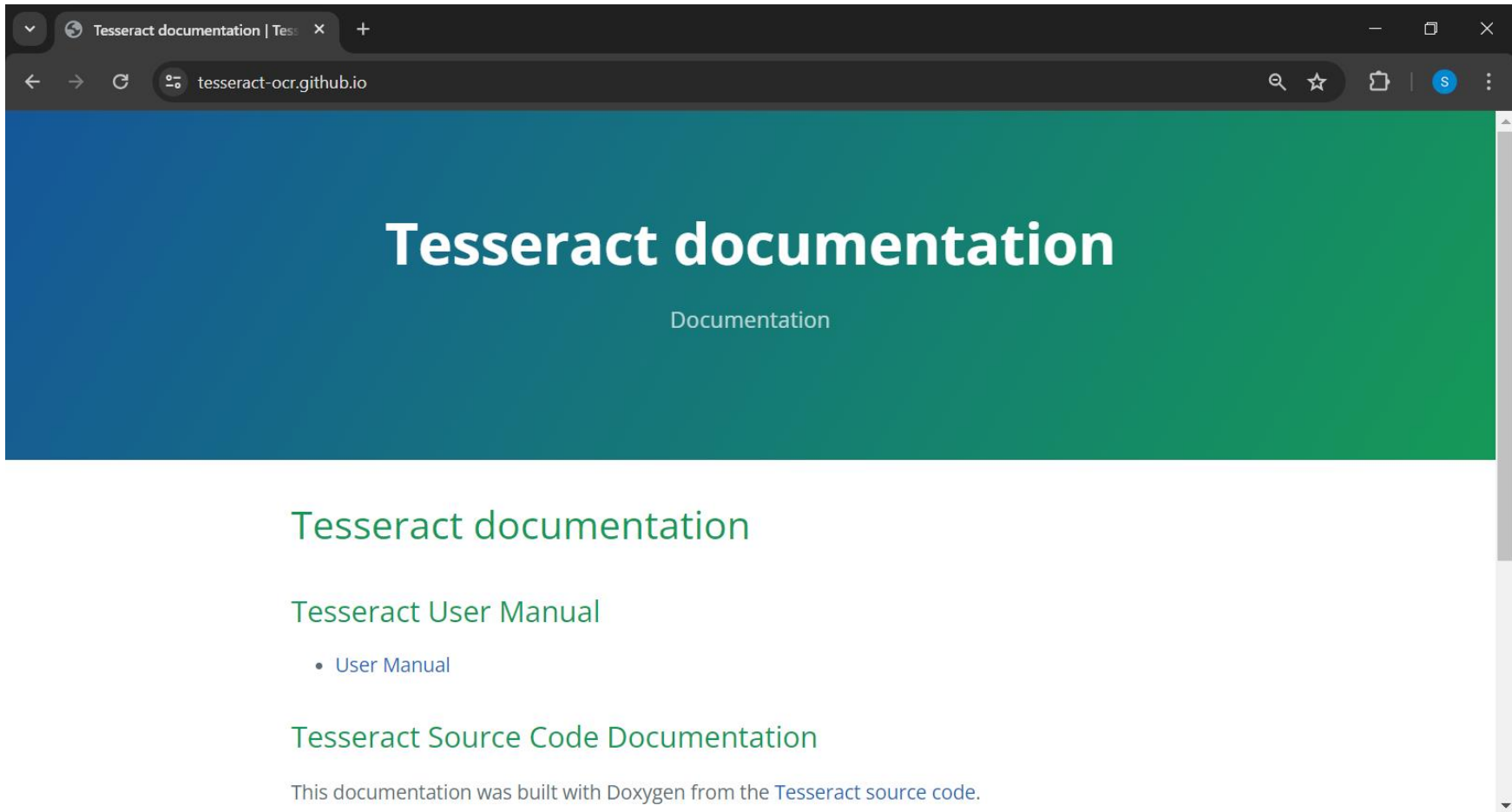
The screenshot shows a web browser displaying the GitHub repository page for Tesseract OCR. The page title is "Tesseract OCR" and it is licensed under Apache-2.0. The page features a "Table of Contents" section with the following links:

- [Tesseract OCR](#)
 - [About](#)
 - [Brief history](#)
 - [Installing Tesseract](#)
 - [Running Tesseract](#)
 - [For developers](#)
 - [Support](#)
 - [License](#)
 - [Dependencies](#)
 - [Latest Version of README](#)

Additionally, the page displays several status badges: "build passing", "sw passing", "coverity passed", "CodeQL passing", "oss-fuzz fuzzing", "license Apache-2.0", and "download all releases".

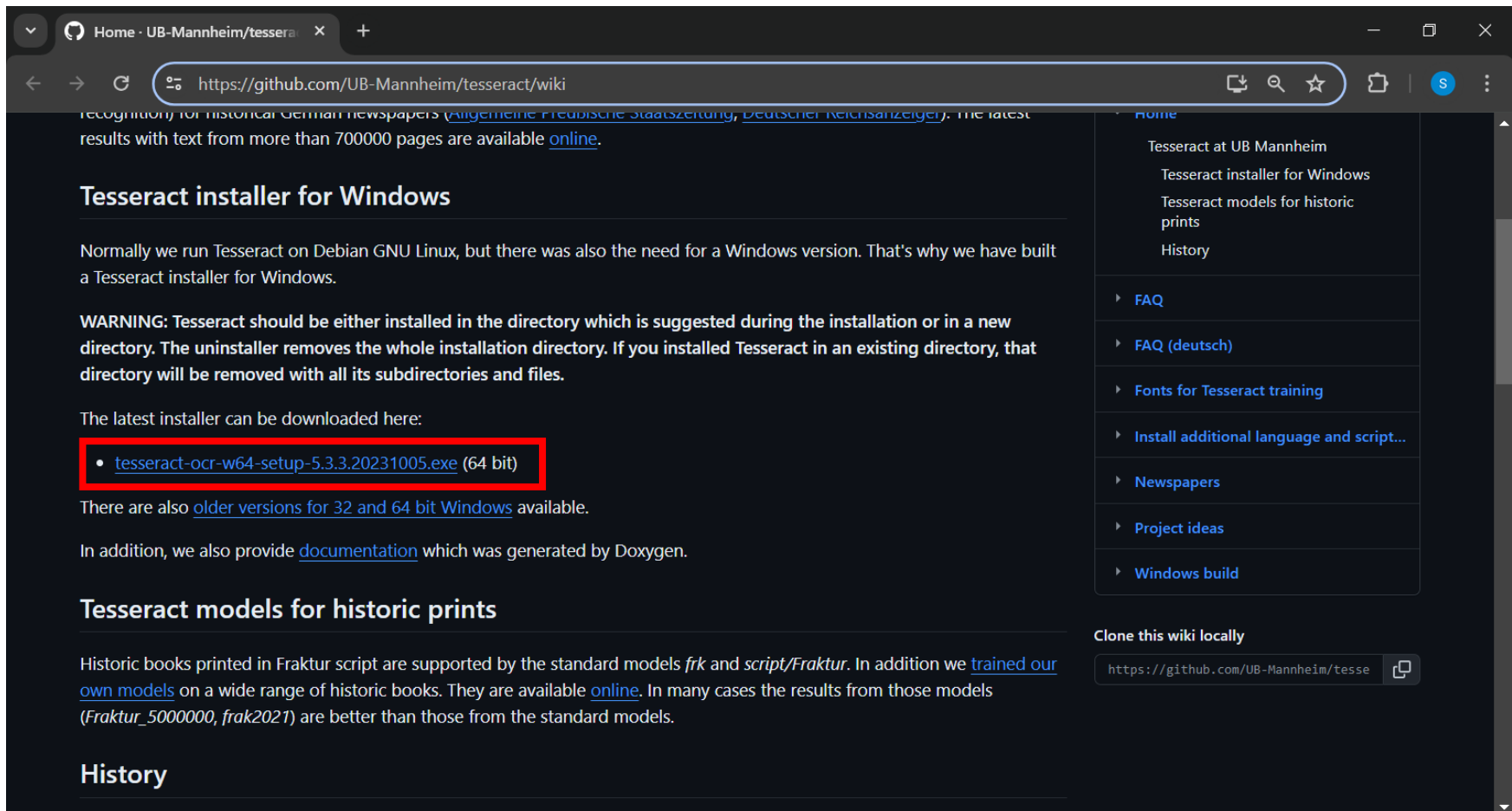
Tesseract OCR

เว็บไซต์ Tesseract OCR → <https://tesseract-ocr.github.io>



การติดตั้ง Tesseract OCR

1. เข้าไปที่เว็บไซต์ <https://github.com/UB-Mannheim/tesseract/wiki> เพื่อดาวน์โหลดไฟล์ติดตั้ง **tesseract-ocr-w64-setup-5.3.3.20231005.exe**



The screenshot shows a web browser window displaying the GitHub Wiki page for Tesseract OCR. The page title is "Tesseract installer for Windows". The content includes a warning about installation directories and a list of download links. The link for "tesseract-ocr-w64-setup-5.3.3.20231005.exe (64 bit)" is highlighted with a red box. The right sidebar contains a navigation menu with links to "FAQ", "FAQ (deutsch)", "Fonts for Tesseract training", "Install additional language and script...", "Newspapers", "Project ideas", and "Windows build".

recognizing for historical German newspapers (*Allgemeine Preussische Staatszeitung, Deutscher Reichsanzeiger*): the latest results with text from more than 700000 pages are available [online](#).

Tesseract installer for Windows

Normally we run Tesseract on Debian GNU Linux, but there was also the need for a Windows version. That's why we have built a Tesseract installer for Windows.

WARNING: Tesseract should be either installed in the directory which is suggested during the installation or in a new directory. The uninstaller removes the whole installation directory. If you installed Tesseract in an existing directory, that directory will be removed with all its subdirectories and files.

The latest installer can be downloaded here:

- [tesseract-ocr-w64-setup-5.3.3.20231005.exe \(64 bit\)](#)

There are also [older versions](#) for 32 and 64 bit Windows available.

In addition, we also provide [documentation](#) which was generated by Doxygen.

Tesseract models for historic prints

Historic books printed in Fraktur script are supported by the standard models *frk* and *script/Fraktur*. In addition we [trained our own models](#) on a wide range of historic books. They are available [online](#). In many cases the results from those models (*Fraktur_5000000*, *frak2021*) are better than those from the standard models.

History

Home

- Tesseract at UB Mannheim
- Tesseract installer for Windows
- Tesseract models for historic prints
- History

▶ [FAQ](#)

▶ [FAQ \(deutsch\)](#)

▶ [Fonts for Tesseract training](#)

▶ [Install additional language and script...](#)

▶ [Newspapers](#)

▶ [Project ideas](#)

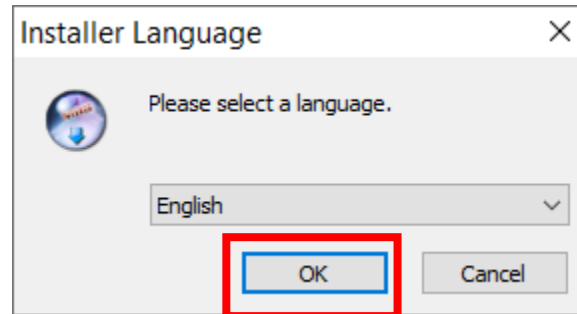
▶ [Windows build](#)

Clone this wiki locally

<https://github.com/UB-Mannheim/tesseract/wiki>

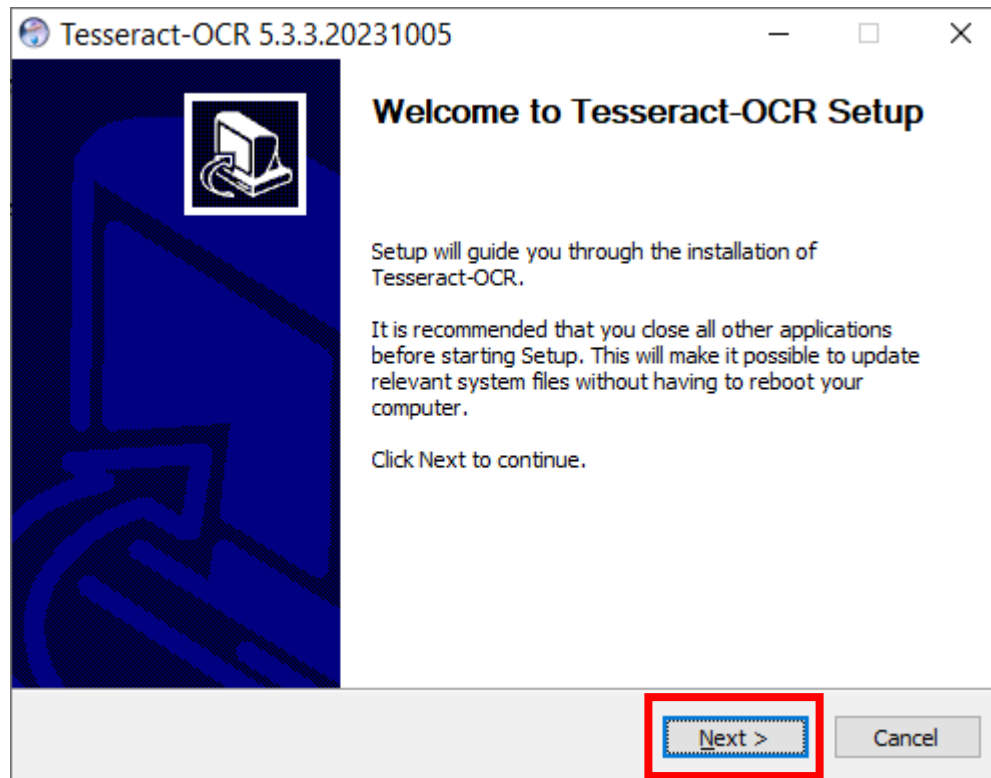
การติดตั้ง Tesseract OCR

2. ให้ดับเบิลคลิกที่ไฟล์ `tesseract-ocr-w64-setup-5.3.3.20231005.exe`
จากนั้น เลือกภาษา ในที่นี้เลือกเป็น English



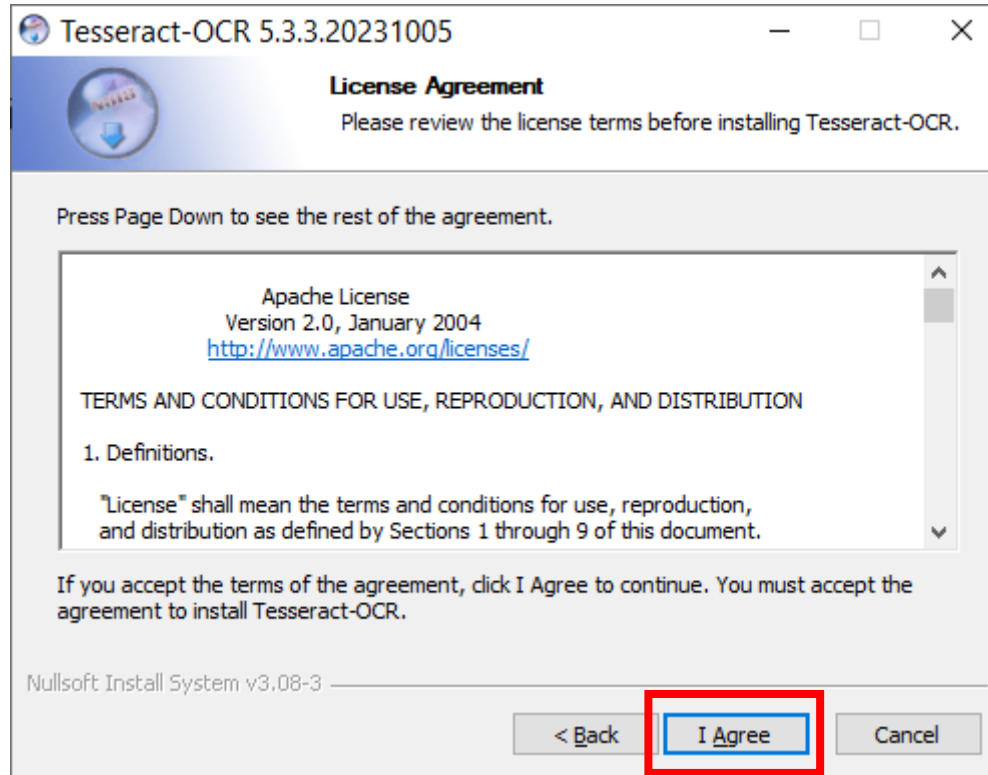
การติดตั้ง Tesseract OCR

3. จากนั้น กดปุ่ม Next



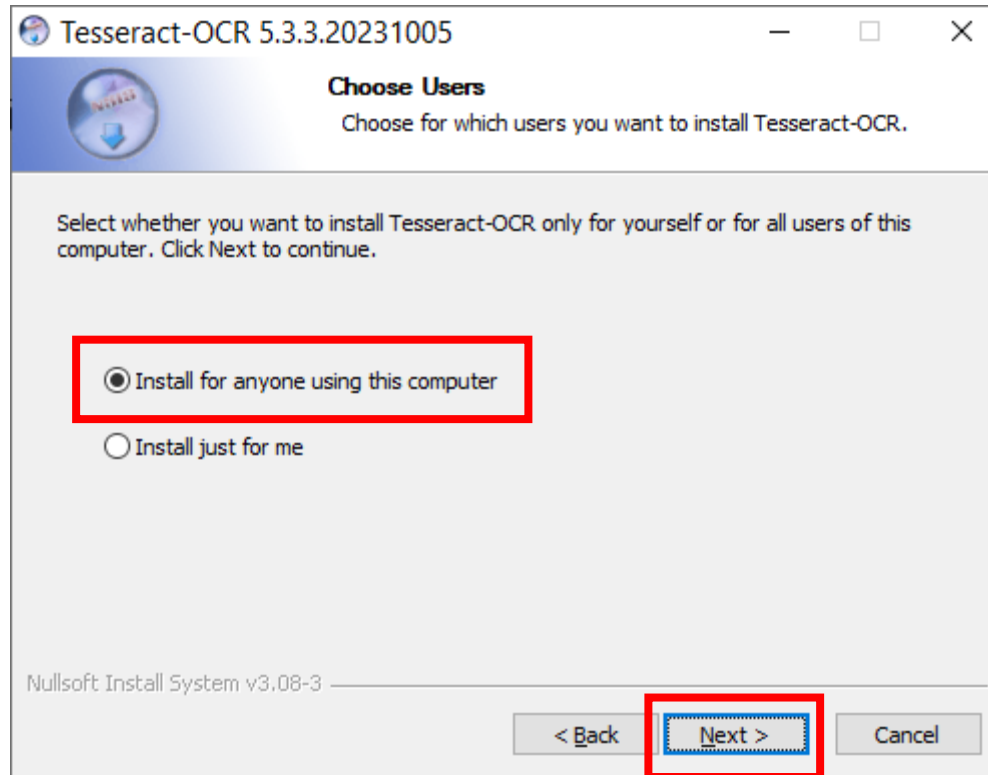
การติดตั้ง Tesseract OCR

4. จากนั้น กดปุ่ม I Agree



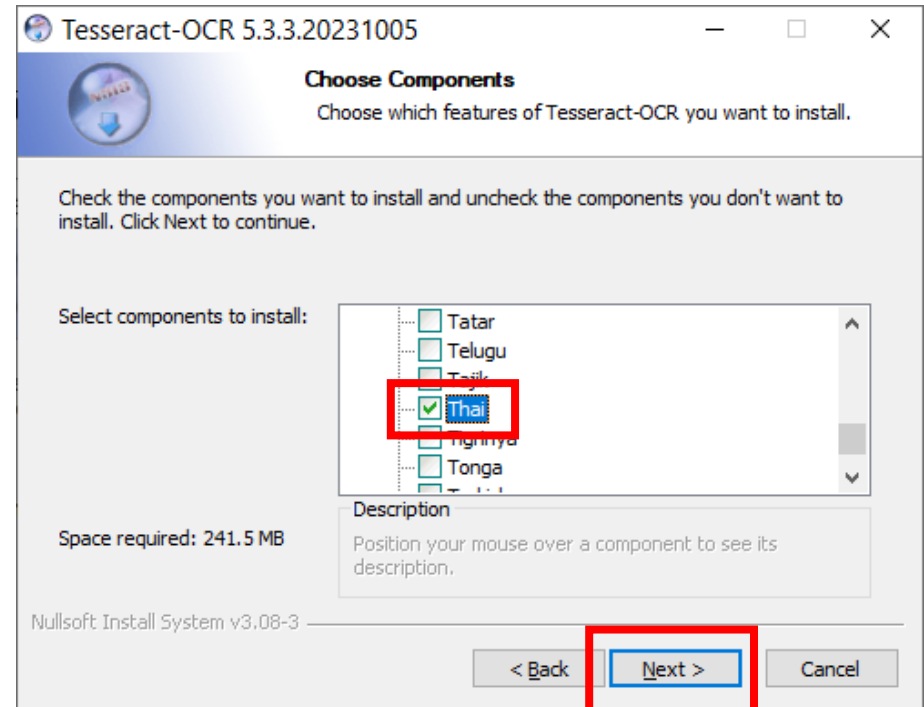
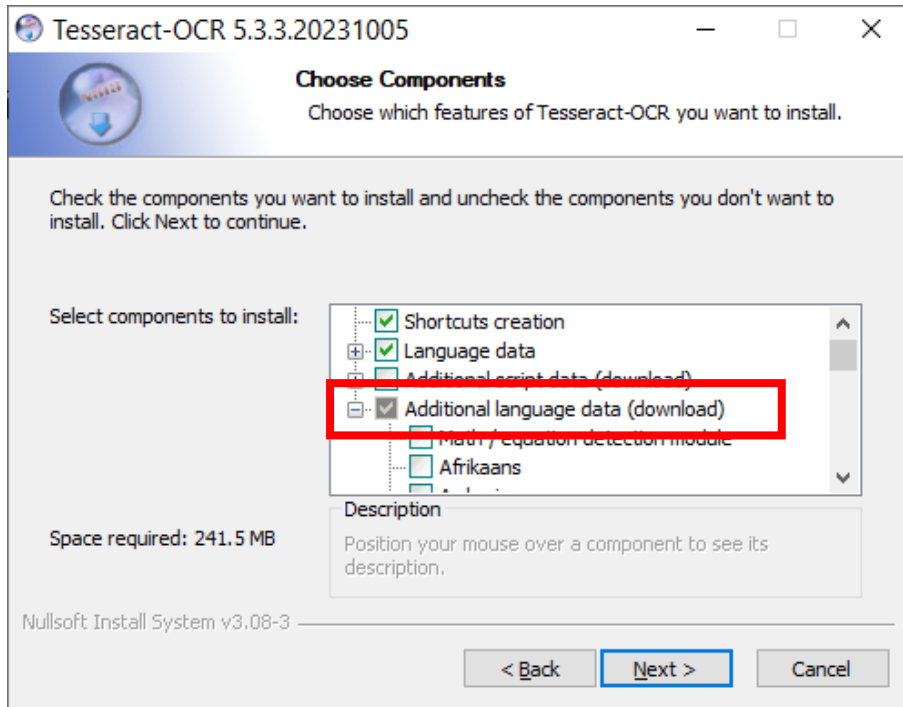
การติดตั้ง Tesseract OCR

5. จากนั้น กดปุ่ม Next



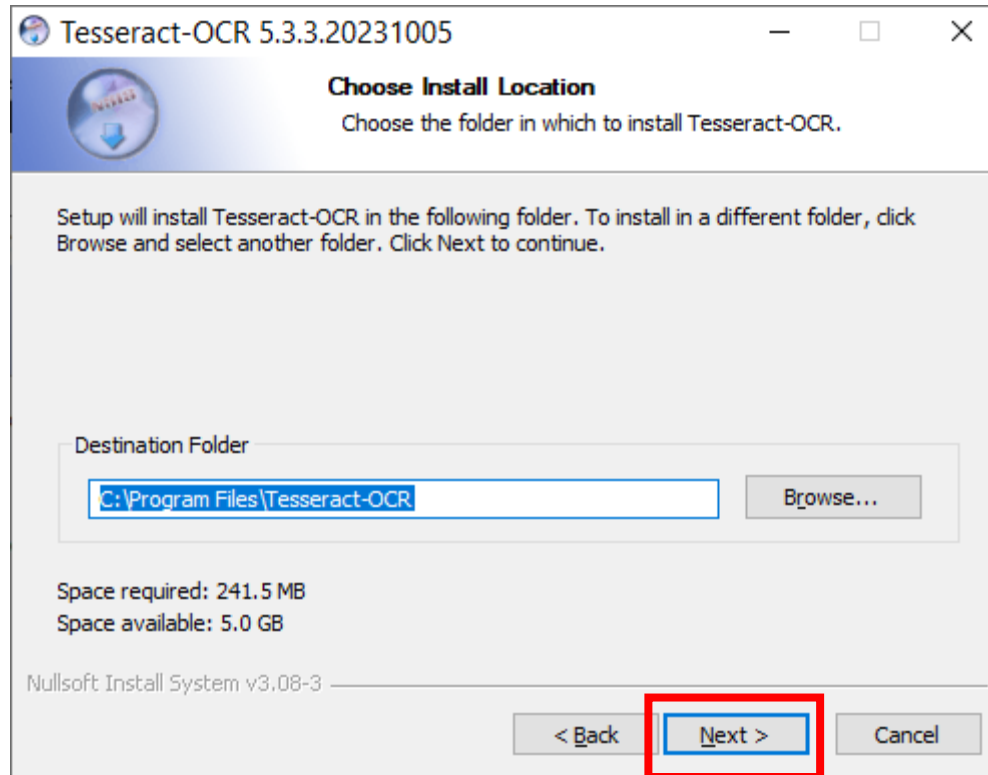
การติดตั้ง Tesseract OCR

6. ที่ Additional language data (download) ให้ติ๊กเลือก Thai จากนั้น กดปุ่ม Next



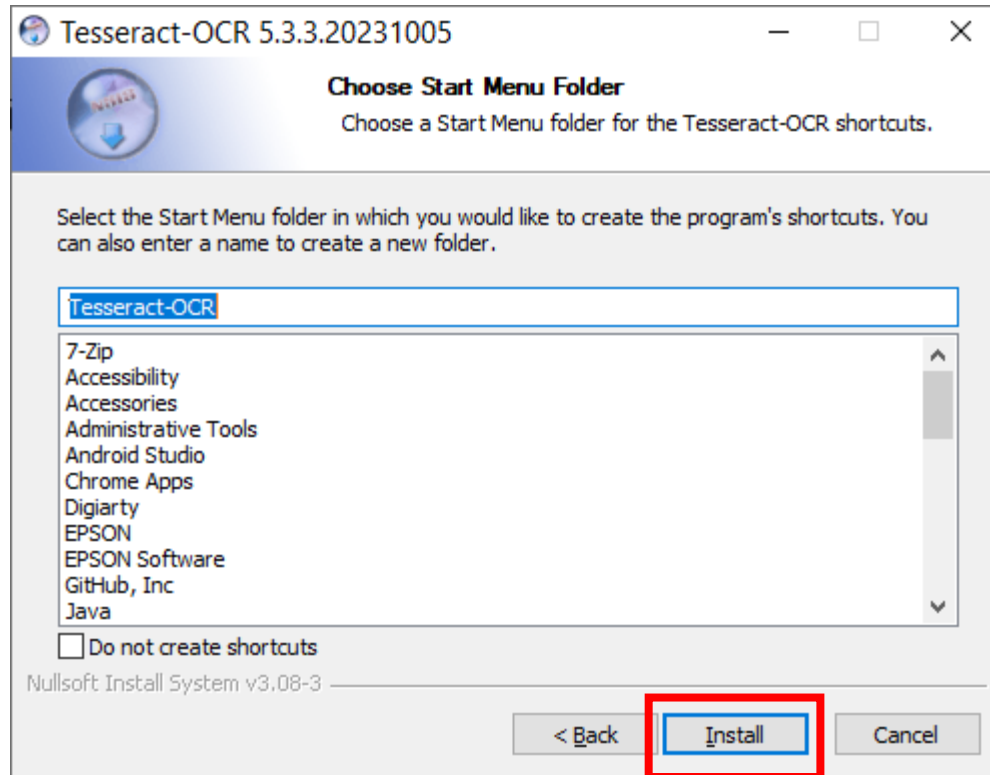
การติดตั้ง Tesseract OCR

7. จากนั้น กดปุ่ม Next



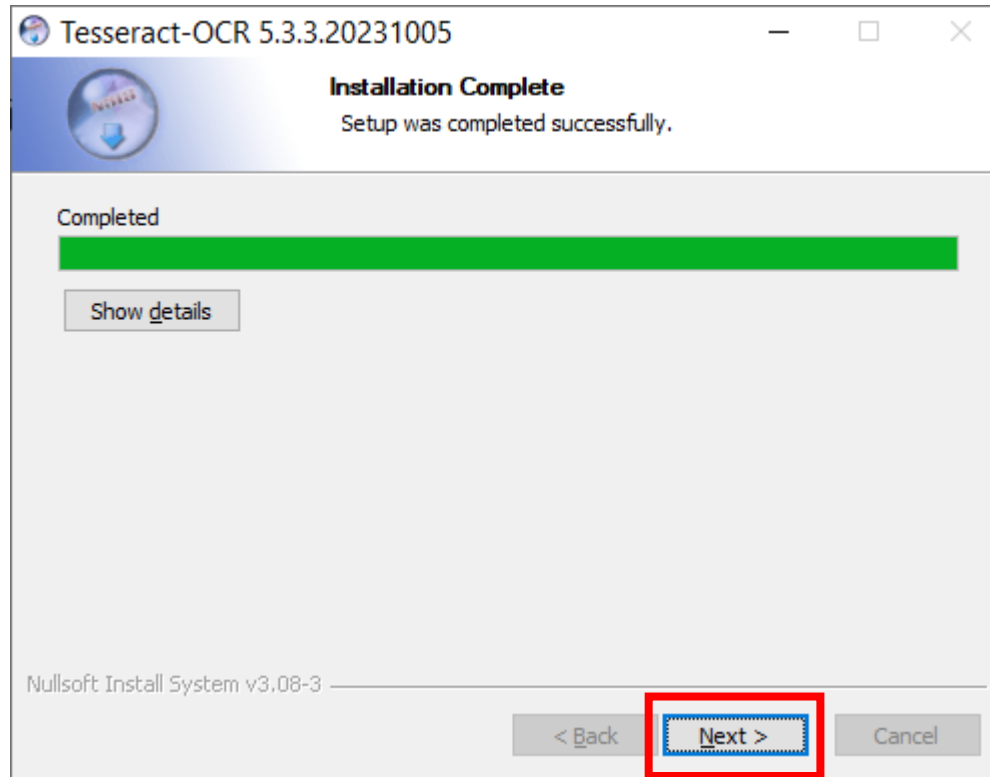
การติดตั้ง Tesseract OCR

8. จากนั้น กดปุ่ม Install



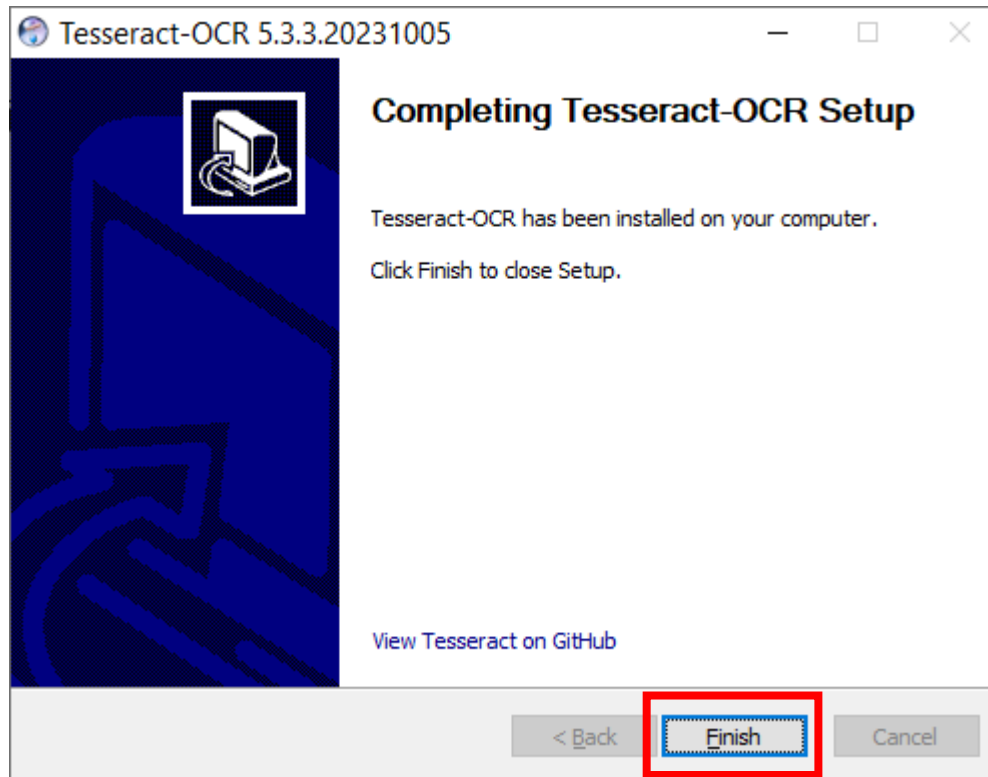
การติดตั้ง Tesseract OCR

9. จากนั้น กดปุ่ม Next



การติดตั้ง Tesseract OCR

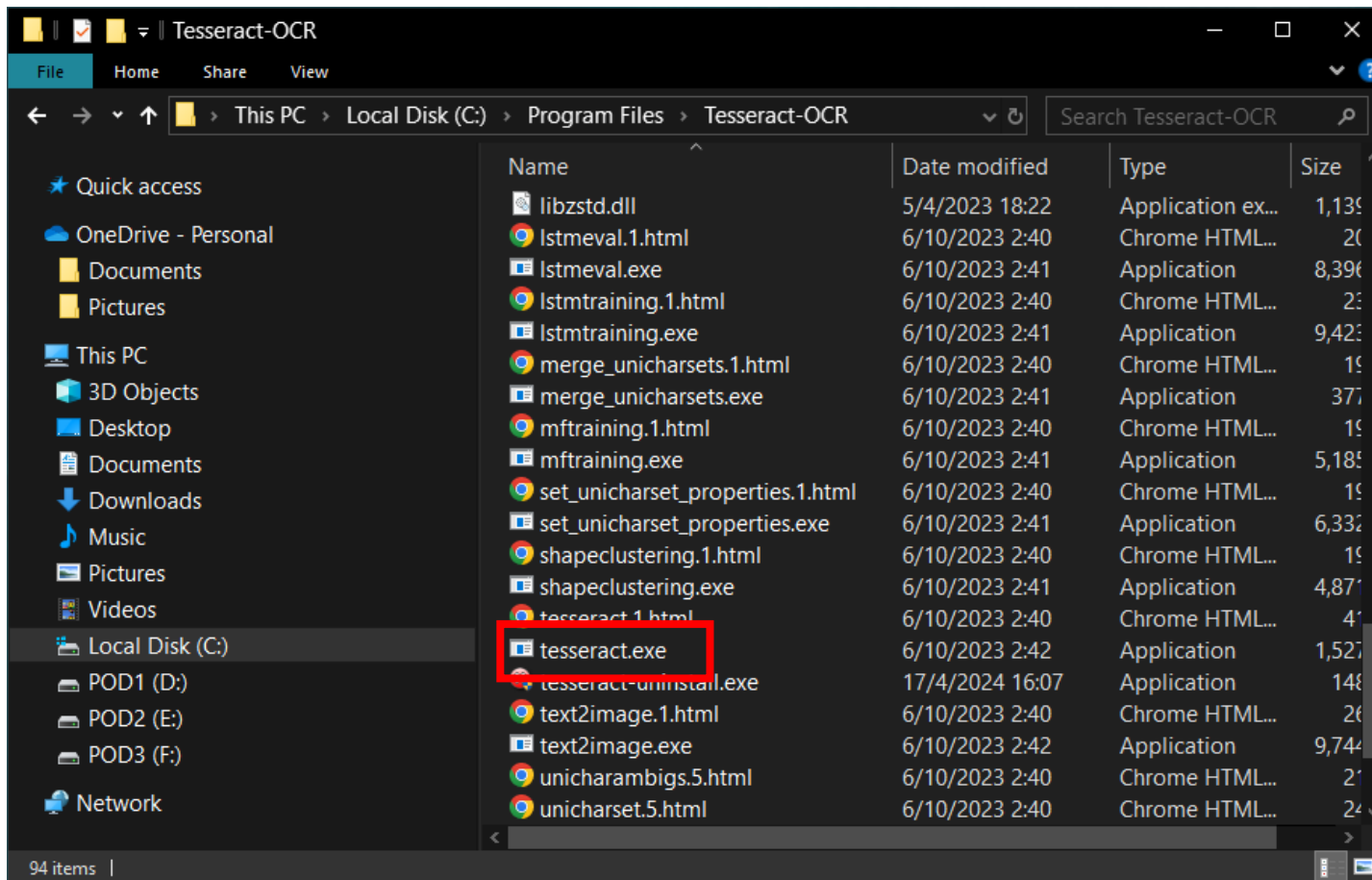
10. จากนั้น กดปุ่ม Finish



การติดตั้ง Tesseract OCR

11. เมื่อเราเข้าไปดูในโฟลเดอร์ที่ติดตั้ง จะพบไฟล์ **tesseract.exe**

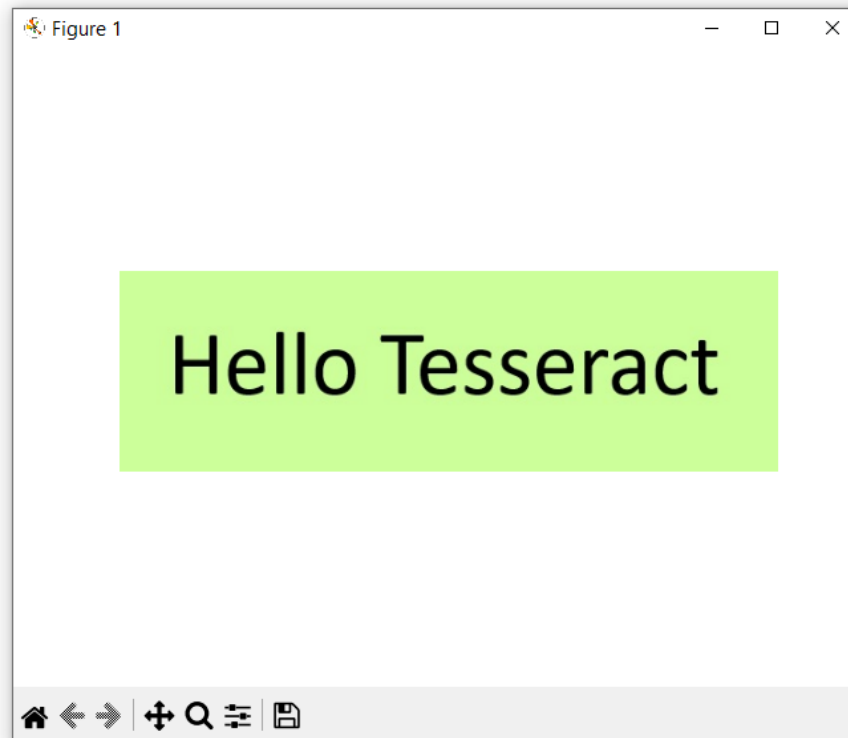
ให้เรา copy path นี้ไว้ใช้งาน **C:\Program Files\Tesseract-OCR\tesseract.exe**



โค้ดโปรแกรมตัวอย่าง

ชื่อไฟล์	โค้ดโปรแกรม
tsr001.py	การใช้ tesseract แปลงภาพเป็นข้อความ ภาษาอังกฤษ ด้วยคำสั่ง image_to_string

```
===== RESTART: D:/pytextc  
Hello Tesseract
```



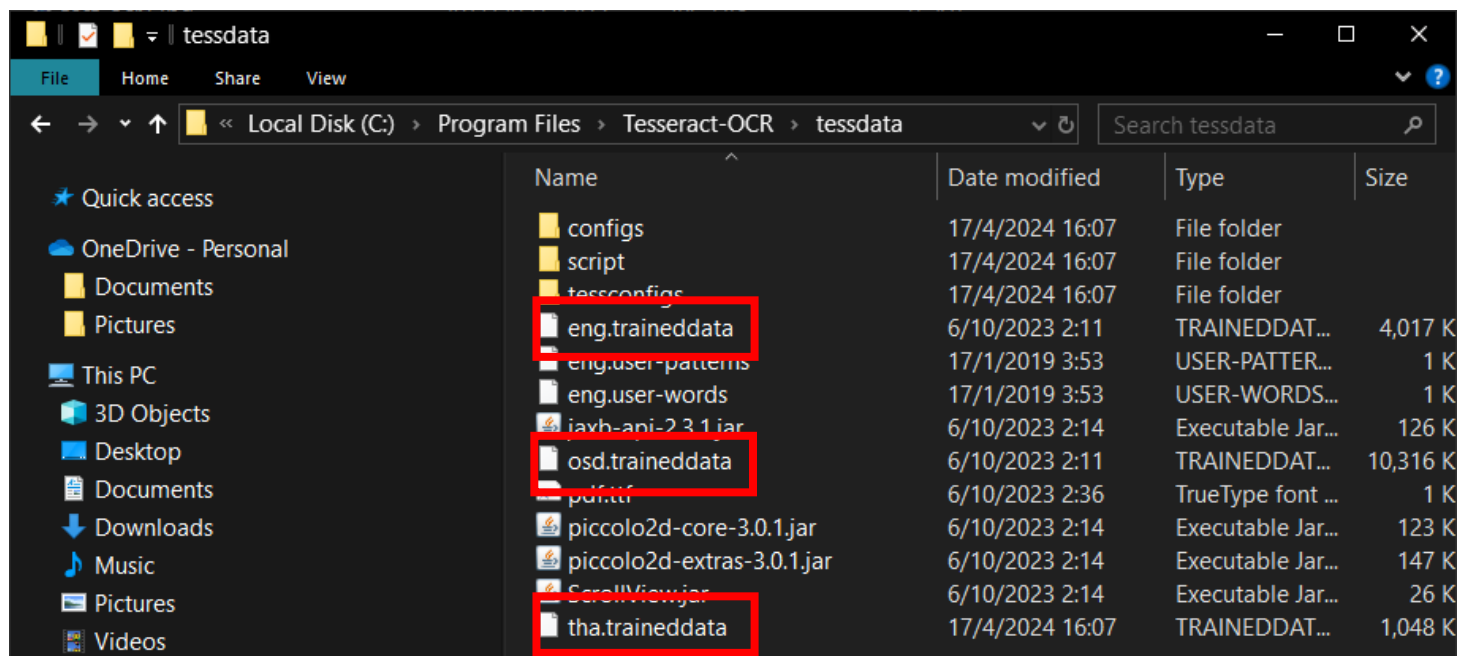
โค้ดโปรแกรมตัวอย่าง

ชื่อไฟล์	โค้ดโปรแกรม
tsr002.py	การดู list ของภาษาที่แปลงได้

`['eng', 'osd', 'tha']`

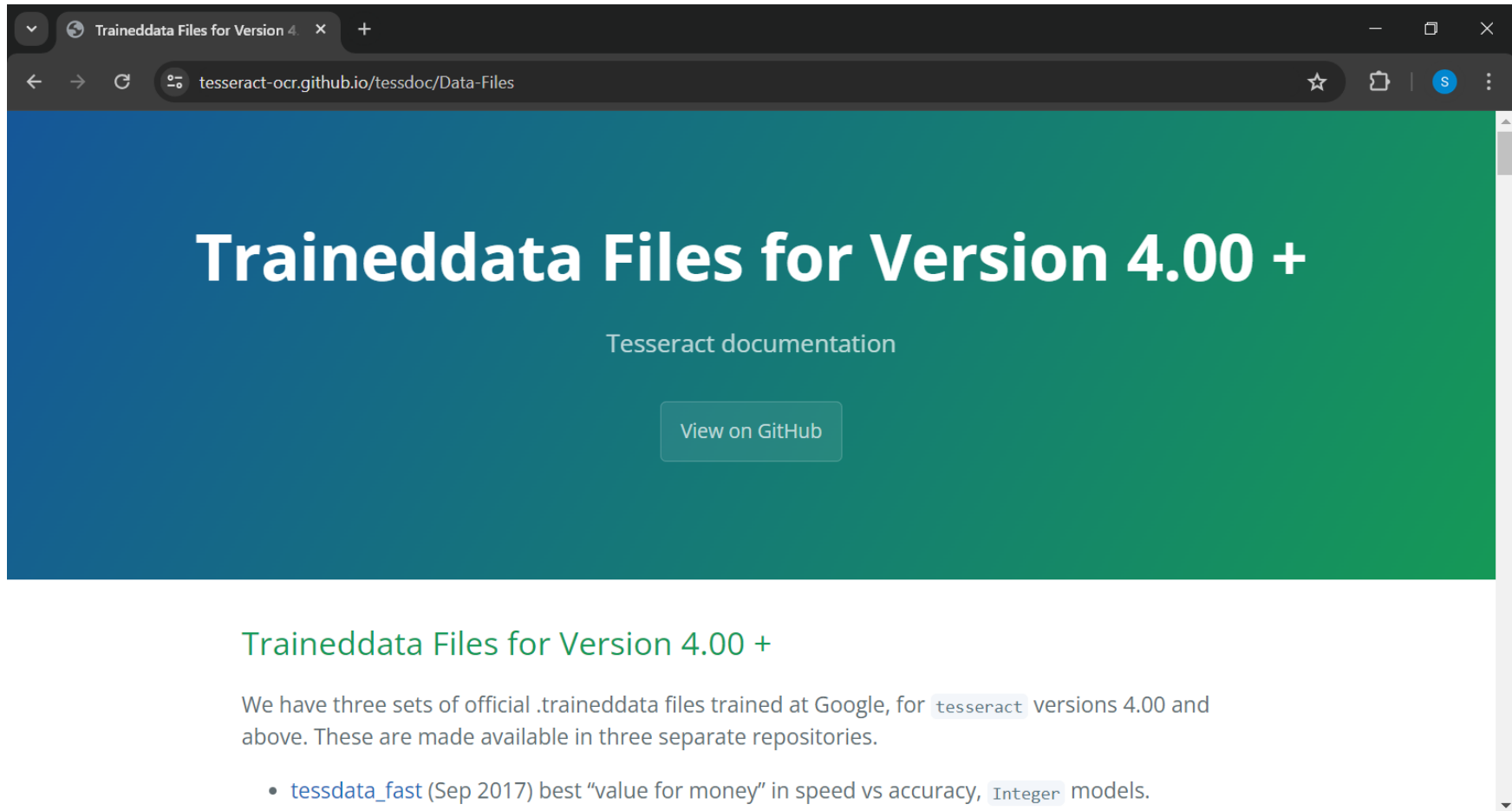
เรายังสามารถดูภาษาที่แปลงได้จากโฟลเดอร์

`C:\Program Files\Tesseract-OCR\tessdata`



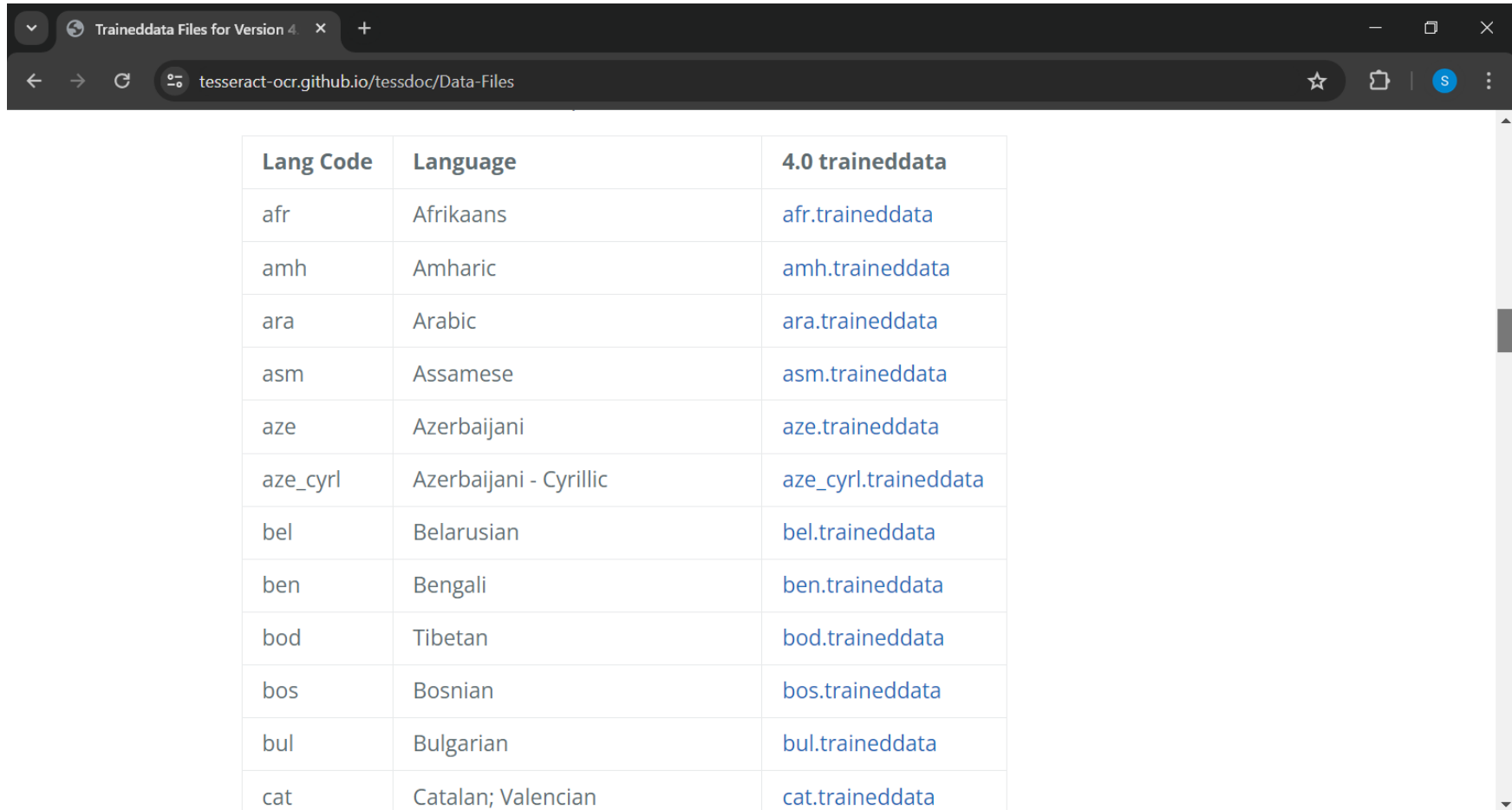
การเพิ่มภาษาที่ต้องการแปลงภาพเป็นข้อความ

1. ให้ไปที่เว็บไซต์ <https://tesseract-ocr.github.io/tessdoc/Data-Files>



การเพิ่มภาษาที่ต้องการแปลงภาพเป็นข้อความ

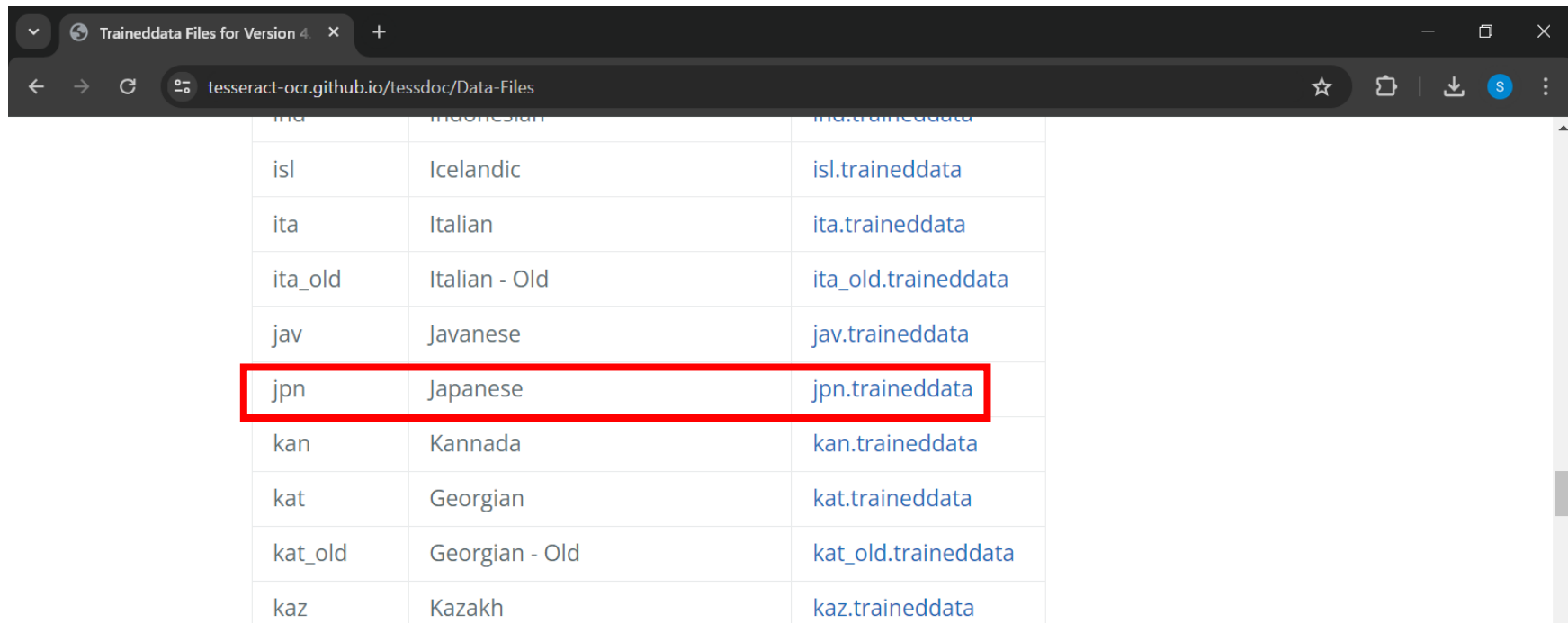
2. เลื่อนลงมาด้านล่าง จะพบ list ของไฟล์ภาษาต่าง ๆ



Lang Code	Language	4.0 traineddata
afr	Afrikaans	afr.traineddata
amh	Amharic	amh.traineddata
ara	Arabic	ara.traineddata
asm	Assamese	asm.traineddata
aze	Azerbaijani	aze.traineddata
aze_cyrl	Azerbaijani - Cyrillic	aze_cyrl.traineddata
bel	Belarusian	bel.traineddata
ben	Bengali	ben.traineddata
bod	Tibetan	bod.traineddata
bos	Bosnian	bos.traineddata
bul	Bulgarian	bul.traineddata
cat	Catalan; Valencian	cat.traineddata

การเพิ่มภาษาที่ต้องการแปลงภาพเป็นข้อความ

3. ตัวอย่างเช่น ถ้าต้องการเพิ่มภาษาญี่ปุ่น (jpn) ให้ดาวน์โหลดไฟล์ [jpn.traineddata](#) ไปเก็บไว้ในโฟลเดอร์ [C:\Program Files\Tesseract-OCR\tessdata](#)

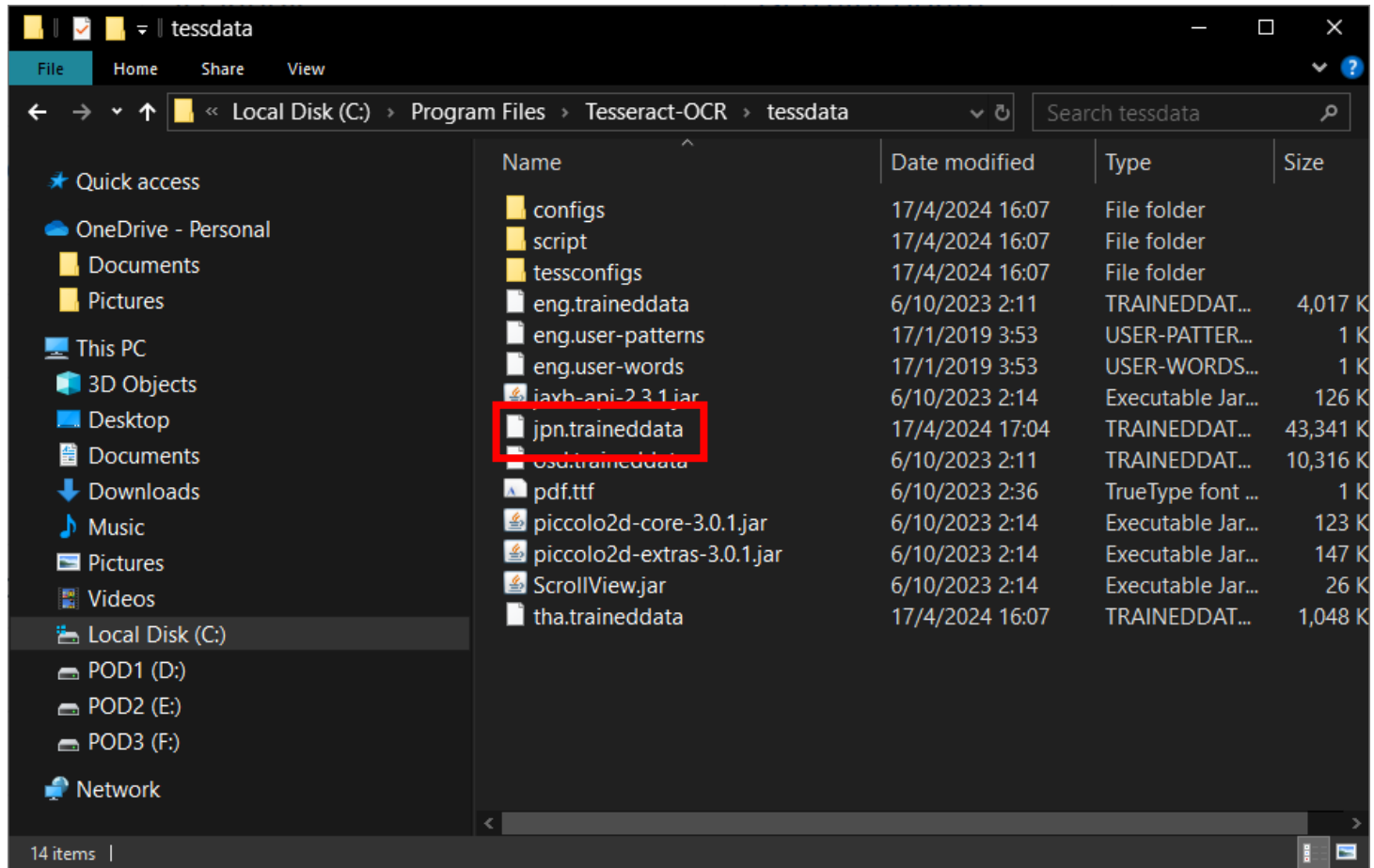


The screenshot shows a web browser window with the address bar displaying tesseract-ocr.github.io/tessdoc/Data-Files. The page content is a table listing various languages and their corresponding trained data files. The row for Japanese (jpn) is highlighted with a red box.

isl	Icelandic	isl.traineddata
ita	Italian	ita.traineddata
ita_old	Italian - Old	ita_old.traineddata
jav	Javanese	jav.traineddata
jpn	Japanese	jpn.traineddata
kan	Kannada	kan.traineddata
kat	Georgian	kat.traineddata
kat_old	Georgian - Old	kat_old.traineddata
kaz	Kazakh	kaz.traineddata

การเพิ่มภาษาที่ต้องการแปลงภาพเป็นข้อความ

4. ไฟล์ภาษาที่อยู่ในโฟลเดอร์ `C:\Program Files\Tesseract-OCR\tessdata`



การเพิ่มภาษาที่ต้องการแปลงภาพเป็นข้อความ

5. เมื่อรันโค้ด `tsr002.py` อีกครั้ง จะเห็นได้ว่า มีจำนวนภาษาที่เพิ่มขึ้น

`['eng', 'jpn', 'osd', 'tha']`

โค้ดโปรแกรมตัวอย่าง

ชื่อไฟล์	โค้ดโปรแกรม
tsr003.py	การใช้ tesseract แปลงภาพเป็นข้อความ ภาษาไทย
tsr004.py	การใช้ tesseract แปลงภาพเป็นข้อความ ภาษาไทย และ ภาษาอังกฤษ
tsr005.py	การกำหนดค่า psm และ oem

การกำหนดค่า Page Segmentation Method (psm)

Page segmentation modes:

- 0 Orientation and script detection (OSD) only.
- 1 Automatic page segmentation with OSD.
- 2 Automatic page segmentation, but no OSD, or OCR. (not implemented)
- 3 Fully automatic page segmentation, but no OSD. (Default)
- 4 Assume a single column of text of variable sizes.
- 5 Assume a single uniform block of vertically aligned text.
- 6 Assume a single uniform block of text.
- 7 Treat the image as a single text line.
- 8 Treat the image as a single word.
- 9 Treat the image as a single word in a circle.
- 10 Treat the image as a single character.
- 11 Sparse text. Find as much text as possible in no particular order.
- 12 Sparse text with OSD.
- 13 Raw line. Treat the image as a single text line, bypassing hacks that are Tesseract-specific.

การกำหนดค่า OCR Engine modes (oem)

OCR Engine modes:

- 0 Legacy engine only.
- 1 Neural nets LSTM engine only.
- 2 Legacy + LSTM engines.
- 3 Default, based on what is available.

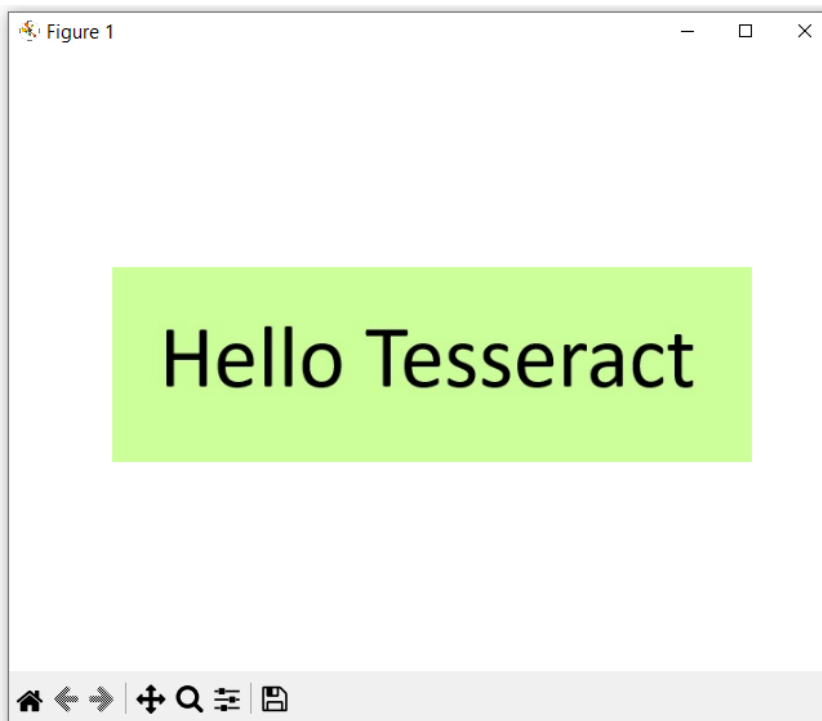
ใช้โมเดล LSTM + โมเดลภาษา + โมเดลการจัดวางข้อความ

โค้ดโปรแกรมตัวอย่าง

ชื่อไฟล์	โค้ดโปรแกรม
tsr006.py	การใช้คำสั่ง image_to_boxes หาดำแหน่งของตัวอักษร

```
['H', '47', '65', '82', '114']  
['e', '91', '64', '123', '102']  
['l', '132', '65', '138', '117']  
['l', '149', '65', '156', '117']  
['o', '165', '64', '199', '102']  
['T', '220', '65', '256', '114']  
['e', '253', '64', '284', '102']  
['s', '290', '64', '315', '102']  
['s', '321', '64', '344', '102']  
['e', '351', '64', '382', '102']  
['r', '391', '65', '411', '102']  
['a', '414', '64', '442', '102']  
['c', '451', '64', '477', '102']  
['t', '481', '64', '503', '110']
```

symbol left bottom right top



โค้ดโปรแกรมตัวอย่าง

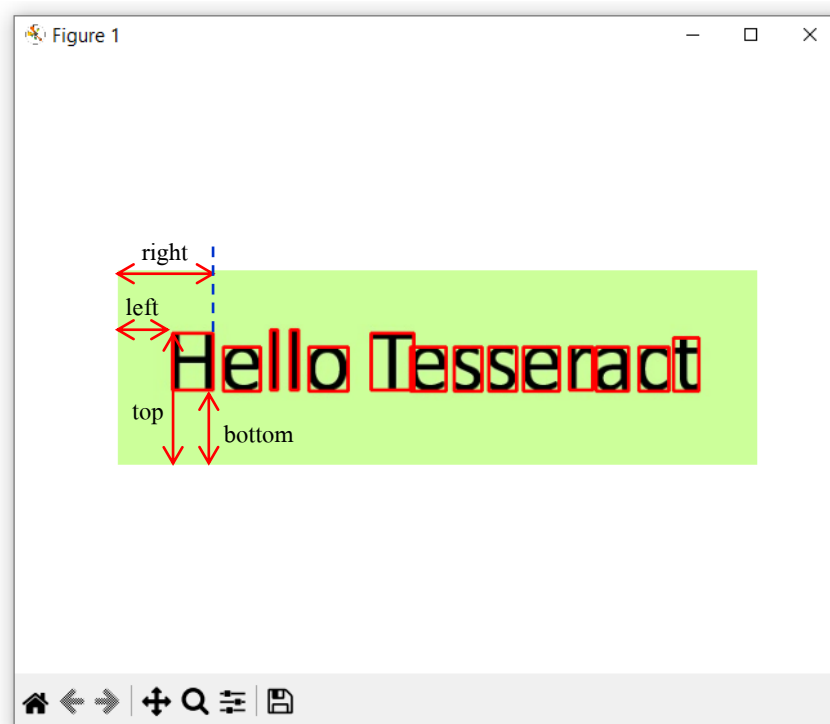
ชื่อไฟล์	โค้ดโปรแกรม
tsr007.py	การวาดกรอบรอบตัวอักษรด้วย OpenCV

```

['H', '47', '65', '82', '114']
['e', '91', '64', '123', '102']
['l', '132', '65', '138', '117']
['l', '149', '65', '156', '117']
['o', '165', '64', '199', '102']
['T', '219', '65', '256', '114']
['e', '253', '64', '284', '102']
['s', '291', '64', '315', '102']
['s', '321', '64', '345', '102']
['e', '351', '64', '382', '102']
['r', '391', '65', '411', '102']
['a', '414', '64', '442', '102']
['c', '451', '64', '477', '102']
['t', '481', '64', '503', '110']

```

symbol left bottom right top



โค้ดโปรแกรมตัวอย่าง

ชื่อไฟล์	โค้ดโปรแกรม
tsr008.py	การใช้คำสั่ง image_to_data ภาษาอังกฤษ
tsr009.py	การใช้คำสั่ง image_to_data ภาษาไทย
tsr010.py	การใช้คำสั่ง image_to_data ภาษาไทย และ ภาษาอังกฤษ
tsr011.py	การใช้คำสั่ง image_to_data พร้อมวาดกรอบ
tsr012.py	การใช้คำสั่ง image_to_data พร้อมวาดกรอบ และ บันทึกภาพ Output
tsr013.py	การแสดงความถี่ที่ตรวจจับได้จาก image_to_boxes ด้วย pygame
tsr014.py	การแสดงความถี่ที่ตรวจจับได้จาก image_to_data ด้วย pygame

EasyOCR

EasyOCR

เว็บไซต์ EasyOCR → <https://github.com/JaidedAI/EasyOCR>

The screenshot shows the GitHub repository page for EasyOCR. The main content area displays the README, which includes the project title 'EasyOCR', a description of its capabilities (ready-to-use OCR with 80+ supported languages), and a list of recent updates. The right sidebar provides additional project statistics and details.

EasyOCR

Ready-to-use OCR with 80+ [supported languages](#) and all popular writing scripts including: Latin, Chinese, Arabic, Devanagari, Cyrillic, etc.

[Try Demo on our website](#)

Integrated into [Huggingface Spaces](#) using [Gradio](#). Try out the Web Demo: [Hugging Face Spaces](#)

What's new

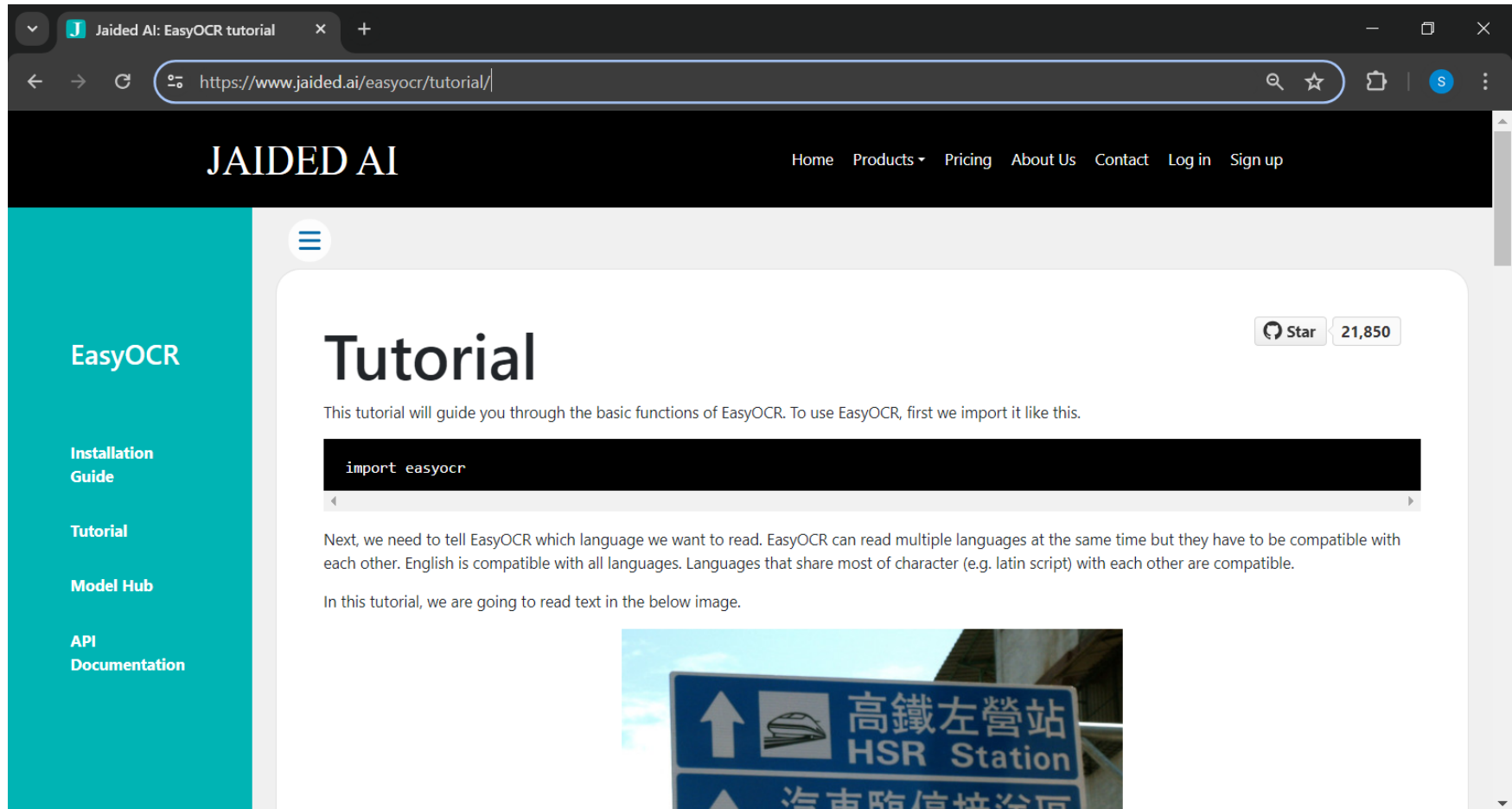
- 4 September 2023 - Version 1.7.1
 - Fix several compatibilities
- 25 May 2023 - Version 1.7.0
 - Add Apple Silicon support (thanks [@rayeesoft](#) and [@ArtemBernatsky](#), see [PR](#))
 - Fix several compatibilities
- 15 September 2022 - Version 1.6.2

Right Sidebar:

- v1.7.0 Latest** on May 25, 2023
- + 21 releases
- Packages:** No packages published
- Used by:** 5.8k (+ 5,762)
- Contributors:** 114 (+ 100 contributors)
- Languages:**
 - Python 77.2%
 - C++ 13.7%
 - Cuda 7.1%
 - Jupyter Notebook 1.1%
 - C 0.6%
 - Ruby 0.2%
 - Other 0.1%

EasyOCR

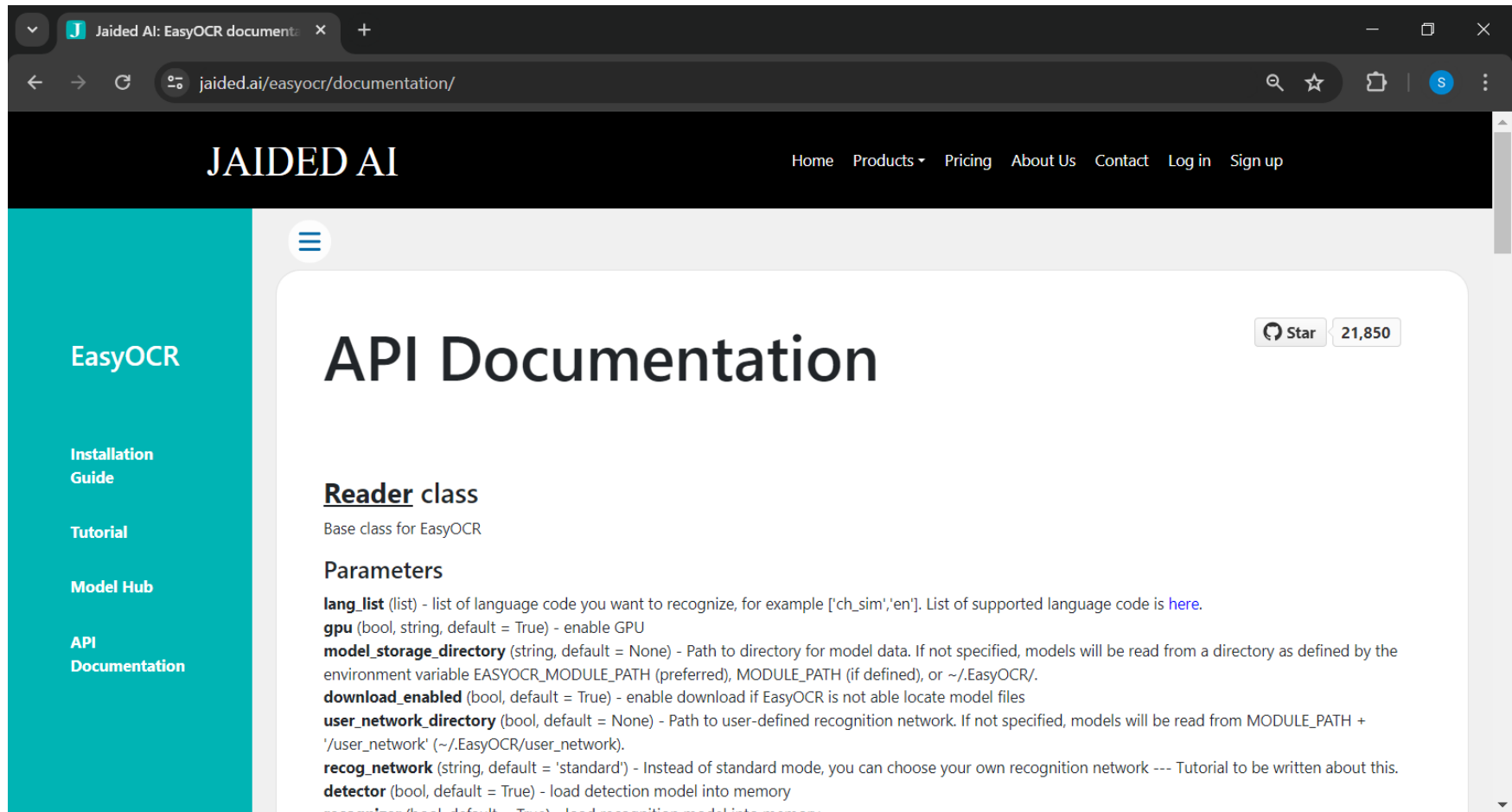
เว็บไซต์ EasyOCR → <https://www.jaided.ai/easyocr/tutorial>



The screenshot shows a web browser window displaying the EasyOCR tutorial page. The browser's address bar shows the URL <https://www.jaided.ai/easyocr/tutorial/>. The page header features the "JAIDED AI" logo and navigation links: Home, Products, Pricing, About Us, Contact, Log in, and Sign up. A teal sidebar on the left contains a menu with the following items: EasyOCR, Installation Guide, Tutorial, Model Hub, and API Documentation. The main content area is titled "Tutorial" and includes a GitHub star badge showing 21,850 stars. The text below the title reads: "This tutorial will guide you through the basic functions of EasyOCR. To use EasyOCR, first we import it like this." Below this text is a code block containing the Python code `import easyocr`. The next paragraph states: "Next, we need to tell EasyOCR which language we want to read. EasyOCR can read multiple languages at the same time but they have to be compatible with each other. English is compatible with all languages. Languages that share most of character (e.g. latin script) with each other are compatible." The final paragraph says: "In this tutorial, we are going to read text in the below image." Below the text is an image of a blue sign for "HSR Station" (High-Speed Rail Station) with Chinese characters "高鐵左營站" and "汽車隨停接洽區".

EasyOCR

เว็บไซต์ EasyOCR → <https://www.jaided.ai/easyocr/documentation>



The screenshot shows a web browser displaying the EasyOCR API Documentation page. The browser's address bar shows the URL `jaided.ai/easyocr/documentation/`. The page features a dark teal header with the JAIDED AI logo and navigation links: Home, Products, Pricing, About Us, Contact, Log in, and Sign up. A teal sidebar on the left contains navigation links: EasyOCR, Installation Guide, Tutorial, Model Hub, and API Documentation (which is highlighted). The main content area is titled "API Documentation" and includes a GitHub Star button showing 21,850 stars. Below the title, the "Reader class" is introduced as the base class for EasyOCR. The "Parameters" section lists several configuration options: `lang_list` (list of language codes), `gpu` (boolean to enable GPU), `model_storage_directory` (string path for model data), `download_enabled` (boolean to enable model downloads), `user_network_directory` (boolean path for user-defined networks), `recog_network` (string for custom recognition networks), and `detector` (boolean to load detection models into memory).

โค้ดโปรแกรมตัวอย่าง

ชื่อไฟล์	โค้ดโปรแกรม
easy001.py	การใช้ EasyOCR แปลงภาพเป็นข้อความ
easy002.py	การใช้ EasyOCR แปลงภาพเป็นข้อความ พร้อมทั้งแสดงกรอบของข้อความ
easy003.py	การใช้ EasyOCR แปลงภาพเป็นข้อความ พร้อมทั้งแสดงกรอบของข้อความ และบันทึกภาพ Output
easy004.py	การใช้ EasyOCR แปลงภาพเป็นข้อความ พร้อมทั้งแสดงกรอบของข้อความ บันทึกภาพ Output และ บันทึกข้อความที่ได้ลงในไฟล์ txt
easy005.py	การแสดงผลข้อความที่ตรวจจับได้โดยใช้ pygame

Gemini

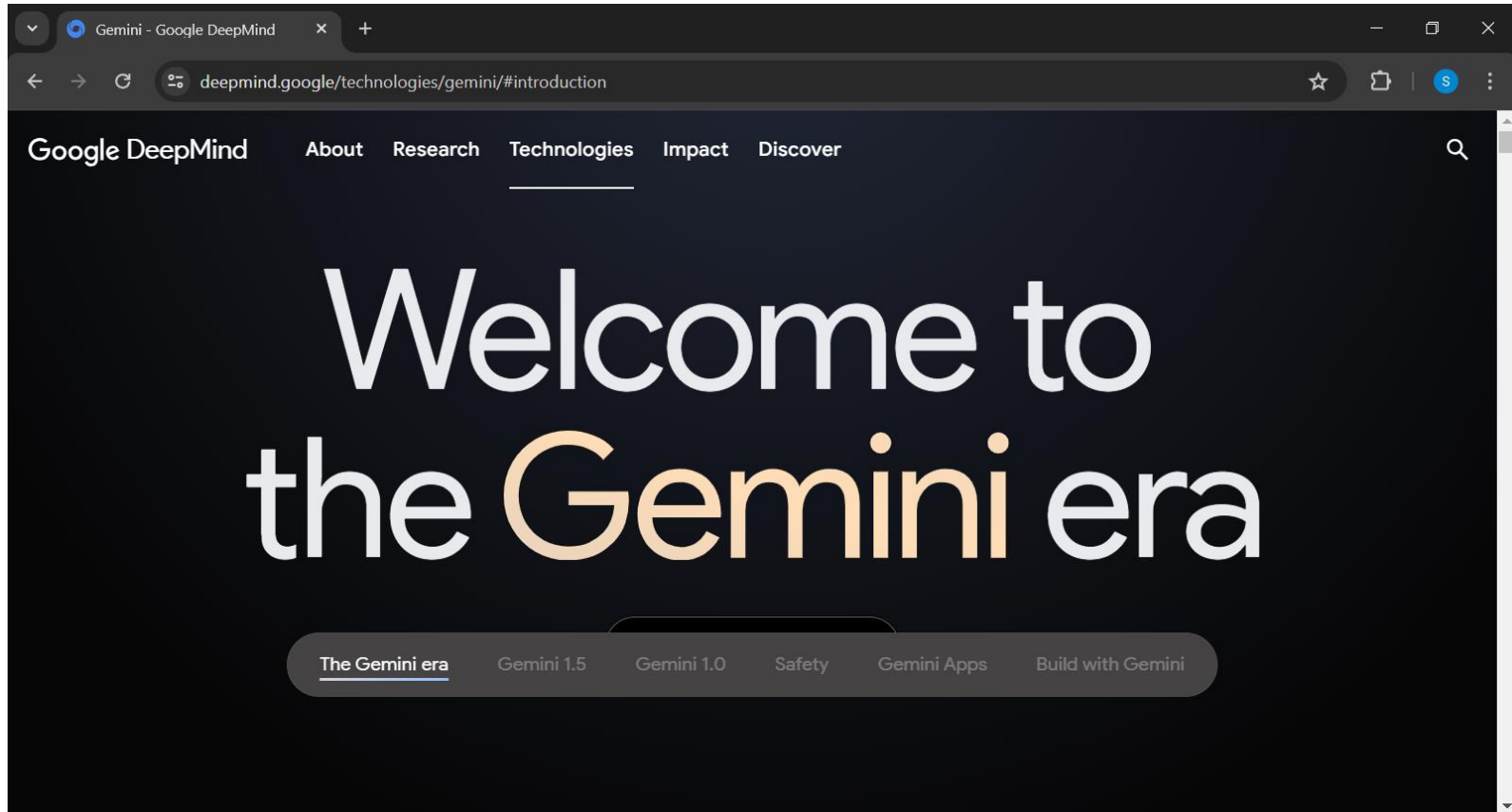
Google Gemini

Google Gemini เป็นโมเดลภาษาขนาดใหญ่ (LLM) รุ่นใหม่ล่าสุดจาก Google AI ที่เปิดตัวเมื่อวันที่ 6 ธันวาคม 2566

Gemini พัฒนาต่อยอดมาจากโมเดล LaMDA และ PaLM 2 โดยมีความพิเศษดังนี้

- รองรับมัลติมีเดีย: Gemini ไม่ได้จำกัดอยู่แค่ข้อความเท่านั้น แต่สามารถประมวลผลข้อมูลประเภทอื่น ๆ ได้อีก เช่น โคลด์ เสียง ภาพ และวิดีโอ
- มี 3 ขนาด:
 - Gemini Ultra: รองรับงานซับซ้อน
 - Gemini Pro: ทำงานหลากหลาย
 - Gemini Nano: ประสิทธิภาพสูง เหมาะกับมือถือ

เว็บไซต์ของ Gemini → <https://deepmind.google/technologies/gemini>



ตัวอย่างการใช้งาน Gemini

gemini-pro (text to text)

- สอบถาม/ค้นหา ข้อมูล
- แปลภาษา
- แต่งนิทาน แต่งกลอน
- ร่างจดหมาย อีเมล
- ขอโค้ด Python Java C++ Go
- ขอตำแหน่งที่อยู่ของสถานที่
- ฯลฯ

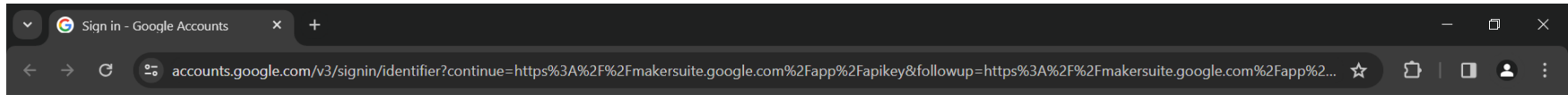
ตัวอย่างการใช้งาน Gemini

gemini-pro-vision (image to text)

- บรรยายภาพ องค์ประกอบต่าง ๆ ภายในภาพ
- วิเคราะห์ภาพ
- นับจำนวนวัตถุในภาพ เช่น คน สัตว์ สิ่งของ
- จำแนก/รู้จำวัตถุในภาพ เช่น รถยนต์ พืช สัตว์ อาหาร
- ตรวจจับข้อความในภาพ
- ฯลฯ

การขอ API Key สำหรับ Gemini

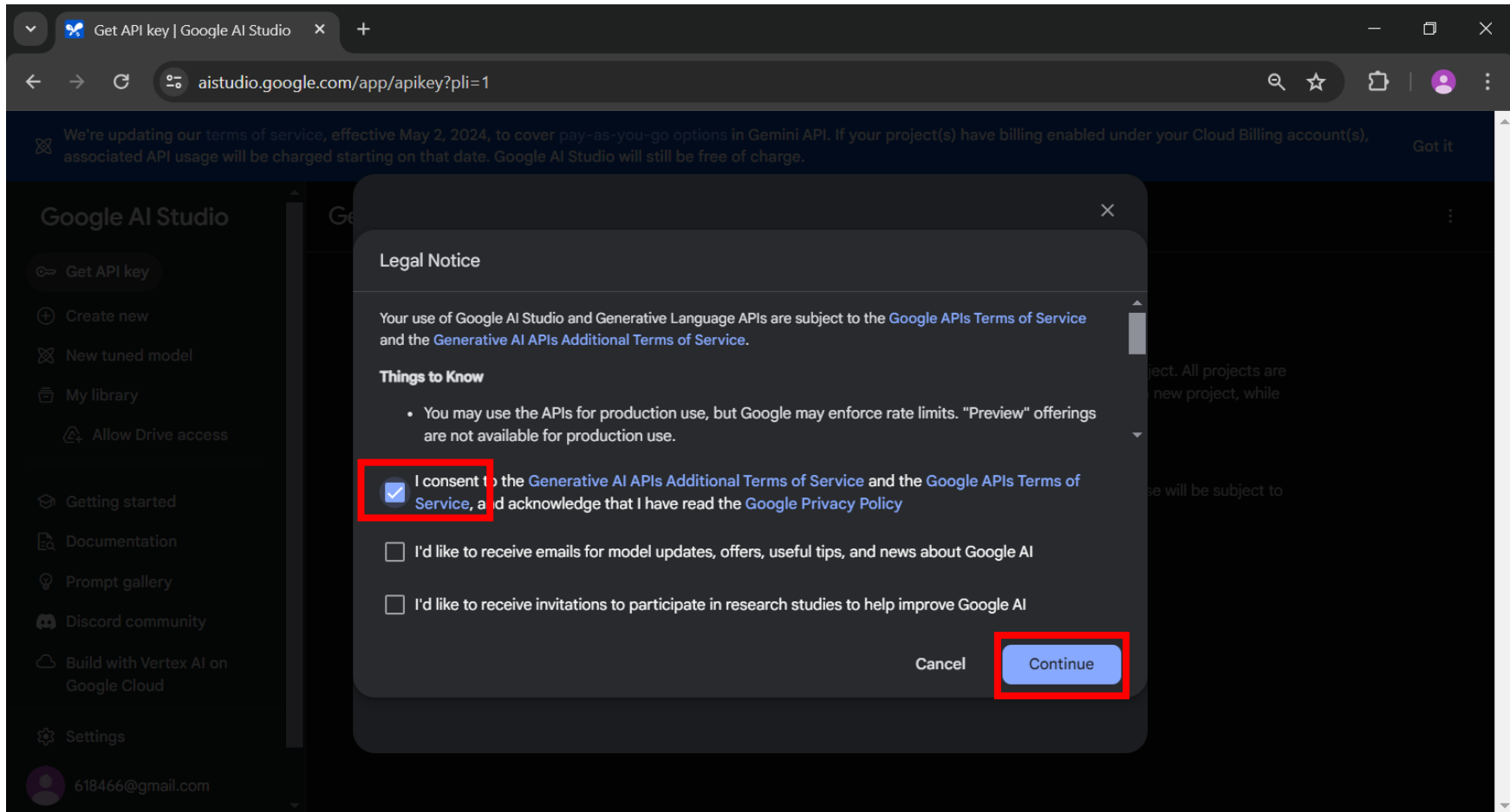
ให้เข้าไปที่เว็บไซต์ → <https://aistudio.google.com/app/apikey>
จากนั้น ทำการ Sign in ด้วย Google Account

The image shows the Google sign-in form. At the top is the Google logo. Below it is the text "Sign in" and "Use your Google Account". There is a text input field with the placeholder "Email or phone" and a vertical cursor. Below the input field is a link "Forgot email?". Further down is the text "Not your computer? Use Guest mode to sign in privately." followed by a link "Learn more about using Guest mode". At the bottom left is a link "Create account" and at the bottom right is a blue button labeled "Next".

English (United States) Help Privacy Terms

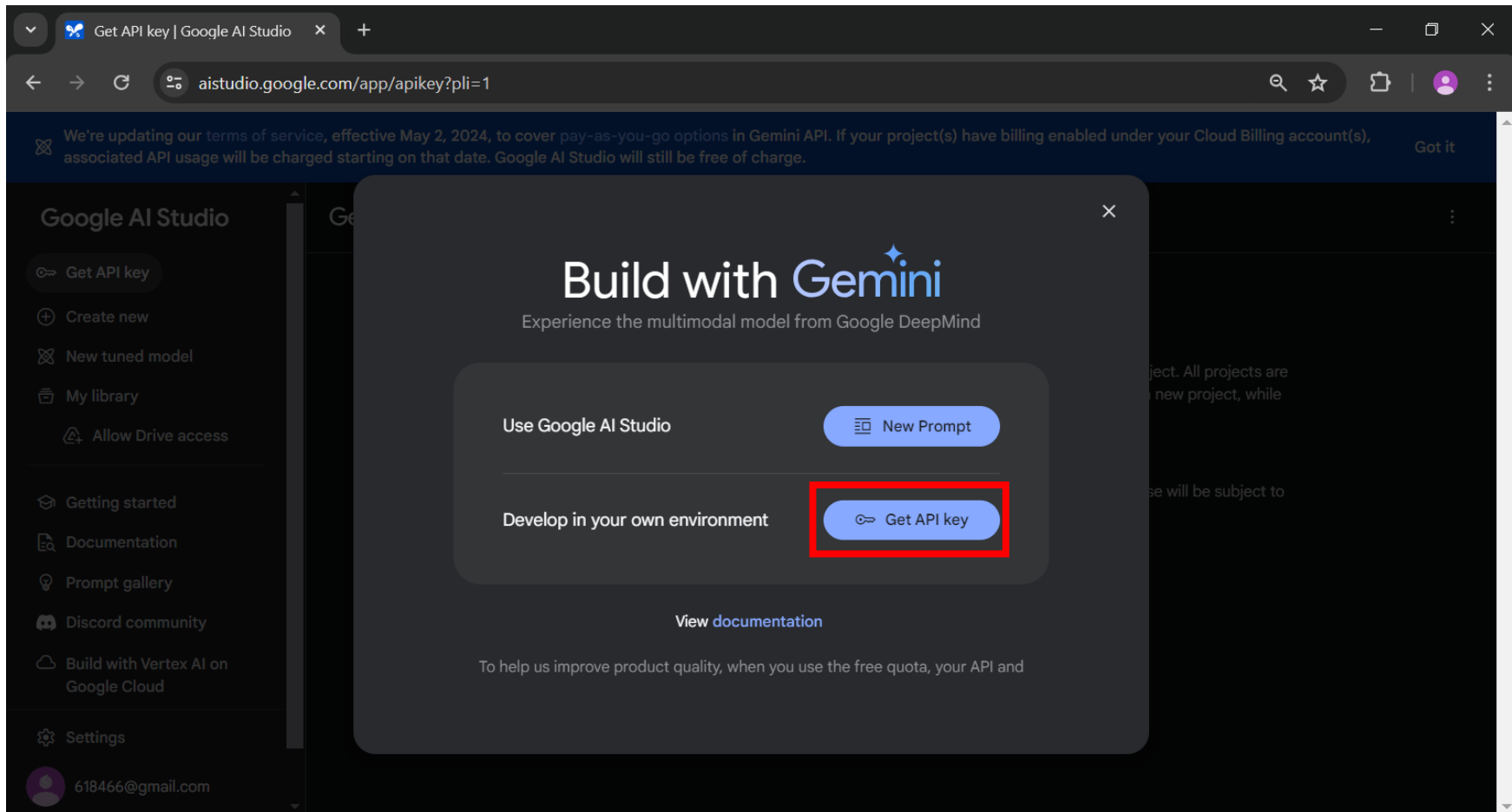
การขอ API Key สำหรับ Gemini

ติ๊กยอมรับเงื่อนไขการใช้งาน แล้วกดปุ่ม **Continue**



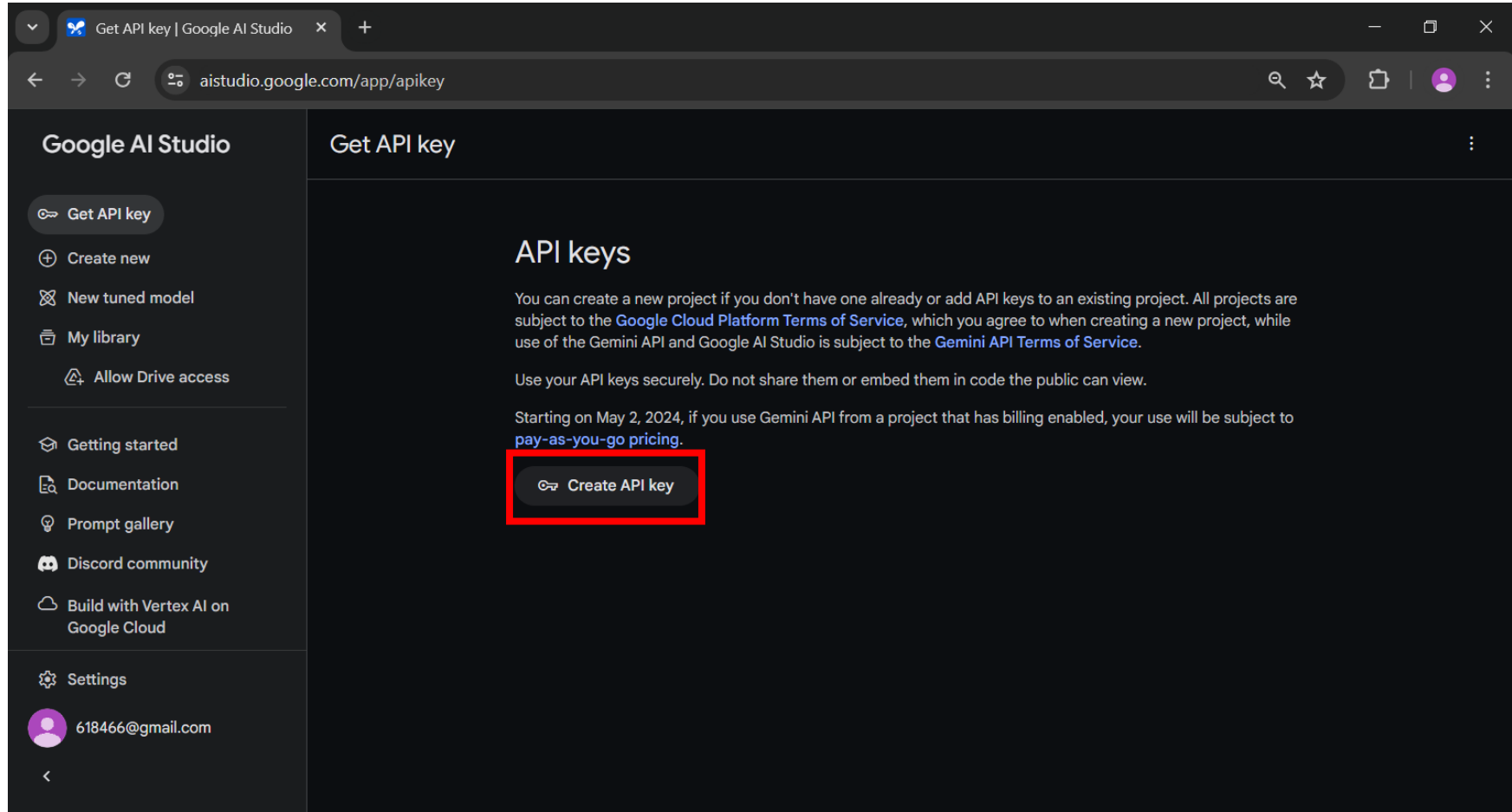
การขอ API Key สำหรับ Gemini

คลิก **Get API key**



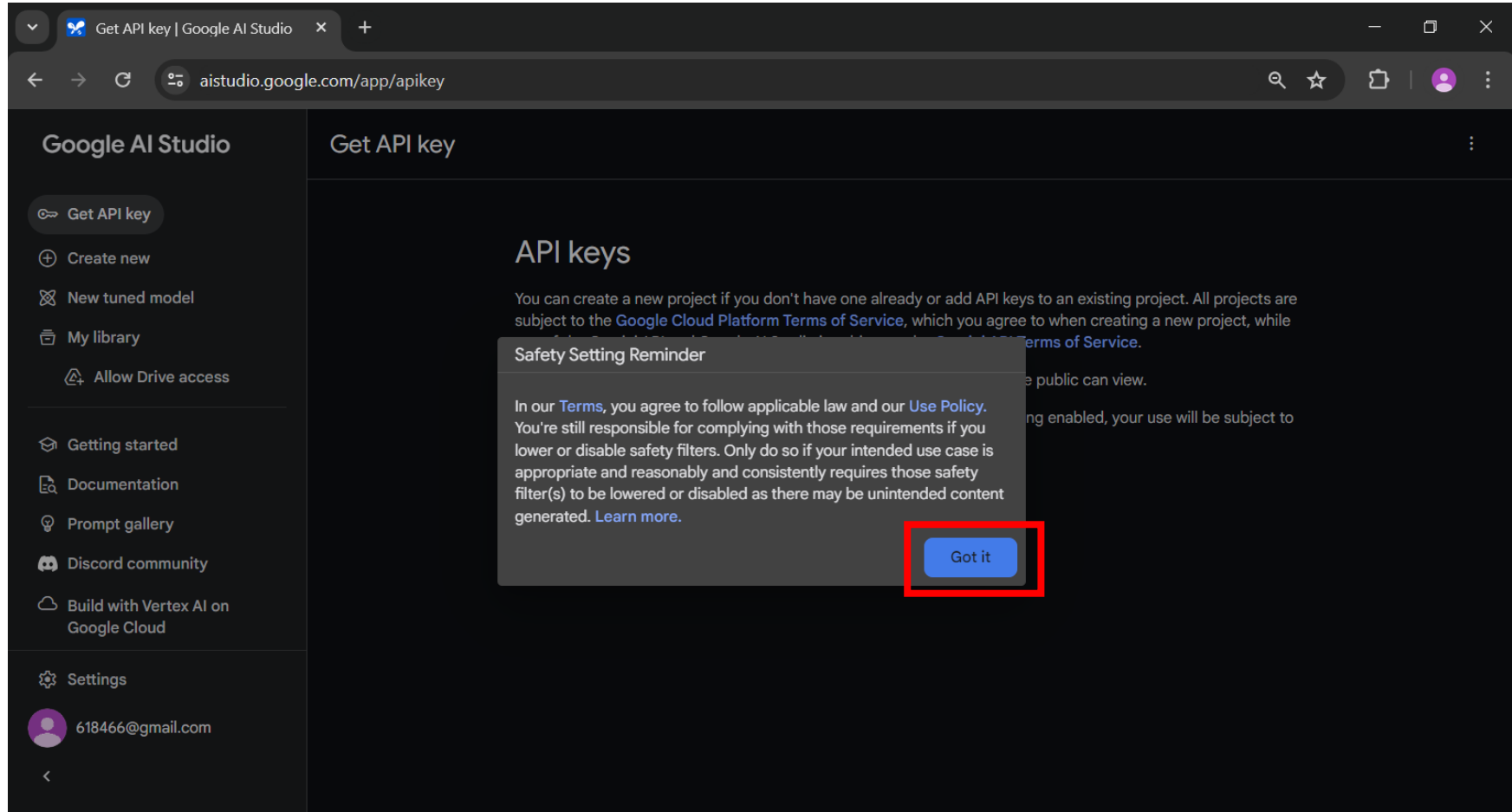
การขอ API Key สำหรับ Gemini

คลิก **Create API key**



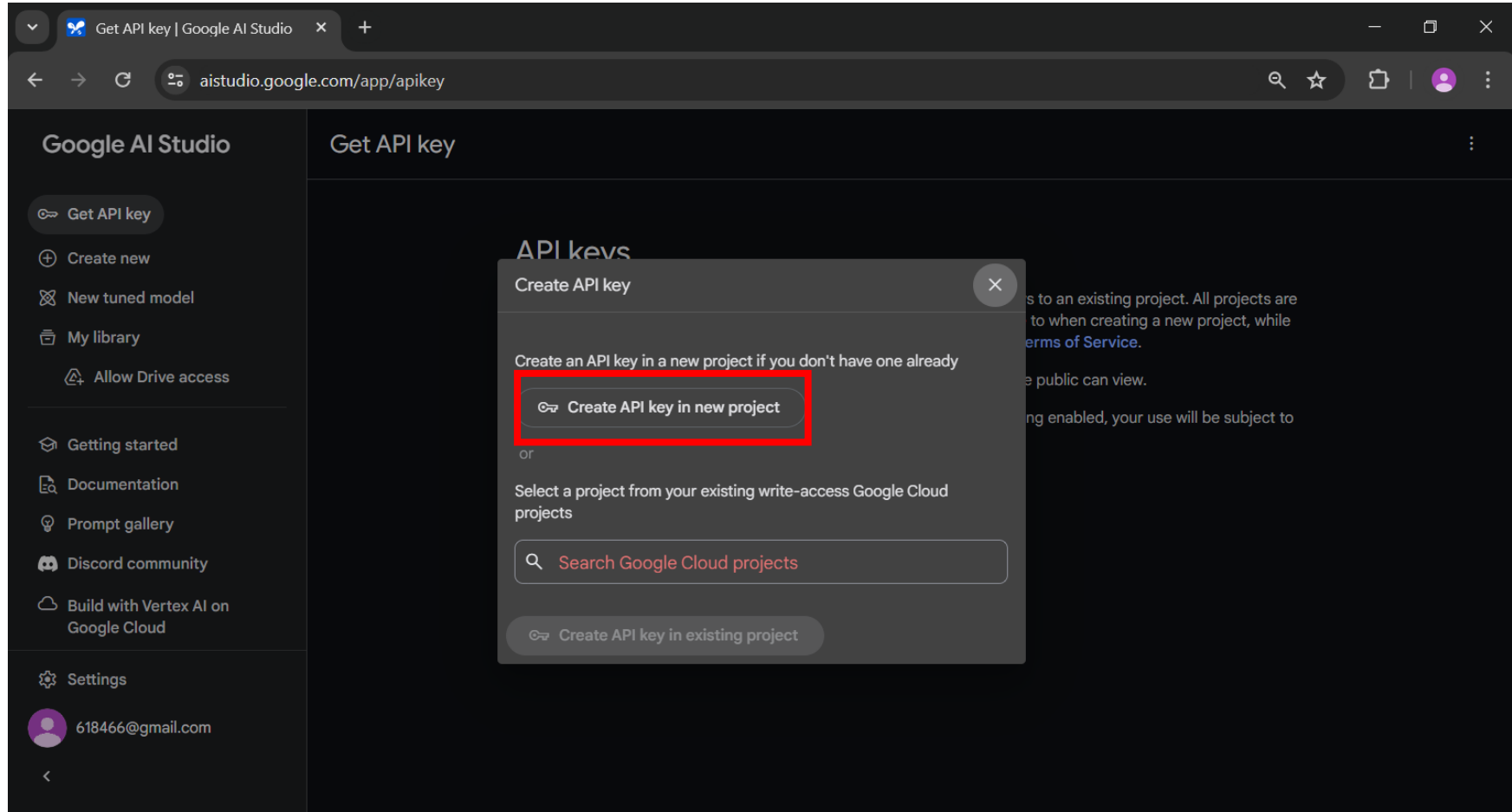
การขอ API Key สำหรับ Gemini

คลิก **Got it**



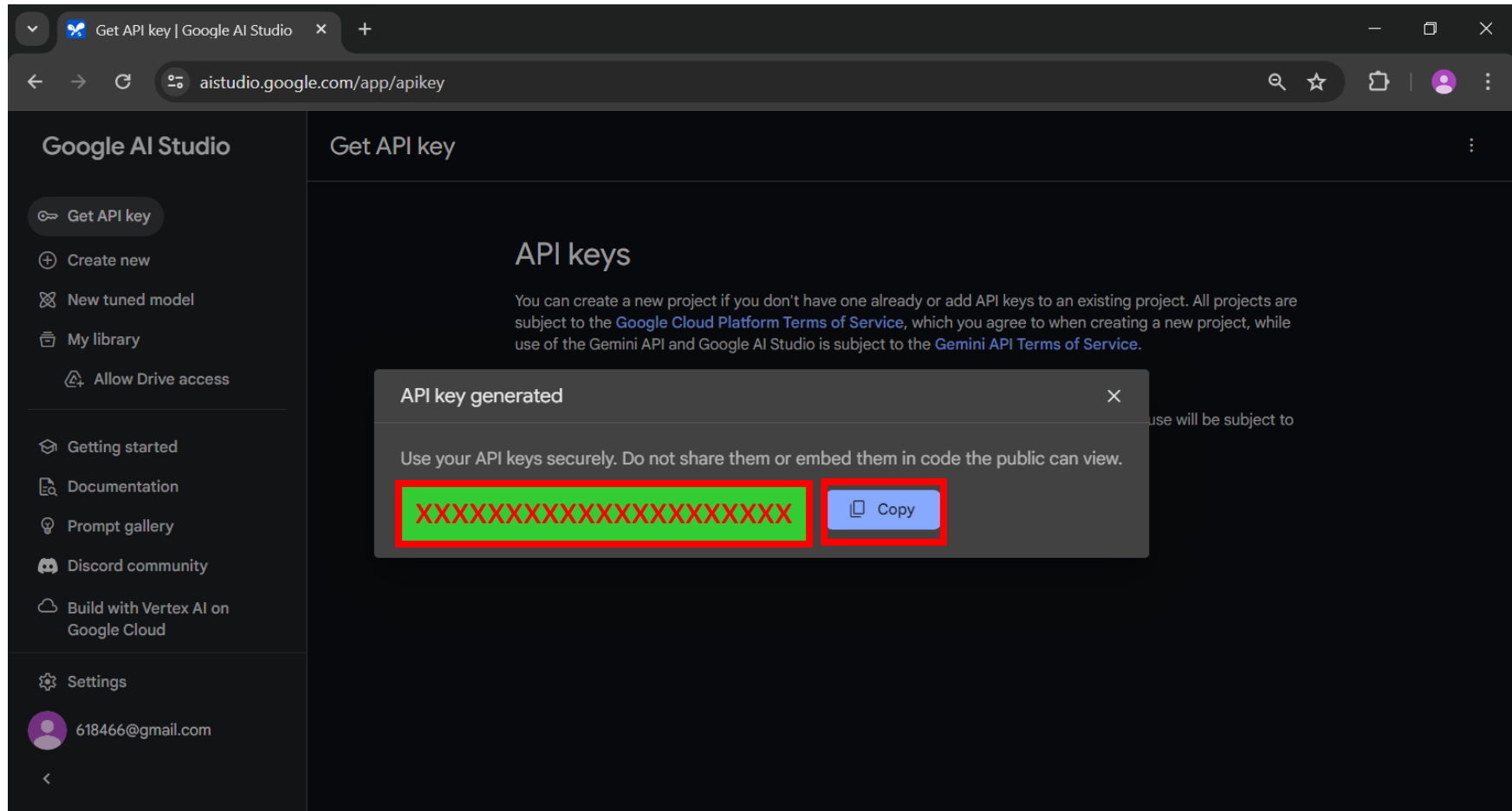
การขอ API Key สำหรับ Gemini

คลิก **Create API key in new project**



การขอ API Key สำหรับ Gemini

เมื่อได้ API key แล้วให้กดปุ่ม Copy เก็บไว้ใช้งาน



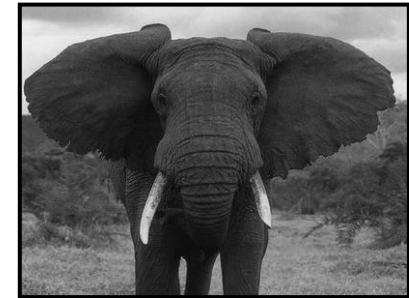
โค้ดโปรแกรมตัวอย่าง

ชื่อไฟล์	โค้ดโปรแกรม
gemini001.py	การใช้งาน gemini-pro-vision (image to text) ตรวจสอบข้อความในภาพ

ประเภทของภาพ (Image Types)

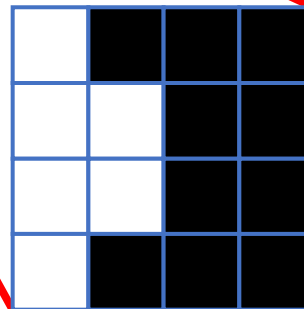
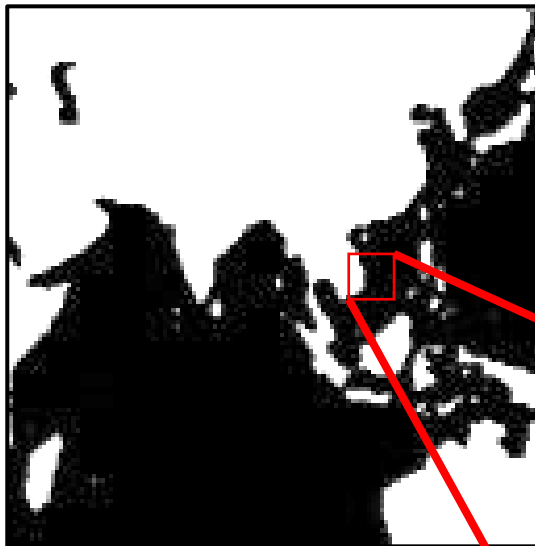
โดยทั่วไปเราสามารถแบ่งประเภทของภาพบิตแมป (Bitmap Image) ตามคุณสมบัติของสีออกได้เป็น 4 ประเภท ดังนี้

- ภาพขาวดำ (Binary Image)
- ภาพระดับเทา (Grayscale Image)
- ภาพสี RGB (Color Image)



ภาพขาวดำ (Binary Image)

ลักษณะของภาพขาวดำ คือ ในแต่ละพิกเซลจะแสดงด้วยค่าแบบไบนารี (Binary) โดยที่ 1 หมายถึงจุดภาพสีขาว และ 0 หมายถึงจุดภาพสีดำ (ในการเขียนโปรแกรมบางครั้งเราจะใช้ 255 (2^8 บิต-1) แทนสีขาว และใช้ค่า 0 แทนสีดำ)

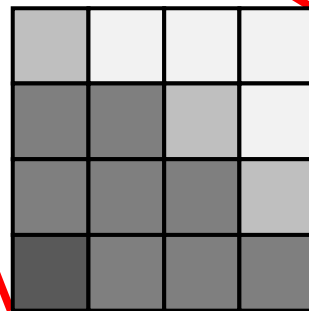
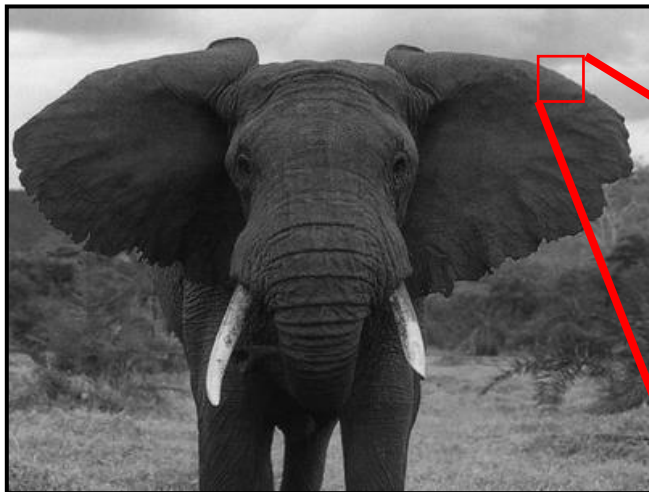


Binary Data

1	0	0	0
1	1	0	0
1	1	0	0
1	0	0	0

ภาพระดับเทา (Grayscale Image)

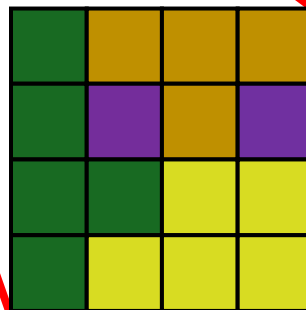
ลักษณะของภาพชนิดนี้ในแต่ละพิกเซล จะมีค่าความเข้มแสงในแต่ละระดับที่แตกต่างกันไป ตั้งแต่สีขาวไปยังสีดำ เราสามารถกำหนดระดับความเข้มแสงนั้นได้ โดยใช้ค่าระดับเทา โดยปกติแล้ว ภาพระดับเทาจะมีความละเอียด (Resolution) เท่ากับ 8 บิต ซึ่งภาพจะมีค่าระดับความเข้มแสงของสีดำเท่ากับ 0 ส่วนค่าระดับความเข้มแสงของสีขาวจะมีค่าเท่ากับ 255 ($2^8\text{บิต}-1$)



Gray Level			
187	214	218	220
128	129	185	222
126	127	128	190
100	124	126	129

ภาพสี RGB (Color Image)

ในแต่ละพิกเซลของภาพสีจะเก็บค่าระดับความเข้มแสงของแม่สีหลัก 3 สีที่ซ้อนกันอยู่ ได้แก่ สีแดง (Red) สีเขียว (Green) และสีน้ำเงิน (Blue) โดยปกติแล้ว ภาพสีที่ใช้กันทั่วไปจะมีความละเอียด (Resolution) เท่ากับ สีแดง 8 บิต สีเขียว 8 บิต และ สีน้ำเงิน 8 บิต



RGB Components

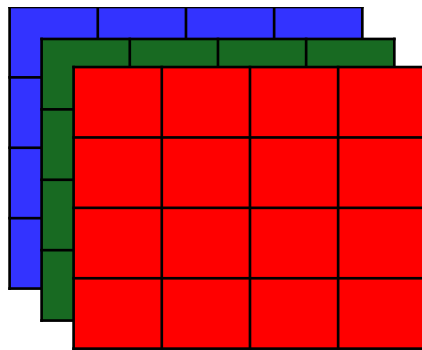
1	200	218	220
0	129	185	222
0	2	128	190
0	128	107	112
253	124	126	129
0	1	3	4

การแปลงภาพสี RGB ไปเป็นภาพ Grayscale

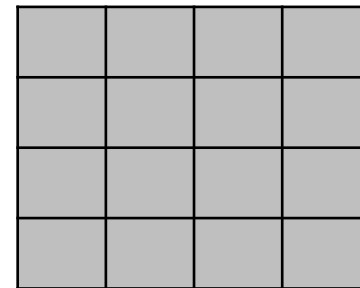
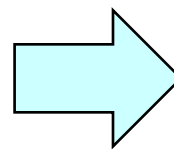
Original RGB



Grayscale Conversion



ค่าอยู่ในช่วง 0 ถึง 255



ค่าอยู่ในช่วง 0 ถึง 255

$$I_{Grayscale} = 0.299R + 0.587G + 0.114B$$

ตัวอย่างการแปลงภาพสี RGB ไปเป็นภาพ Grayscale

255	60	48	120
1	20	36	189
10	77	89	113
46	87	52	3

R

32	145	20	0
47	66	159	32
7	14	198	245
153	14	7	8

G

11	39	87	148
12	36	47	85
13	59	87	102
104	115	69	87

B

จาก
$$I_{Grayscale} = 0.299R + 0.587G + 0.114B$$

$$I_{Grayscale}(1,1) = 0.299(255) + 0.587(32) + 0.114(11) = 96.2830 \approx 96$$

$$I_{Grayscale}(1,2) = 0.299(60) + 0.587(145) + 0.114(39) = 107.5010 \approx 108$$

...

...

...

$$I_{Grayscale}(4,4) = 0.299(3) + 0.587(8) + 0.114(87) = 15.5110 \approx 16$$

96	108	36	53
29	49	109	85
9	38	153	189
115	47	28	16

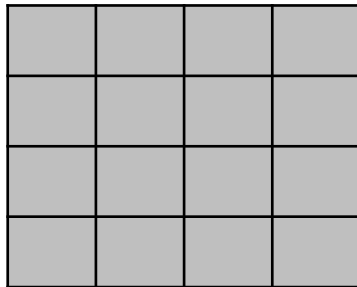
I_{Grayscale}

การแปลงภาพ Grayscale ไปเป็นภาพ Binary

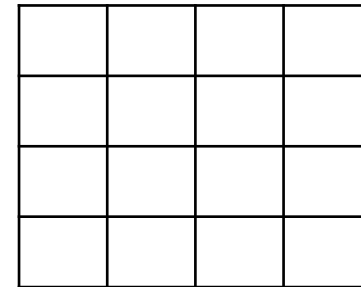
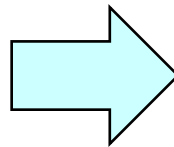
Grayscale



BW



ค่าอยู่ในช่วง 0 ถึง 255



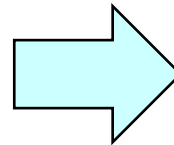
มีค่าเป็น 0 (ดำ) หรือ 1 (ขาว)
ในกรณีภาพ 8 bit สีขาวจะมีค่าเป็น 255

ตัวอย่างการแปลงภาพ Grayscale ไปเป็นภาพ Binary

96	108	36	53
29	49	109	85
9	38	153	189
115	47	28	16

$I_{Grayscale}$

$Th = 100$



0	255	0	0
0	0	255	0
0	0	255	255
255	0	0	0

I_{BW}

มีค่าเป็น 0 (ดำ) หรือ 1 (ขาว)

ในที่นี้ เป็นภาพ 8 bit สีขาวจะมีค่าเป็น 255

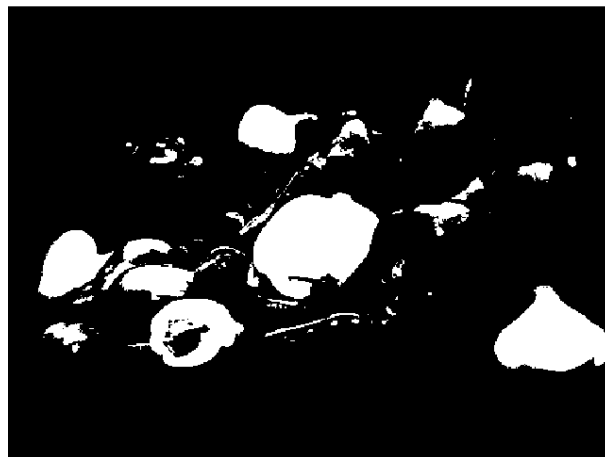
$$I_{BW} = \begin{cases} 0 & ; I_{Grayscale} \leq Th \\ 255 & ; I_{Grayscale} > Th \end{cases}$$

ตัวอย่างการแปลงภาพ Grayscale ไปเป็นภาพ Binary

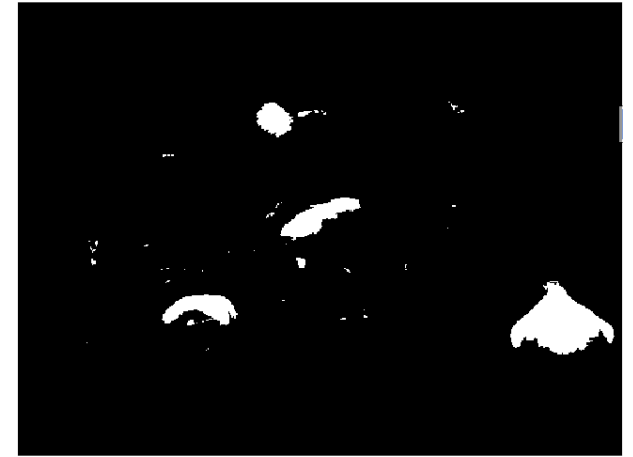
Grayscale



$Th = 80$



$Th = 140$



$Th = 200$

Dilation (การขยาย)

ถ้า A และ B เป็นเซตใน Z^2 การขยายตัวของเซต A ที่ถูกกระทำด้วยเซต B จะสามารถเขียนเป็นสัญลักษณ์ได้ดังนี้

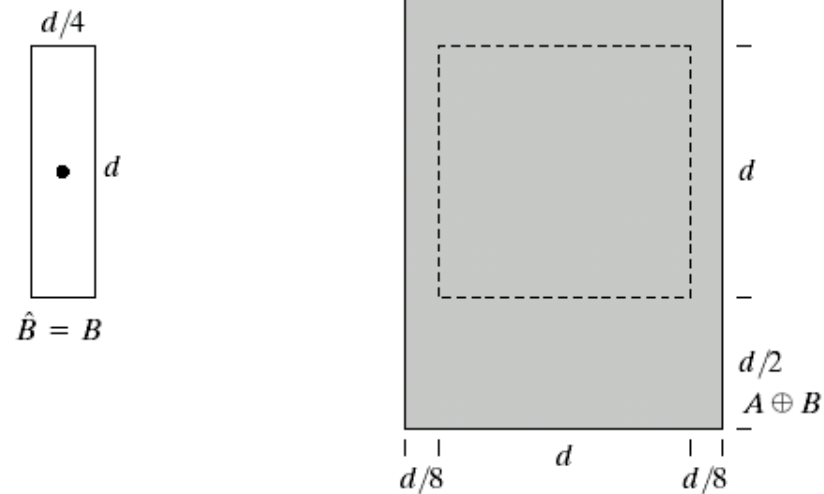
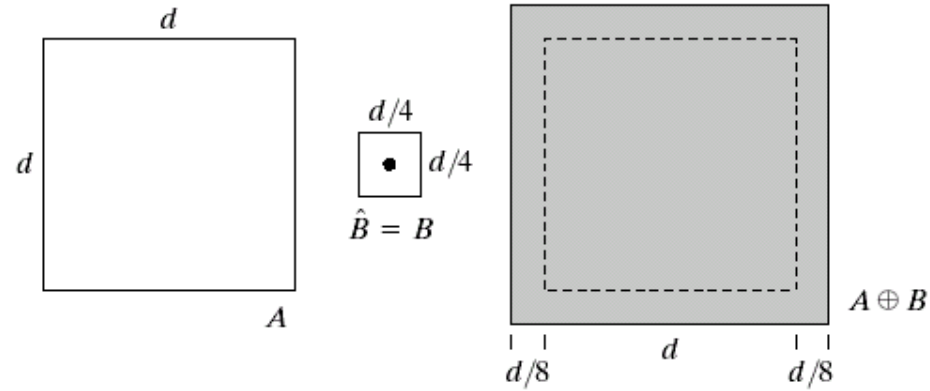
$$A \oplus B = \left\{ z \mid \left(\hat{B} \right)_z \cap A \neq \emptyset \right\}$$

หรือ

$$A \oplus B = \left\{ z \mid \left[\left(\hat{B} \right)_z \cap A \right] \in A \right\}$$

โดยที่เซต B เรียกว่า Structuring Element

ตัวอย่างการทำ Dilation



การคำนวณ Dilation ตัวอย่างที่ 1

0	0	0	0	0
0	1	1	0	0
0	0	1	1	0
0	1	1	0	0
0	0	0	0	0

ภาพขาวดำขนาด 5x5 พิกเซล

เซต A

0	1	0
1	1	1
0	1	0

Structuring Element ขนาด 3x3

เซต B

เราจะเคลื่อนที่ B ไปทุกๆ พิกเซลที่เป็นสมาชิกของ A โดยมีจุดศูนย์กลางกลางของ B เป็นจุดพิจารณา พิกเซลใดที่เกิดจากการ Union กันระหว่าง B กับส่วนหนึ่งของ A ที่ B เคลื่อนที่ผ่าน ค่าของพิกเซลในตำแหน่งนั้นของภาพเอาต์พุตจะมีค่าเป็น 1

การคำนวณ Dilation ตัวอย่างที่ 1

จุดที่ 1

B

0	1	0
1	①	1
0	1	0

เราจะเคลื่อนที่ B ไปทั้งหมด 6 จุด

เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 6 จุดนั่นเอง

A

0	1	0	0	0
1	①	1	0	0
0	1	1	1	0
0	1	1	0	0
0	0	0	0	0

ภาพอินพุต

0	1	0	0	0
1	1	1	0	0
0	1	1	1	0
0	1	1	0	0
0	0	0	0	0

ภาพเอาต์พุต

การคำนวณ Dilation ตัวอย่างที่ 1

จุดที่ 2

B

0	1	0
1	①	1
0	1	0

เราจะเคลื่อนที่ B ไปทั้งหมด 6 จุด

เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 6 จุดนั่นเอง

A

0	0	1	0	0
0	1	①	1	0
0	0	1	1	0
0	1	1	0	0
0	0	0	0	0

ภาพอินพุต

0	1	1	0	0
1	1	1	1	0
0	1	1	1	0
0	1	1	0	0
0	0	0	0	0

ภาพเอาต์พุต

การคำนวณ Dilation ตัวอย่างที่ 1

จุดที่ 3

B

0	1	0
1	①	1
0	1	0

เราจะเคลื่อนที่ B ไปทั้งหมด 6 จุด

เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 6 จุดนั่นเอง

A

0	0	0	0	0
0	1	1	0	0
0	1	①	1	0
0	1	1	0	0
0	0	0	0	0

ภาพอินพุต

0	1	1	0	0
1	1	1	1	0
0	1	1	1	0
0	1	1	0	0
0	0	0	0	0

ภาพเอาต์พุต

การคำนวณ Dilation ตัวอย่างที่ 1

จุดที่ 4

B

0	1	0
1	①	1
0	1	0

เราจะเคลื่อนที่ B ไปทั้งหมด 6 จุด

เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 6 จุดนั่นเอง

A

0	0	0	0	0
0	1	1	1	0
0	0	1	①	1
0	1	1	1	0
0	0	0	0	0

ภาพอินพุต

0	1	1	0	0
1	1	1	1	0
0	1	1	1	1
0	1	1	1	0
0	0	0	0	0

ภาพเอาต์พุต

การคำนวณ Dilation ตัวอย่างที่ 1

จุดที่ 5

B

0	1	0
1	①	1
0	1	0

เราจะเคลื่อนที่ B ไปทั้งหมด 6 จุด

เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 6 จุดนั่นเอง

A

0	0	0	0	0
0	1	1	0	0
0	1	1	1	0
1	①	1	0	0
0	1	0	0	0

ภาพอินพุต

0	1	1	0	0
1	1	1	1	0
0	1	1	1	1
1	1	1	1	0
0	1	0	0	0

ภาพเอาต์พุต

การคำนวณ Dilation ตัวอย่างที่ 1

จุดที่ 6 (จุดสุดท้าย)

B

0	1	0
1	①	1
0	1	0

เราจะเคลื่อนที่ B ไปทั้งหมด 6 จุด
เนื่องจาก A มีพิกเซลที่เป็น 1
จำนวน 6 จุดนั่นเอง

A

0	0	0	0	0
0	1	1	0	0
0	0	1	1	0
0	1	①	1	0
0	0	1	0	0

ภาพอินพุต

0	1	1	0	0
1	1	1	1	0
0	1	1	1	1
1	1	1	1	0
0	1	1	0	0

ภาพเอาต์พุต

การคำนวณ Dilation ตัวอย่างที่ 1

สรุปภาพเอาต์พุตที่ได้ เมื่อ B เคลื่อนที่ครบ 6 จุดของ A

B

0	1	0
1	1	1
0	1	0

ภาพเอาต์พุตที่ได้ วัตถุจะมีขนาดใหญ่ขึ้น
เนื่องจากการทำ Dilation เป็นการขยายนั่นเอง

A

0	0	0	0	0
0	1	1	0	0
0	0	1	1	0
0	1	1	0	0
0	0	0	0	0

ภาพอินพุต

0	1	1	0	0
1	1	1	1	0
0	1	1	1	1
1	1	1	1	0
0	1	1	0	0

ภาพเอาต์พุต

การคำนวณ Dilation ตัวอย่างที่ 2

0	0	0	0	0
0	0	1	1	0
0	0	1	0	0
0	1	1	0	0
0	0	0	0	0

ภาพขาวดำขนาด 5x5 พิกเซล

เซต *A*

0	1	0
0	1	1
0	0	0

Structuring Element ขนาด 3x3

เซต *B*

เราจะเคลื่อนที่ *B* ไปทั้งหมด 5 จุด

เนื่องจาก *A* มีพิกเซลที่เป็น 1

จำนวน 5 จุดนั่นเอง

การคำนวณ Dilation ตัวอย่างที่ 2

จุดที่ 1

B

0	1	0
0	1	1
0	0	0

เราจะเคลื่อนที่ B ไปทั้งหมด 5 จุด

เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 5 จุดนั่นเอง

A

0	0	1	0	0
0	0	1	1	0
0	0	1	0	0
0	1	1	0	0
0	0	0	0	0

ภาพอินพุต

0	0	1	0	0
0	0	1	1	0
0	0	1	0	0
0	1	1	0	0
0	0	0	0	0

ภาพเอาต์พุต

การคำนวณ Dilation ตัวอย่างที่ 2

จุดที่ 2

B

0	1	0
0	1	1
0	0	0

เราจะเคลื่อนที่ B ไปทั้งหมด 5 จุด

เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 5 จุดนั่นเอง

A

0	0	0	1	0
0	0	1	1	1
0	0	1	0	0
0	1	1	0	0
0	0	0	0	0

ภาพอินพุต

0	0	1	1	0
0	0	1	1	1
0	0	1	0	0
0	1	1	0	0
0	0	0	0	0

ภาพเอาต์พุต

การคำนวณ Dilation ตัวอย่างที่ 2

จุดที่ 3

B

0	1	0
0	1	1
0	0	0

เราจะเคลื่อนที่ B ไปทั้งหมด 5 จุด

เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 5 จุดนั่นเอง

A

0	0	0	0	0
0	0	1	1	0
0	0	1	1	0
0	1	1	0	0
0	0	0	0	0

ภาพอินพุต

0	0	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0
0	0	0	0	0

ภาพเอาต์พุต

การคำนวณ Dilation ตัวอย่างที่ 2

จุดที่ 4

B

0	1	0
0	1	1
0	0	0

เราจะเคลื่อนที่ B ไปทั้งหมด 5 จุด

เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 5 จุดนั่นเอง

A

0	0	0	0	0
0	0	1	1	0
0	1	1	0	0
0	1	1	0	0
0	0	0	0	0

ภาพอินพุต

0	0	1	1	0
0	0	1	1	1
0	1	1	1	0
0	1	1	0	0
0	0	0	0	0

ภาพเอาต์พุต

การคำนวณ Dilation ตัวอย่างที่ 2

จุดที่ 5 (จุดสุดท้าย)

B

0	1	0
0	1	1
0	0	0

เราจะเคลื่อนที่ B ไปทั้งหมด 5 จุด

เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 5 จุดนั่นเอง

A

0	0	0	0	0
0	0	1	1	0
0	0	1	0	0
0	1	1	1	0
0	0	0	0	0

ภาพอินพุต

0	0	1	1	0
0	0	1	1	1
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

ภาพเอาต์พุต

การคำนวณ Dilation ตัวอย่างที่ 2

รูปภาพเอาต์พุตที่ได้ เมื่อ B เคลื่อนที่ครบ 5 จุดของ A

B

0	1	0
0	1	1
0	0	0

ภาพเอาต์พุตที่ได้ วัตถุจะมีขนาดใหญ่ขึ้น
เนื่องจากการทำ Dilation เป็นการขยายนั่นเอง

A

0	0	0	0	0
0	0	1	1	0
0	0	1	0	0
0	1	1	0	0
0	0	0	0	0

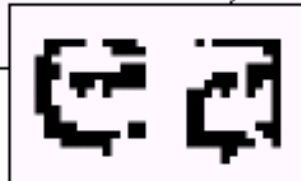
ภาพอินพุต

0	0	1	1	0
0	0	1	1	1
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

ภาพเอาต์พุต

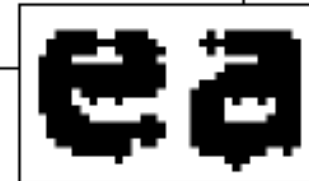
ตัวอย่างการนำ Dilation ไปใช้งาน

Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.



Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

A



B

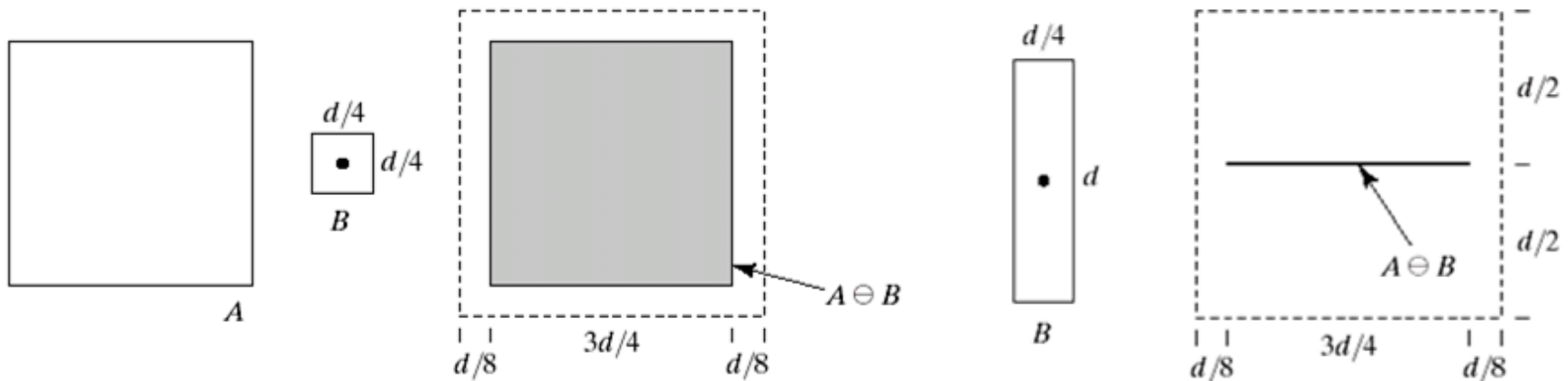
0	1	0
1	1	1
0	1	0

Erosion (การกร่อน)

ถ้า A และ B เป็นเซตใน Z^2 การกร่อนของเซต A ที่ถูกกระทำด้วยเซต B จะสามารถเขียนเป็นสัญลักษณ์ได้ดังนี้

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

โดยที่เซต B เรียกว่า Structuring Element



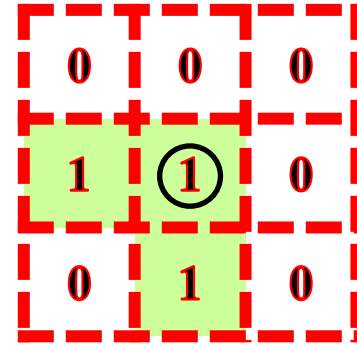
การคำนวณ Erosion ตัวอย่างที่ 1

0	0	0	0	0
0	1	1	0	0
0	1	1	1	0
0	0	1	1	0
0	0	0	0	0

ภาพขาวดำขนาด 5x5 พิกเซล

เซต A

เราจะเคลื่อนที่ B ไปทุกๆ พิกเซลที่เป็นสมาชิกของ A โดยมีจุดศูนย์กลางของ B เป็นจุดพิจารณา สมาชิกจุดใดของ A ที่เมื่อเราวาง B ลงไปแล้วสามารถทำให้ B อยู่ภายในเซต A ได้ทั้งหมด สมาชิกจุดนั้นของ A จะถูกคงค่าไว้เพื่อให้ปรากฏในภาพเอาต์พุต ในทางตรงกันข้ามถ้าเราวาง B ลงไปแล้วและไม่สามารถทำให้ B อยู่ภายในเซต A ได้ทั้งหมด จุดนั้นของ A จะถูกลบทิ้ง นั่นคือจะไม่ปรากฏในภาพเอาต์พุตนั่นเอง

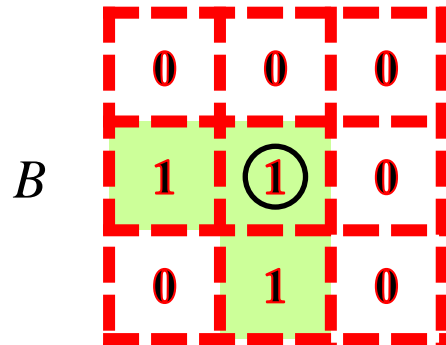


Structuring Element ขนาด 3x3

เซต B

การคำนวณ Erosion ตัวอย่างที่ 1

จุดที่ 1

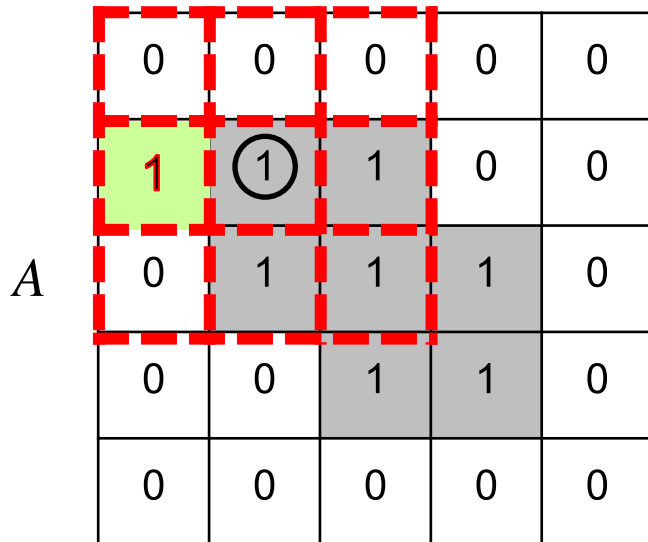


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

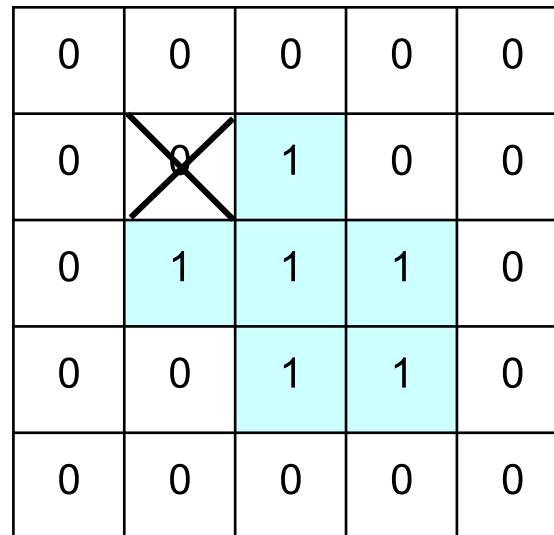
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B ไม่อยู่ใน A ดังนั้น จุดที่ 1 ของ A จะโดนลบทิ้ง



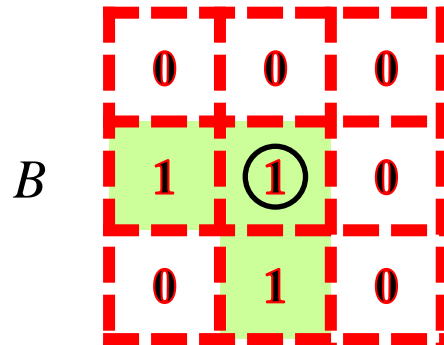
ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 1

จุดที่ 2

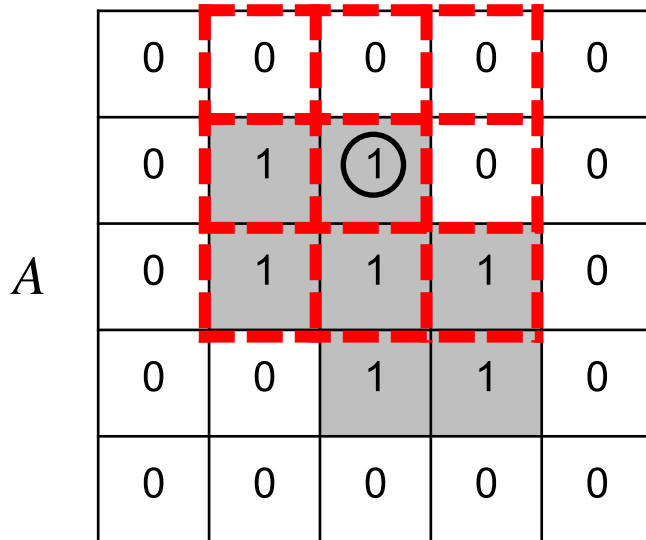


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

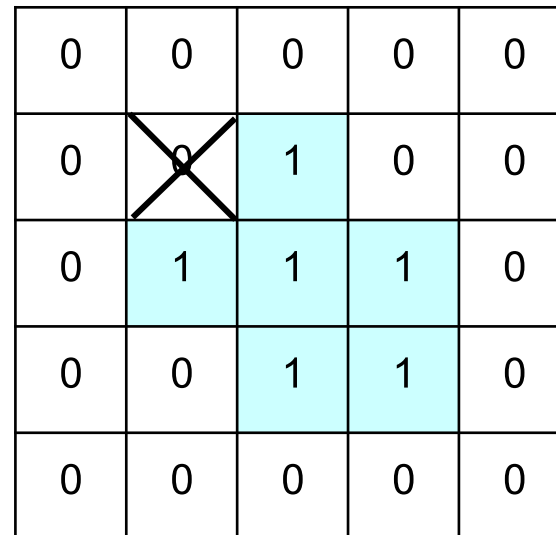
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B อยู่ใน A ดังนั้น จุดที่ 2 ของ A จะไม่โดนลบทิ้ง



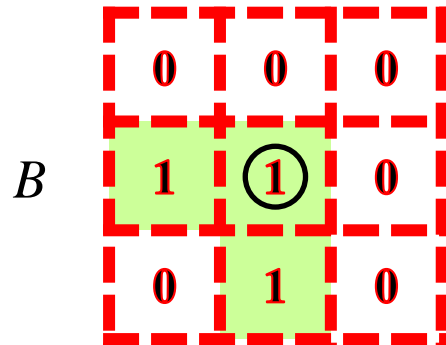
ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 1

จุดที่ 3

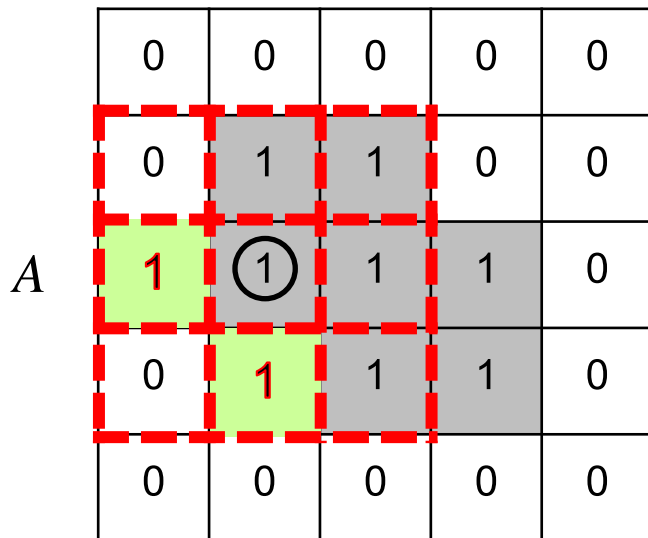


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

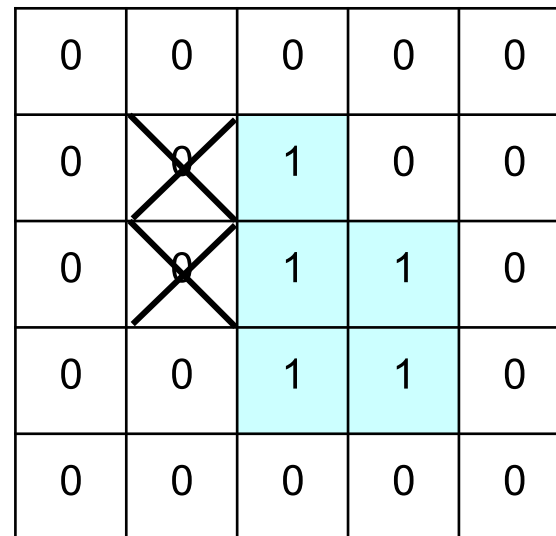
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B ไม่อยู่ใน A ดังนั้น จุดที่ 3 ของ A จะโดนลบทิ้ง



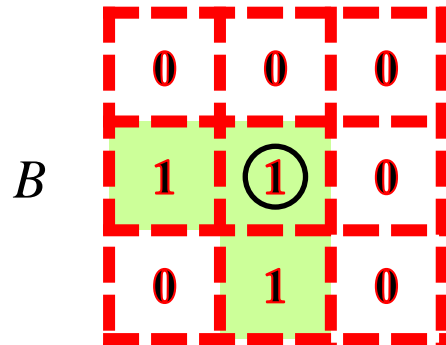
ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 1

จุดที่ 4

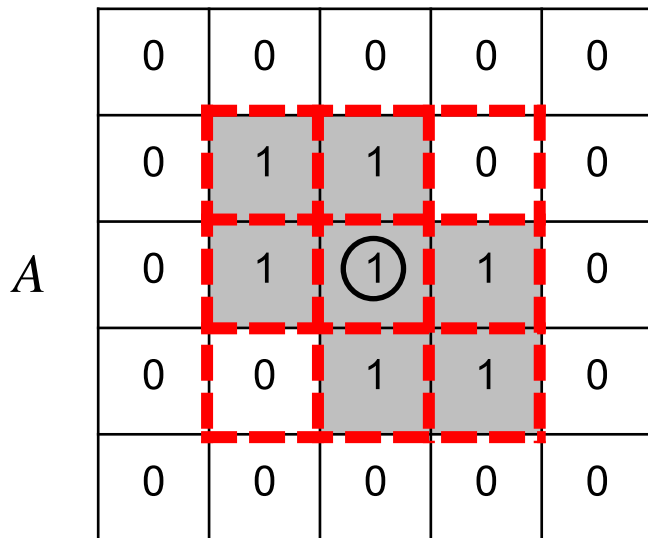


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

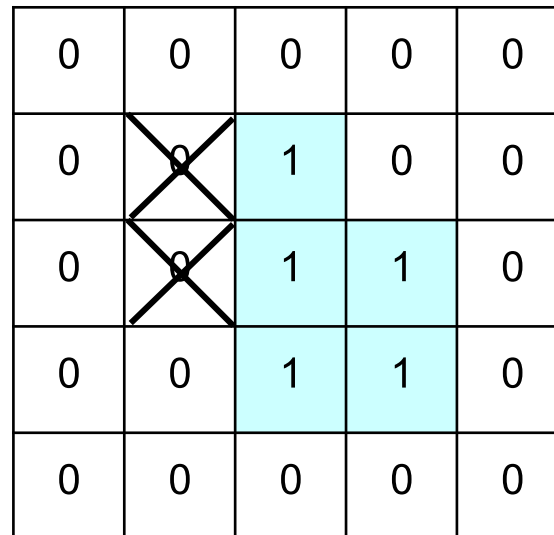
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B อยู่ใน A ดังนั้น จุดที่ 4 ของ A จะไม่โดนลบทิ้ง



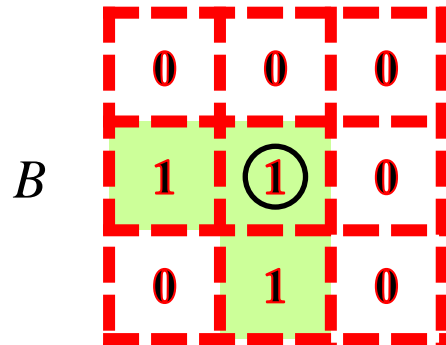
ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 1

จุดที่ 5

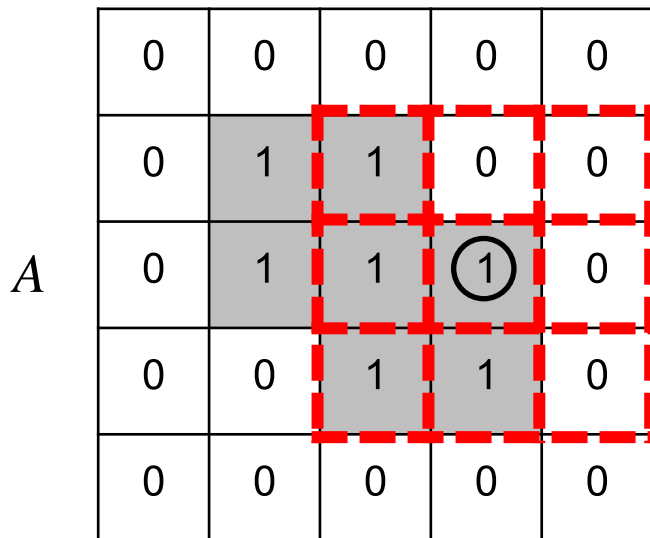


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

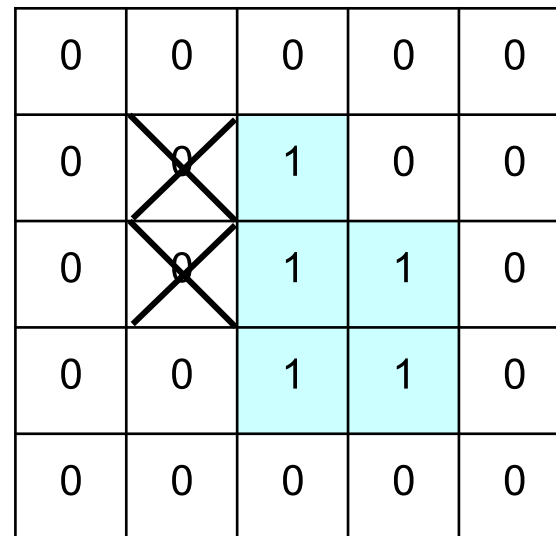
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B อยู่ใน A ดังนั้น จุดที่ 5 ของ A จะไม่โดนลบทิ้ง



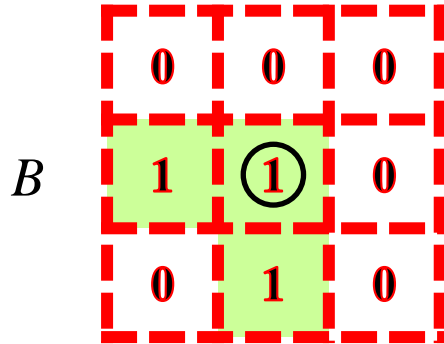
ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 1

จุดที่ 6

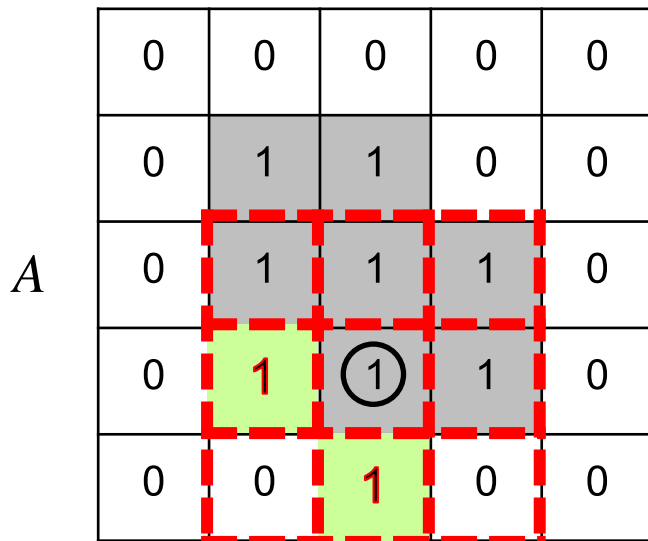


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

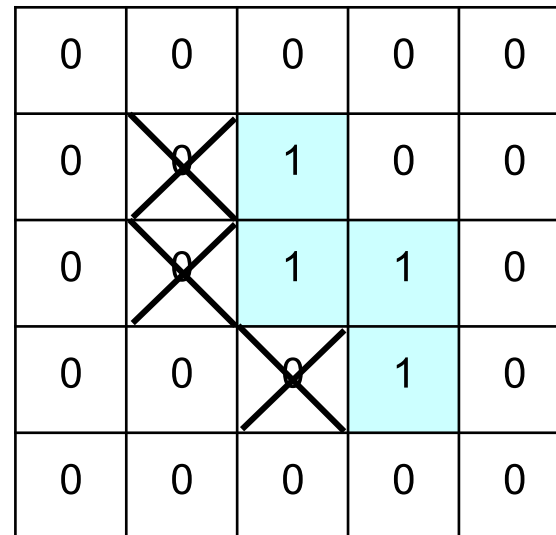
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B ไม่อยู่ใน A ดังนั้น จุดที่ 6 ของ A จะโดนลบทิ้ง



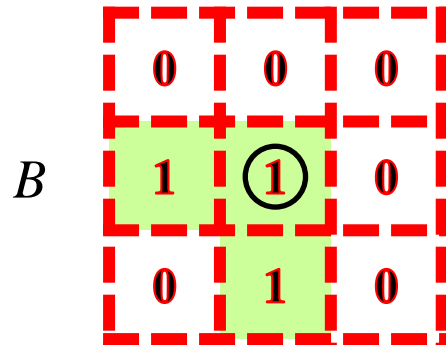
ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 1

จุดที่ 7 (จุดสุดท้าย)

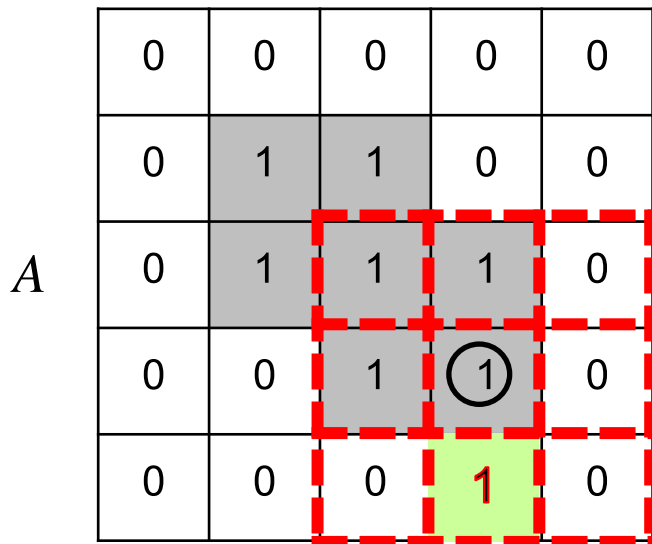


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

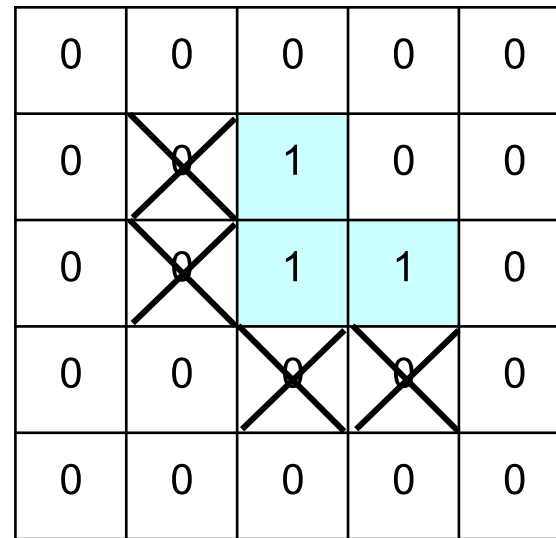
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B ไม่อยู่ใน A ดังนั้น จุดที่ 7 ของ A จะโดนลบทิ้ง



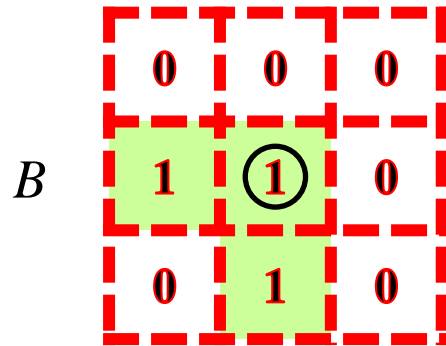
ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 1

รูปภาพเอาต์พุตที่ได้ เมื่อ B เคลื่อนที่ครบ 7 จุดของ A



ภาพเอาต์พุตที่ได้ วัตถุจะมีขนาดเล็กลง
เนื่องจากการทำ Erosion เป็นการกร่อนนั่นเอง

A

0	0	0	0	0
0	1	1	0	0
0	1	1	1	0
0	0	1	1	0
0	0	0	0	0

ภาพอินพุต

0	0	0	0	0
0	0	1	0	0
0	0	1	1	0
0	0	0	0	0
0	0	0	0	0

ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 2

0	0	0	0	0
0	0	1	1	0
0	1	1	1	0
0	0	1	1	0
0	0	0	0	0

ภาพขาวดำขนาด 5x5 พิกเซล

เซต *A*

0	1	0
0	1	1
0	1	0

Structuring Element ขนาด 3x3

เซต *B*

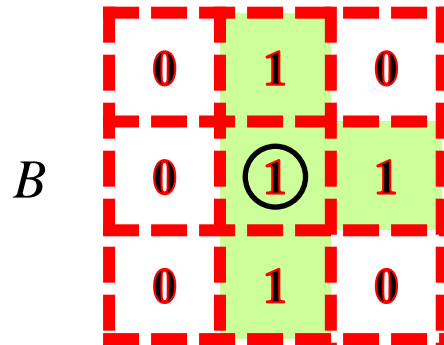
เราจะเคลื่อนที่ *B* ไปทั้งหมด 7 จุด

เนื่องจาก *A* มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

การคำนวณ Erosion ตัวอย่างที่ 2

จุดที่ 1

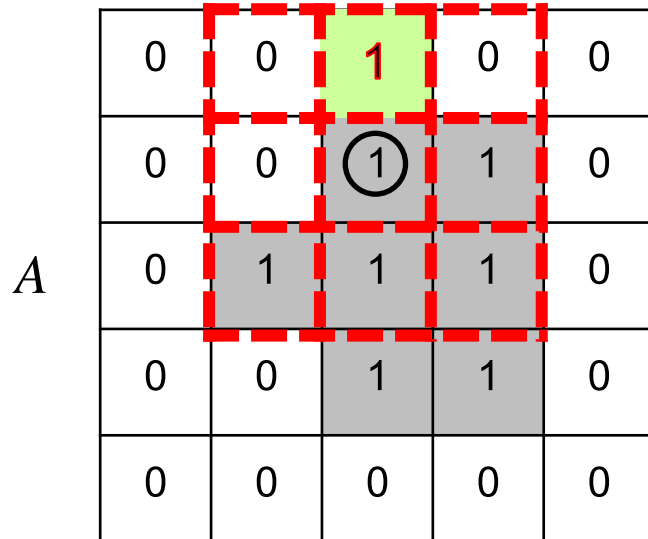


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

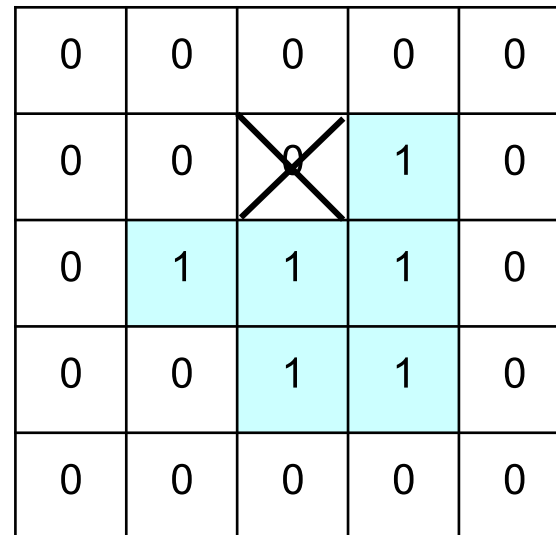
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B ไม่อยู่ใน A ดังนั้น จุดที่ 1 ของ A จะโดนลบทิ้ง



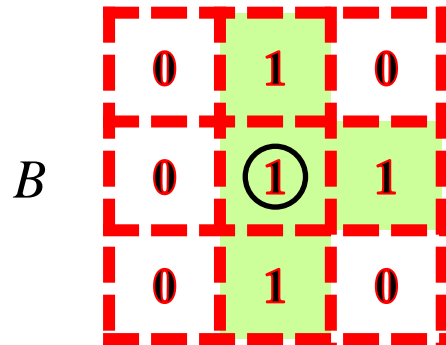
ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 2

จุดที่ 2

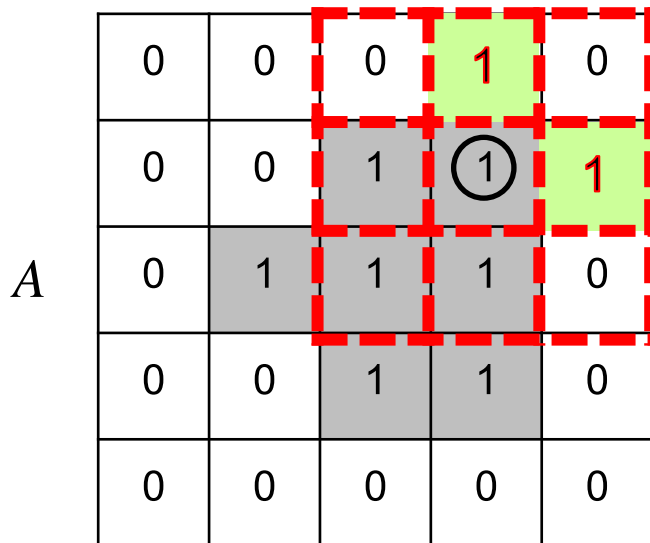


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

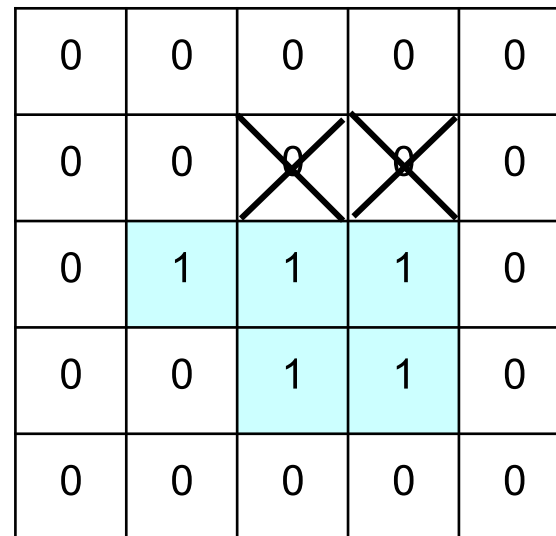
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B ไม่อยู่ใน A ดังนั้น จุดที่ 2 ของ A จะโดนลบทิ้ง



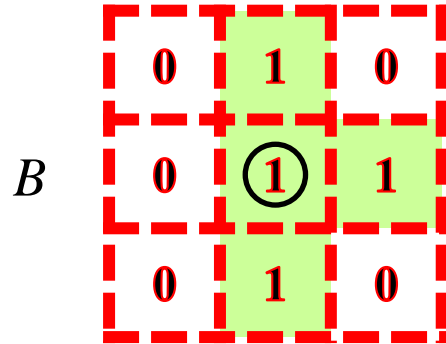
ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 2

จุดที่ 3

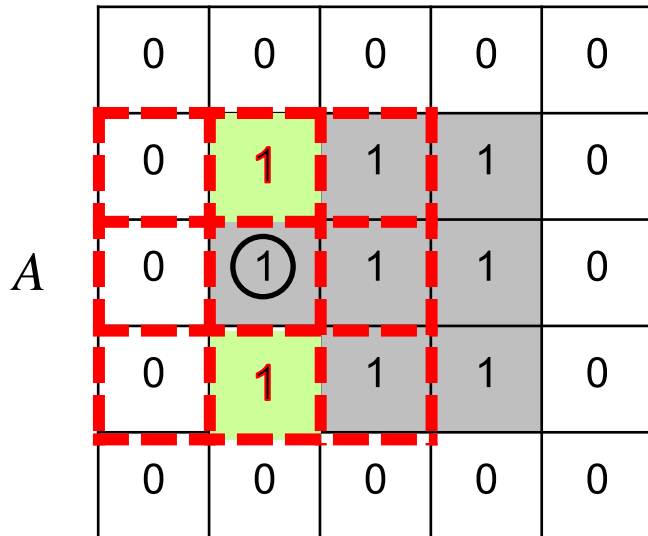


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

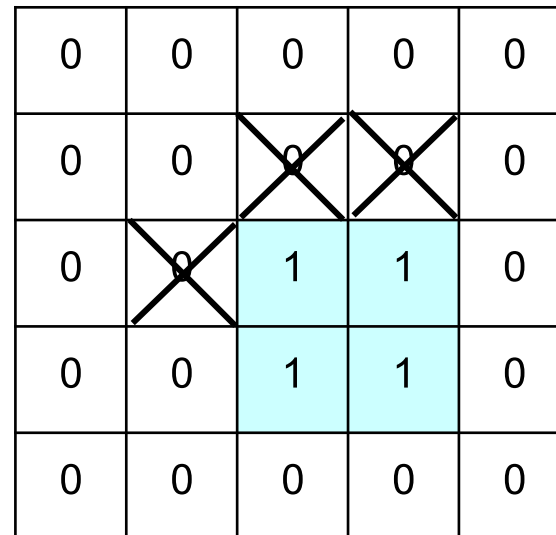
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B ไม่อยู่ใน A ดังนั้น จุดที่ 3 ของ A จะโดนลบทิ้ง



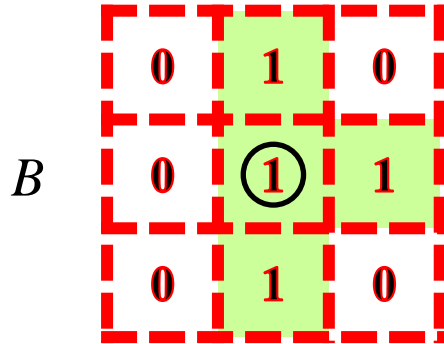
ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 2

จุดที่ 4

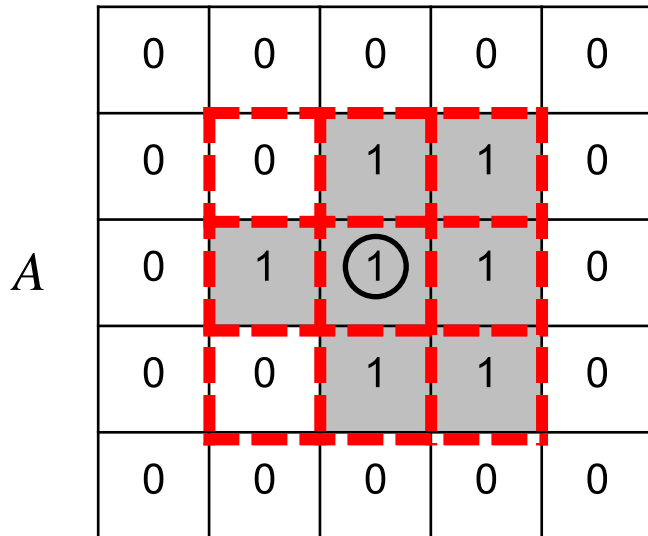


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

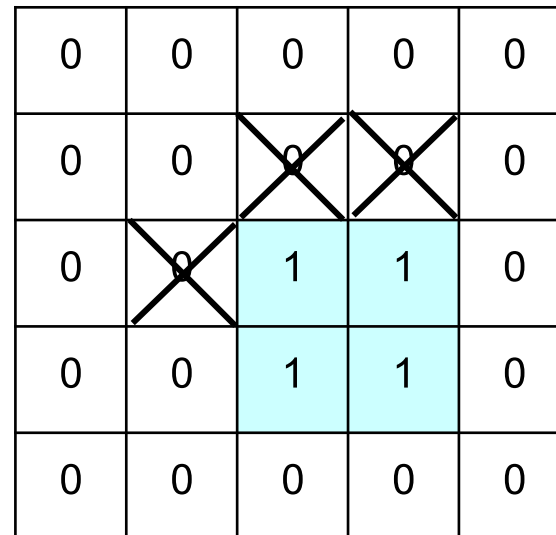
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B อยู่ใน A ดังนั้น จุดที่ 4 ของ A จะไม่โดนลบทิ้ง



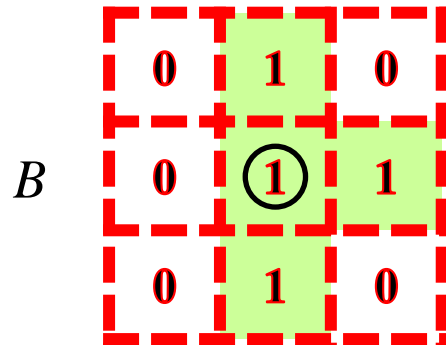
ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 2

จุดที่ 5

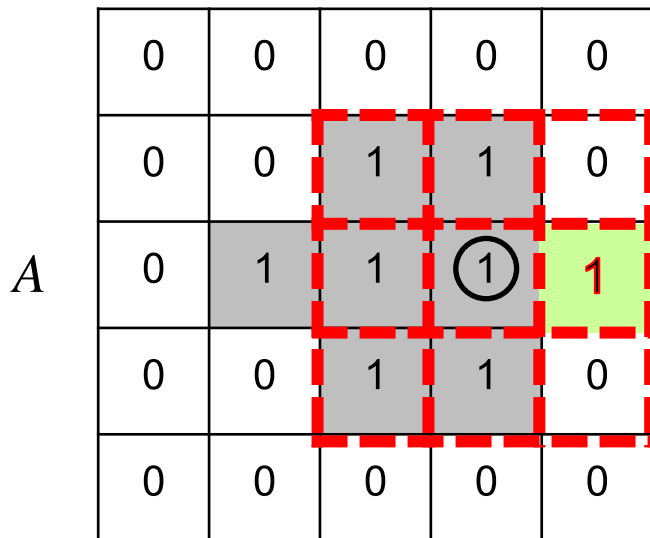


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

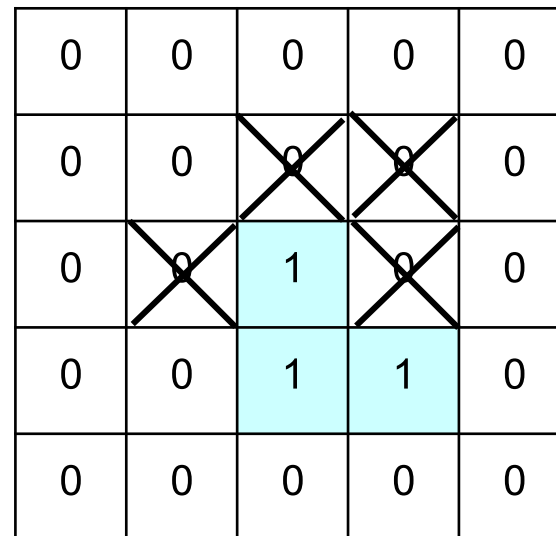
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B ไม่อยู่ใน A ดังนั้น จุดที่ 5 ของ A จะโดนลบทิ้ง



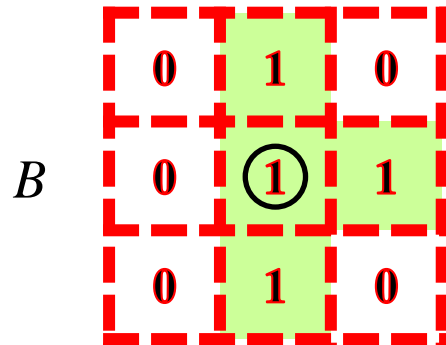
ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 2

จุดที่ 6

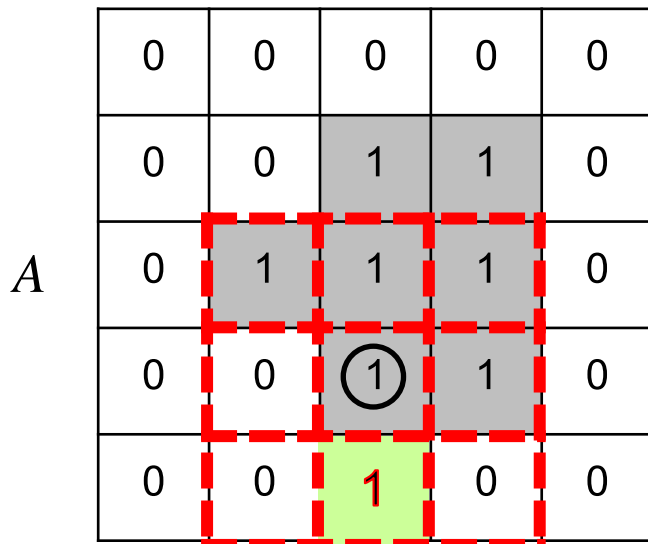


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

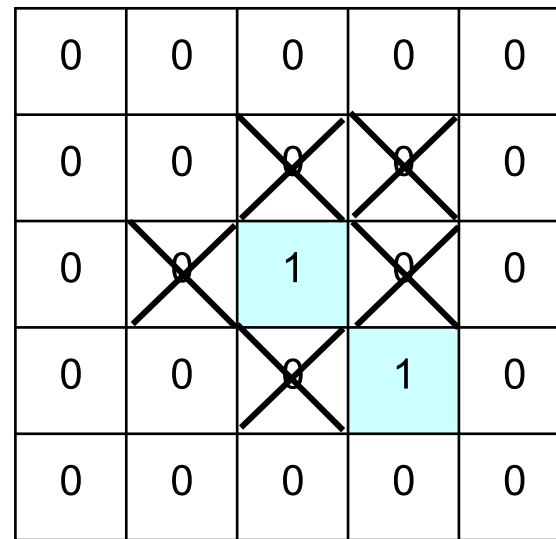
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B ไม่อยู่ใน A ดังนั้น จุดที่ 6 ของ A จะโดนลบทิ้ง



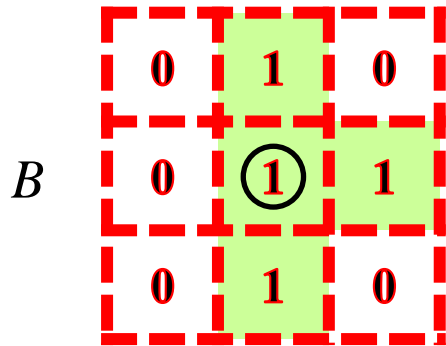
ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 2

จุดที่ 7 (จุดสุดท้าย)

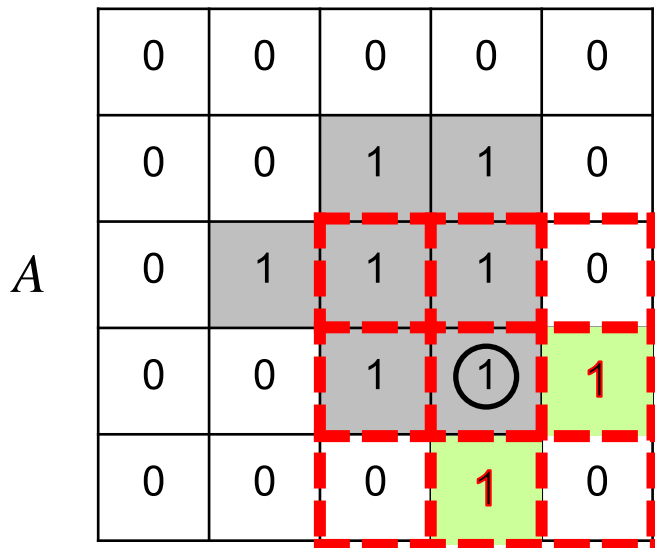


เราจะเคลื่อนที่ B ไปทั้งหมด 7 จุด

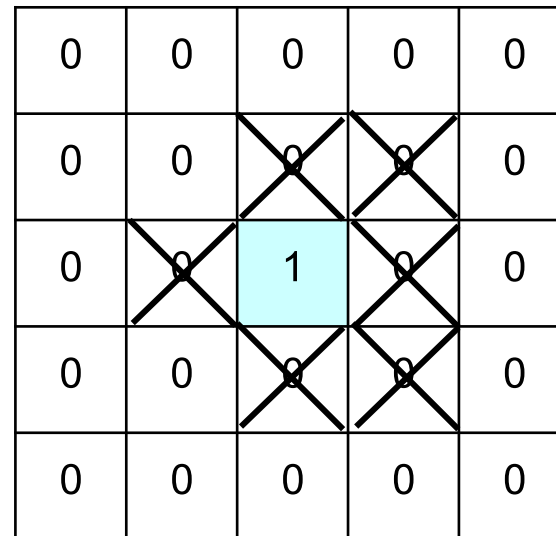
เนื่องจาก A มีพิกเซลที่เป็น 1

จำนวน 7 จุดนั่นเอง

B ไม่อยู่ใน A ดังนั้น จุดที่ 7 ของ A จะโดนลบทิ้ง



ภาพอินพุต



ภาพเอาต์พุต

การคำนวณ Erosion ตัวอย่างที่ 2

รูปภาพเอาต์พุตที่ได้ เมื่อ B เคลื่อนที่ครบ 7 จุดของ A

B

0	1	0
0	1	1
0	1	0

ภาพเอาต์พุตที่ได้ วัตถุจะมีขนาดเล็กลง
เนื่องจากการทำ Erosion เป็นการกร่อนนั่นเอง

A

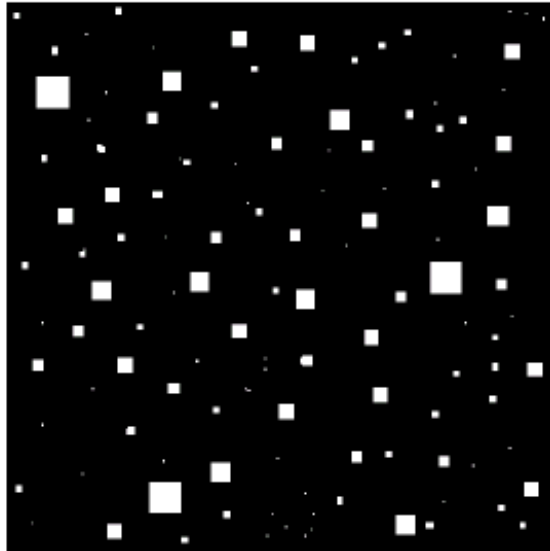
0	0	0	0	0
0	0	1	1	0
0	1	1	1	0
0	0	1	1	0
0	0	0	0	0

ภาพอินพุต

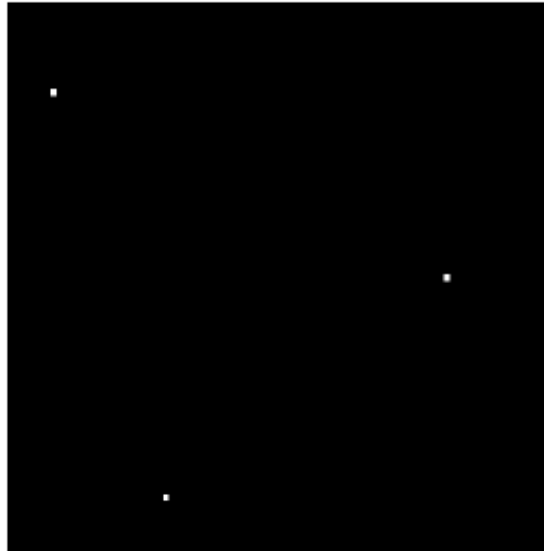
0	0	0	0	0
0	0	0	0	0
0	0	1	0	0
0	0	0	0	0
0	0	0	0	0

ภาพเอาต์พุต

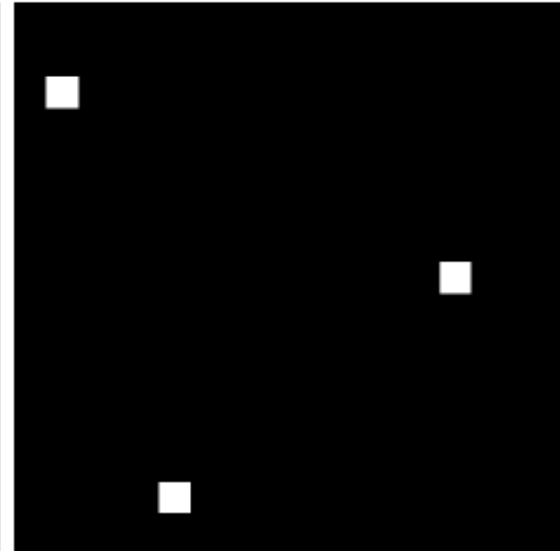
ตัวอย่างการนำ Erosion และ Dilation ไปใช้งาน



ภาพอินพุต



ภาพที่ได้
จากการทำ Erosion



ภาพที่ได้
จากการทำ Dilation

Opening (การเปิด)

การเปิดของเซต A โดย B ทำได้โดยการกร่อน A ด้วย B แล้วตามด้วยการขยายผลลัพธ์ด้วย B

$$A \circ B = (A \ominus B) \oplus B$$

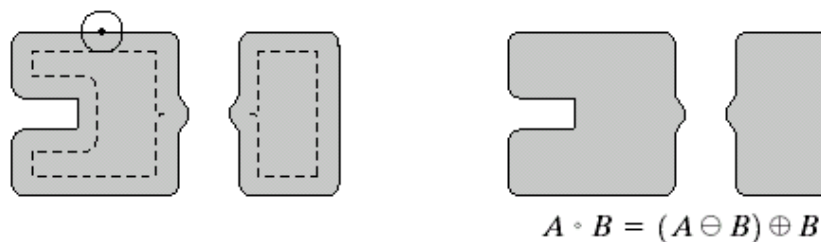
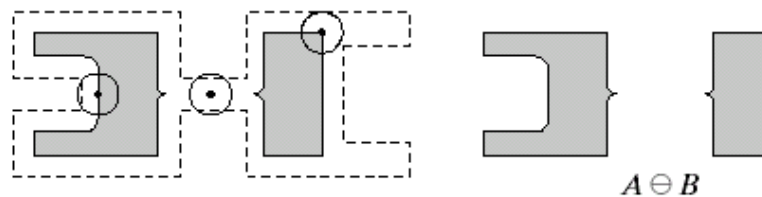
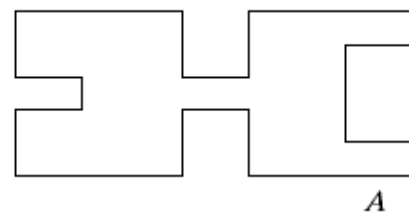
Closing (การปิด)

การปิดของเซต A โดย B ทำได้โดยการขยาย A ด้วย B แล้วตามด้วยการกร่อนผลลัพธ์ด้วย B

$$A \bullet B = (A \oplus B) \ominus B$$

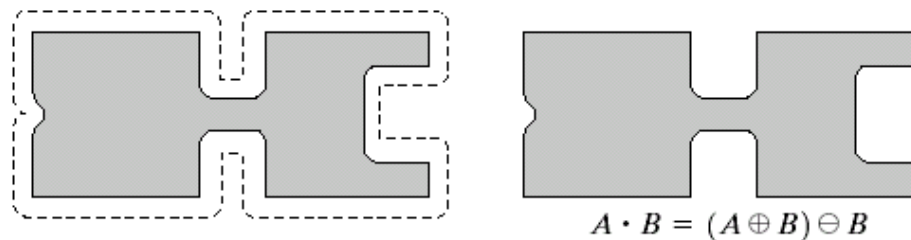
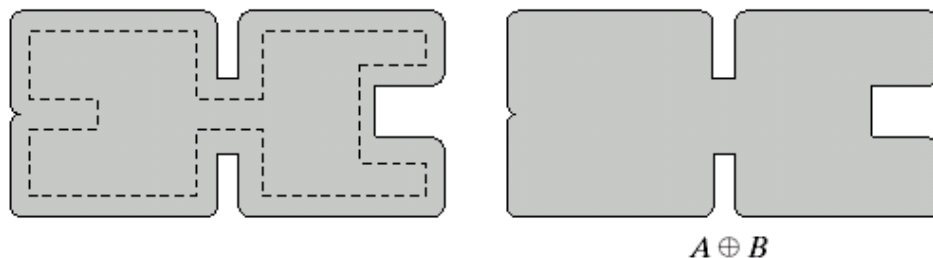
Opening (การเปิด)

$$A \circ B = (A \ominus B) \oplus B$$



Closing (การปิด)

$$A \bullet B = (A \oplus B) \ominus B$$



เทคนิคการประมวลผลภาพด้วย OpenCV

ชื่อไฟล์	โค้ดโปรแกรม
tncv001.py	การแปลงภาพ RGB เป็น Grayscale
tncv002.py	การแปลงภาพ Grayscale เป็นภาพขาว-ดำ
tncv003.py	การปรับขนาดภาพ
tncv004.py	การ crop ภาพบางส่วน
tncv005.py	การปรับความสว่างของภาพ
tncv006.py	การหมุนภาพ
tncv007.py	การเบลอภาพ
tncv008.py	การทำ erosion dilation opening closing

โค้ดโปรแกรมตัวอย่าง Image → text → speech

ชื่อไฟล์	โค้ดโปรแกรม
img2speech001.py	การเปลี่ยนข้อความเป็นเสียงพูด
img2speech002.py	การแปลงภาพเป็นเสียงพูด โดยใช้ EasyOCR
img2speech003.py	การแปลงภาพเป็นเสียงพูด โดยใช้ Gemini บรรยายภาพ
img2speech004.py	การแปลงภาพเป็นเสียงพูด โดยใช้ Gemini ตรวจสอบข้อความในภาพ

รายการอ้างอิง

- [1] <https://www.python.org>
- [2] <https://opencv.org>
- [3] <https://www.canva.com/> (PRO)
- [4] <https://gemini.google.com/app>
- [5] <https://github.com/JaidedAI/EasyOCR>
- [6] <https://www.jaided.ai/easyocr/tutorial>
- [7] <https://tesseract-ocr.github.io>
- [8] Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing Second Edition, Prentice Hall, 2001.
- [9] Alberto Fernandez Villan, “Mastering OpenCV 4 with Python”, Packt Publishing Ltd., March 2019.



ขอบคุณครับ

