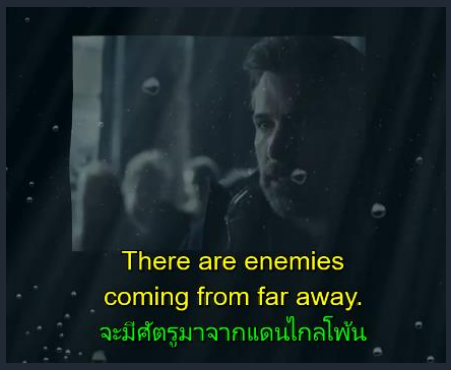
Subtitle Tools

บทบรรยายใต้ภาพ (subtitle) ช่วยเพิ่มความเข้าใจในการชมภาพยนตร์ได้มากขึ้น ดังตัวอย่างการแสดงบทบรรยายภาษาอังกฤษ ภาษาไทย และทั้งไทยและอังกฤษ ข้างล่างนี้

ในบางกรณีผู้สร้างบทบรรยายจะใส่ไม่เพียงบทพูดเท่านั้น แต่ยังเพิ่มคำบรรยายอื่น ๆ เช่น ใครพูด ลักษณะเสียง อื่น ๆ เพื่อให้ผู้ชมที่บกพร่องทางการได้ยินสามารถเข้าใจเนื้อหาได้สมบูรณ์ขึ้น เรียกบทบรยายแบบนี้ว่า แบบ SDH (Subtitles for the deaf and hard of hearing) ดังตัวอย่างข้างล่างนี้

ผู้สร้างบทบรรยายยังสามารถใส่สีสัน และลักษณะของอักษร เพื่อความชัดเจน ได้อีกด้วย ดังตัวอย่างข้างล่างนี้



รูปแบบของแฟ้ม .srt

ซอฟต์แวร์แสดงภาพยนตร์ทั้งหลาย จะให้ผู้ชมเลือกแฟ้มบทบรรยายได้ รูปแบบบทบรรยายใต้ภาพที่ได้รับความนิยมรูปแบบหนึ่งคือ **`srt`** (SubRip subTitle https://en.wikipedia.org/wiki/SubRip) เป็นแฟ้มข้อความ ภายในประกอบด้วยบทบรรยายหลาย ๆ ข้อความ เรียงตามเวลาที่เริ่มและเลิกแสดงข้อความ แต่ละบท มีอย่างน้อย 4 บรรทัด ดังนี้

- เลขลำดับของบทบรรยาย
- เวลาเริ่มแสดง **`-->`** เวลาเลิกแสดง

  เวลาอยู่ในรูปแบบ **`HH:MM:SS,ms`** แทนเลขชั่วโมง เลขนาที เลขวินาที และเลขมิลลิวินาที สามจำนวนแรกเป็นเลขสองหลัก ส่วนเลขมิลลิวินาทีเป็นเลขสามหลัก เช่น **`00:12:02,103`**
- ข้อความบรรยาย มีได้หลายบรรทัด โดยบรรทัดสุดท้ายต้องเป็นบรรทัดว่าง

เช่น ข้อความในตารางข้างล่างนี้ ทางซ้ายแสดงบทบรรยายปกติทั่วไป ส่วนทางขวาเป็นแบบ SDH และยังมีการใส่สีกำกับข้อความต่าง ๆ ด้วย

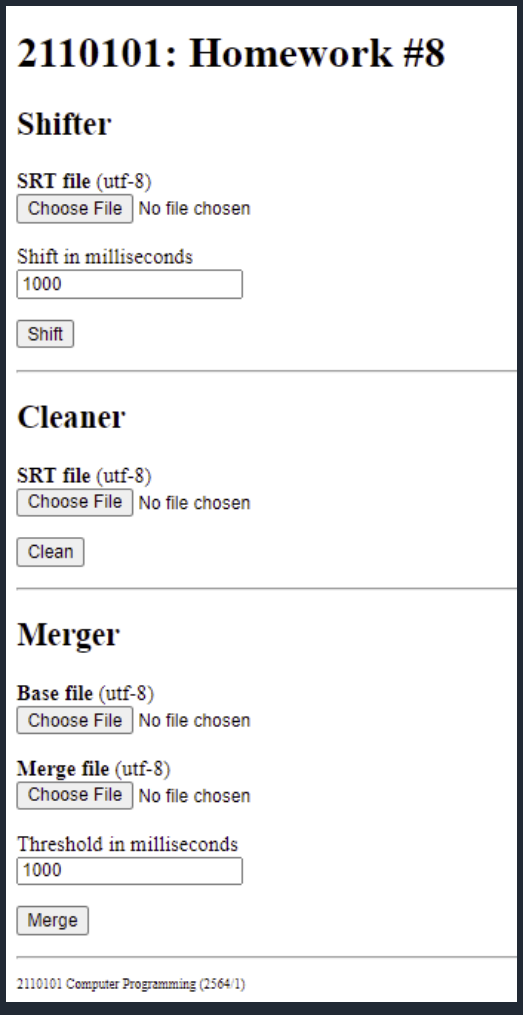
```
1
00:00:05,563 --> 00:00:08,525
Bruce Vayne.

2
00:00:08,608 --> 00:00:10,276
Bruce Wayne.

3
00:00:28,086 --> 00:00:29,629
Talk.

4
00:00:30,922 --> 00:00:32,549
I believe there is a stranger.

5
00:00:32,632 --> 00:00:35,343
Comes to this village
from the sea.
```

```
1
00:00:05,243 --> 00:00:08,205
[man] <font color="#ffff00">Bruce Vayne.</font>

2
00:00:08,288 --> 00:00:09,956
<font color="#ffff00">Bruce Wayne. </font>

3
00:00:11,416 --> 00:00:12,667
[speaking Icelandic]

4
00:00:21,760 --> 00:00:23,053
[speaks Icelandic]

5
00:00:27,766 --> 00:00:29,309
[in English] <font color="#ffff00">Talk.</font>

6
00:00:30,602 --> 00:00:32,229
<font color="#ffff00">I believe there is a stranger. </font>

7
00:00:32,312 --> 00:00:35,023
<font color="#ffff00">Comes to this village</font>
<font color="#ffff00">from the sea. </font>
```

สิ่งที่ต้องทำ

โปรแกรมของการบ้านนี้เป็น web application ให้บริการจัดการแฟ้ม **srt** ทางเว็บ เมื่อสั่งโปรแกรมทำงานด้วย Thonny ผู้ใช้สามารถเปิดเว็บบราวเซอร์แล้วไปที่ **http://127.0.0.1:5000/** จะแสดงหน้าจอดังรูปทางขวา มีให้ 3 บริการคือ

- **Shifter**: ให้บริการเลื่อนเวลาต่าง ๆ ในแฟ้ม .srt ถอยหลัง หรือไปข้างหน้าตามที่ระบุ เหมาะกับกรณีที่เรามีแฟ้ม srt ที่แสดงบทบรรยายไม่ตรงกับภาพที่แสดง ซึ่งอาจแสดงบทบรรยายเร็ว หรือช้าเกินไป
- **Cleaner**: ให้บริการลบบทบรรยายต่าง ๆ ที่อยู่ในเครื่องหมาย **( )**, **{ }**, **[ ]** และ **< >** (รวมทั้งเครื่องหมายนี้) ออกให้หมด และลบบทบรรยายที่ไม่มีตัวอักษรหรือตัวเลขใด ๆ ในบทออกด้วย
- **Merger**: ให้บริการผสานบทบรรยาย **srt** สองแฟ้มให้เป็นแฟ้มเดียว เช่น รวมบทบรรยายภาษาไทยหนึ่งแฟ้ม กับบทบรรยายภาษาอังกฤษอีกหนึ่งแฟ้ม เข้าด้วยกัน



จะสั่งทำงาน web application นี้ได้ ต้องมีแฟ้ม 3 แฟ้มดังนี้

- **webapp.py**: มีให้แล้ว (สั่ง run แฟ้มนี้ใน Thonny)
- **home.html**: มีให้แล้ว
- **hw8.py**: นิสิตต้องเขียน 3 ฟังก์ชัน (จะเขียนฟังก์ชันเสริมอื่นเพิ่มในแฟ้มนี้ก็ได้) (นิสิตส่งแฟ้ม **hw8.py** นี้เลย ไม่ต้องเพิ่มเลขประจำตัวนิสิตในชื่อแฟ้ม)

ส่งเฉพาะแฟ้มนี้เข้าระบบ CourseVille

นำทั้ง 3 แฟ้มนี้ใส่ใน folder เดียวกัน แล้วสั่งทำงานแฟ้ม **webapp.py** ด้วย Thonny

download ZIP ที่มีแฟ้มทั้งสาม และแฟ้ม srt ตัวอย่างได้ที่นี่

ฟังก์ชันที่ต้องเขียน

`def shift(file_in, time_shift, file_out):`

- **`file_in`** เป็นสตริงระบุชื่อแฟ้ม **`srt`** (แฟ้มมี **`encoding`** เป็น **`utf-8`**)
- **`time_shift`** เป็นจำนวนเต็มแทนปริมาณมิลลิวินาที (เป็นได้ทั้งค่าบวก ศูนย์และ ค่าลบ)
- **`file_out`** เป็นสตริงระบุชื่อแฟ้มผลลัพธ์ที่มีการปรับเวลาบทบรรยายของ **`file_in`** (แฟ้มมี **`encoding`** เป็น **`utf-8`**)
- ฟังก์ชันนี้ไม่คืนอะไร สิ่งที่ฟังก์ชันทำ คืออ่านข้อมูลบทบรรยายจากแฟ้ม **`file_in`** แล้วบวกค่าเวลาทั้งหลายไปอีก **`time_shift`**
  - ถ้า **`time_shift`** เป็นค่าบวก แสดงว่า ต้องการให้แสดงบทบรรยายช้าลงจากเดิม
  - ถ้า **`time_shift`** เป็นค่าลบ แสดงว่า ต้องการให้แสดงบทบรรยายเร็วขึ้นจากเดิม
    - ถ้าบวกเวลาด้วยค่าลบแล้ว ได้เวลาเป็นค่าลบ ให้แทนเวลาด้วยค่า **`00:00:00,000`**
    - ถ้าบวกเวลาด้วยค่าลบแล้ว ได้เวลาทั้งเริ่มและเลิก น้อยกว่าหรือเท่ากับศูนย์ทั้งคู่ ให้ลบบทนั้นทิ้ง (ดูตัวอย่างข้างล่าง)
- ตัวอย่าง: แฟ้ม **`test.srt`** มีข้อมูลดังแสดงข้างล่างนี้ทางซ้าย

  คำสั่ง **`shift('test.srt', 4500, 'test1_shifted.srt')`**

  สร้างแฟ้ม **`'test1_shifted.srt'`** ที่มีข้อมูลดังแสดงข้างล่างนี้ทางขวา

`test.srt`

```
1
00:00:04,000 --> 00:00:05,000
Hello, I'm Johnny Cash.

2
00:00:05,000 --> 00:00:08,000
[crowd cheering and applauding]

3
00:00:08,000 --> 00:00:11,000
[outlaw country music playing] ♪♫

4
00:00:11,000 --> 00:00:12,500
♪♪♪

5
00:00:13,000 --> 00:00:17,000
♪ I hear the train a comin'
It's rolling round the bend ♪
```

`test1_shifted.srt`

```
1
00:00:08,500 --> 00:00:09,500
Hello, I'm Johnny Cash.

2
00:00:09,500 --> 00:00:12,500
[crowd cheering and applauding]

3
00:00:12,500 --> 00:00:15,500
[outlaw country music playing] ♪♫

4
00:00:15,500 --> 00:00:17,000
♪♪♪

5
00:00:17,500 --> 00:00:21,500
♪ I hear the train a comin'
It's rolling round the bend ♪
```

คำสั่ง **`shift('test1.srt', -5500, 'test2_shifted.srt')`**

สร้างแฟ้ม **`'test2_shifted.srt'`** ที่มีข้อมูลดังแสดงข้างล่างนี้ทางขวา

`test.srt`

```
1
00:00:04,000 --> 00:00:05,000
Hello, I'm Johnny Cash.

2
00:00:05,000 --> 00:00:08,000
[crowd cheering and applauding]

3
00:00:08,000 --> 00:00:11,000
[outlaw country music playing] ♪♫

4
00:00:11,000 --> 00:00:12,500
♪♪♪

5
00:00:13,000 --> 00:00:17,000
♪ I hear the train a comin'
It's rolling round the bend ♪
```

`test2_shifted.srt`

```
1
00:00:00,000 --> 00:00:02,500
[crowd cheering and applauding]

2
00:00:02,500 --> 00:00:05,500
[outlaw country music playing] ♪♫

3
00:00:05,500 --> 00:00:07,000
♪♪♪

4
00:00:07,500 --> 00:00:11,500
♪ I hear the train a comin'
It's rolling round the bend ♪
```

`def merge(base_file, merge_file, threshold, file_out):`

- **`base_file`** เป็นสตริงระบุชื่อแฟ้ม **`srt`** (แฟ้มมี **`encoding`** เป็น **`utf-8`**)
- **`merge_file`** เป็นสตริงระบุชื่อแฟ้ม **`srt`** (แฟ้มมี **`encoding`** เป็น **`utf-8`**)
- **`threshold`** เป็นจำนวนเต็ม (ไม่ติดลบ)
- **`file_out`** เป็นสตริงระบุชื่อแฟ้มผลลัพธ์จากการผสาน **`base_file`** และ **`merge_file`** (แฟ้มมี **`encoding`** เป็น **`utf-8`**)
- ฟังก์ชันนี้ไม่คืนอะไร ทำหน้าที่ผสานบทบรรยายในแฟ้ม **`base_file`** และ **`merge_file`** เข้าด้วยกัน โดยมีหลักการผสานดังนี้
  - นำแต่ละบทบรรยายใน **`merge_file`** ไปรวมกับบทบรรยายใน **`base_file`** ที่มีเวลาเริ่มใกล้เคียงกันที่สุดและต้องใกล้กันไม่เกินค่า **`threshold`** หลังรวมแล้วให้เลือกใช้เวลาเริ่มและเลิกแสดงของ **`base_file`** ในแฟ้มผลลัพธ์

    เช่น จากตัวอย่างข้างล่าง บทบรรยายที่มีสีพื้นเหมือนกัน ถูกรวมเข้าด้วยกัน ในที่นี้กำหนดให้ **`threshold`** มีค่า **`1000`**
  - บทบรรยายที่ไม่มีสีพื้นครอบในตัวอย่าง คือบทที่ไม่สามารถหาบทบรรยายของอีกแฟ้มที่ใกล้เคียงกันที่ไม่เกินค่า **`threshold`** ได้

`test2_en.srt (base_file)`

```
...
5
00:09:08,608 --> 00:09:10,276
Bruce Wayne.

6
00:09:28,086 --> 00:09:29,629
Talk.

7
00:09:30,922 --> 00:09:32,549
I believe there is a stranger.

8
00:09:32,632 --> 00:09:35,343
Comes to this village
from the sea.

9
00:09:35,427 --> 00:09:38,513
He comes in the winter
when the people are hungry.

...
94
00:19:59,675 --> 00:20:01,969
St. Brigid's had
a school trip today.

95
00:20:25,910 --> 00:20:27,703
Quiet! Shut up!

96
00:20:32,792 --> 00:20:35,252
Down with the modern world.
```

`test2_th.srt (merge_file)`

```
...
5
00:09:08,135 --> 00:09:09,762
บรูซ เวย์น

6
00:09:11,263 --> 00:09:13,974
{\an8}พายุแรงจัด เฮลิคอปเตอร์เข้าไม่ได้หกวัน

7
00:09:14,433 --> 00:09:15,267
{\an8}แล้วเขามาจากไหน

8
00:09:15,684 --> 00:09:17,269
{\an8}เขาบอกว่าปีนเขาขึ้นมา

9
00:09:17,728 --> 00:09:18,771
{\an8}เป็นไปไม่ได้

10
00:09:27,780 --> 00:09:29,198
พูด

11
00:09:30,532 --> 00:09:35,037
ผมเชื่อว่ามีชายแปลกหน้า
จากทะเลมายังหมู่บ้านนี้

12
00:09:35,204 --> 00:09:37,956
ในฤดูหนาวเวลาที่ผู้คนหิวโหย

...
99
00:19:59,286 --> 00:20:01,497
เด็กโรงเรียนเซนต์บริจิดมาวันนี้

100
00:20:25,521 --> 00:20:26,355
เงียบ!

101
00:20:26,522 --> 00:20:27,398
หุบปาก!

102
00:20:32,486 --> 00:20:34,780
โลกสมัยใหม่ล่มสลาย

...
```

`test2_merged.srt`

```
...
7
00:09:08,608 --> 00:09:10,276
Bruce Wayne.
บรูซ เวย์น

8
00:09:11,263 --> 00:09:13,974
{\an8}พายุแรงจัด เฮลิคอปเตอร์เข้าไม่ได้หกวัน

9
00:09:14,433 --> 00:09:15,267
{\an8}แล้วเขามาจากไหน

10
00:09:15,684 --> 00:09:17,269
{\an8}เขาบอกว่าปีนเขาขึ้นมา

11
00:09:17,728 --> 00:09:18,771
{\an8}เป็นไปไม่ได้

12
00:09:28,086 --> 00:09:29,629
Talk.
พูด

13
00:09:30,922 --> 00:09:32,549
I believe there is a stranger.
ผมเชื่อว่ามีชายแปลกหน้า
จากทะเลมายังหมู่บ้านนี้

14
00:09:32,632 --> 00:09:35,343
Comes to this village
from the sea.

15
00:09:35,427 --> 00:09:38,513
He comes in the winter
when the people are hungry.
ในฤดูหนาวเวลาที่ผู้คนหิวโหย

...
106
00:19:59,675 --> 00:20:01,969
St. Brigid's had
a school trip today.
เด็กโรงเรียนเซนต์บริจิดมาวันนี้

107
00:20:25,910 --> 00:20:27,703
Quiet! Shut up!
เงียบ!
หุบปาก!

108
00:20:32,792 --> 00:20:35,252
Down with the modern world.
โลกสมัยใหม่ล่มสลาย

...
```

`def clean(file_in, file_out):`

- **`file_in`** เป็นสตริงระบุชื่อแฟ้ม **`srt`** (มี **`encoding`** เป็น **`utf-8`**)
- **`file_out`** เป็นสตริงระบุชื่อแฟ้มผลลัพธ์ (แฟ้มมี **`encoding`** เป็น **`utf-8`**)
- ฟังก์ชันนี้ไม่คืนอะไร สิ่งที่ฟังก์ชันทำ คืออ่านข้อมูลบทบรรยายจากแฟ้ม **`file_in`** เพื่อสร้างแฟ้ม **`file_out`** ตามขั้นตอนข้างล่างนี้

1. ลบข้อความใน **`file_in`** ที่อยู่ระหว่างเครื่องหมาย **`(`** กับ **`)`**, **`[`** กับ **`]`**, **`{`** กับ **`}`** และ **`<`** กับ **`>`** (รวมทั้งเครื่องหมาย **`( )`**, **`[ ]`**, **`{ }`** และ **`< >`**) ออกให้หมด คู่เครื่องหมายเหล่านี้ไม่จำเป็นต้องอยู่ในบรรทัดเดียวกัน และไม่มีกรณีที่เครื่องหมายเหล่านี้ปรากฏซ้อน ๆ กัน เช่น **`<[b<b>]>`**
2. หลังทำขั้นตอนที่ **`1`** ให้ลบบรรทัดของบทที่ข้อความบรรยายไม่มีตัวอักษรและตัวเลขเลย (อาจมีเครื่องหมายอื่นอยู่) ออก การทดสอบว่า ตัวอักขระในตัวแปร **`c`** เป็นตัวอักษรหรือตัวเลขหรือไม่ ให้ใช้ **`c.isalnum()`** ซึ่งคืน **`True`** หรือ **`False`** (**ห้ามทดสอบด้วยวิธีอื่น**)
3. หลังทำขั้นตอนที่ **`2`** หากบทบรรยายไม่มีคำบรรยายเหลืออยู่เลย ให้ลบบทนั้นออก
4. บันทึกบทบรรยายที่เหลือไปยังแฟ้ม **`file_out`**

คำสั่ง **`clean('test.srt', 'test_cleaned.srt')`**
สร้างแฟ้ม **`'test_cleaned.srt'`** ที่มีข้อมูลดังแสดงข้างล่างนี้ทางขวา

`test.srt`

```
1
00:00:04,000 --> 00:00:05,000
Hello, I'm Johnny Cash.

2
00:00:05,000 --> 00:00:08,000
[crowd cheering and applauding]

3
00:00:08,000 --> 00:00:11,000
[outlaw country music playing] ♪♫

4
00:00:11,000 --> 00:00:12,500
♪♪♪

5
00:00:13,000 --> 00:00:17,000
♪ I hear the train a comin'
It's rolling round the bend ♪
```

`test_cleaned.srt`

```
1
00:00:04,000 --> 00:00:05,000
Hello, I'm Johnny Cash.

2
00:00:13,000 --> 00:00:17,000
♪ I hear the train a comin'
It's rolling round the bend ♪
```

ข้อแนะนำ

- ในแฟ้ม `zip` (ที่มีให้ `download` ในหน้าที่ 2) มีแฟ้ม `srt` หลายแฟ้ม ที่สามารถใช้ทดสอบการทำงานของฟังก์ชันได้ดังนี้
  - `test1_en.srt`, `test2_en.srt`, `test3_en.srt` และ `test4_en.srt` เป็นแฟ้มบทบรรยายภาษาอังกฤษ
  - `test2_th.srt`, `test3_th.srt` และ `test4_th.srt` เป็นแฟ้มบทบรรยายภาษาไทย
  - `test2_en_cleaned.srt`, `test2_th_cleaned.srt`, `test3_en_cleaned.srt` และ `test3_th_cleaned.srt` เป็นแฟ้มที่ได้จากการ `clean` แฟ้ม `test2_en.srt`, `test2_th.srt`, `test3_en.srt` และ `test3_th.srt` ตามลำดับ
  - `test2_en_th_merged.srt` เป็นแฟ้มบทบรรยายภาษาอังกฤษและไทย ที่ได้มาจากใช้คำสั่ง
    `merge('test2_en.srt', 'test2_th.srt', 1000, 'test2_en_th_merged.srt')`
  - `test2_th_en_merged.srt` เป็นแฟ้มบทบรรยายภาษาอังกฤษและไทย ที่ได้มาจากใช้คำสั่ง
    `merge('test2_th.srt', 'test2_en.srt', 1000, 'test2_th_en_merged.srt')`
  - `test3_en_th_merged.srt` เป็นแฟ้มบทบรรยายภาษาอังกฤษและไทย ที่ได้มาจากใช้คำสั่ง
    `merge('test3_en.srt', 'test3_th.srt', 1000, 'test3_en_th_merged.srt')`
  - `test3_th_en_merged.srt` เป็นแฟ้มบทบรรยายภาษาอังกฤษและไทย ที่ได้มาจากใช้คำสั่ง
    `merge('test3_th.srt', 'test3_en.srt', 1000, 'test3_th_en_merged.srt')`
  - `test4_en_th_merged.srt` เป็นแฟ้มที่ได้จากคำสั่ง
    `merge('test4_en.srt', 'test4_th.srt', 1000, 'test4_en_th_merged.srt')`

- แฟ้ม `srt` บางแฟ้ม อาจมีค่าของเลขลำดับบท ไม่เป็นไปตามกฎ คือ ไม่ได้เรียง `1,2,3, ...` ดังนั้น ชุดคำสั่งที่อ่านเลขลำดับไม่ต้องสนใจค่าของเลขลำดับในแฟ้มที่อ่าน แต่แฟ้มผลลัพธ์ที่สร้างจากฟังก์ชันที่ให้เขียนในการบ้านนี้ ต้องมีเลขลำดับที่เรียง `1,2,3,...` ตามข้อกำหนด
- เนื่องจากแฟ้ม `srt` ถูกบันทึกไว้โดยเข้ารหัสในรูปแบบ `utf-8` ดังนั้น คำสั่งที่เปิดแฟ้มเพื่ออ่าน หรือเพื่อบันทึก จะต้องมี `encoding='utf-8'` อยู่ในคำสั่ง `open` ด้วย เช่น

```
fin = open(file_in , encoding='utf-8')
```

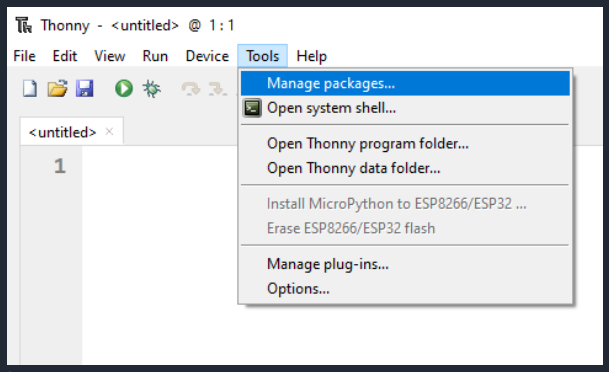
หรือ

```
fout = open(file_out, 'w', encoding='utf-8')
```

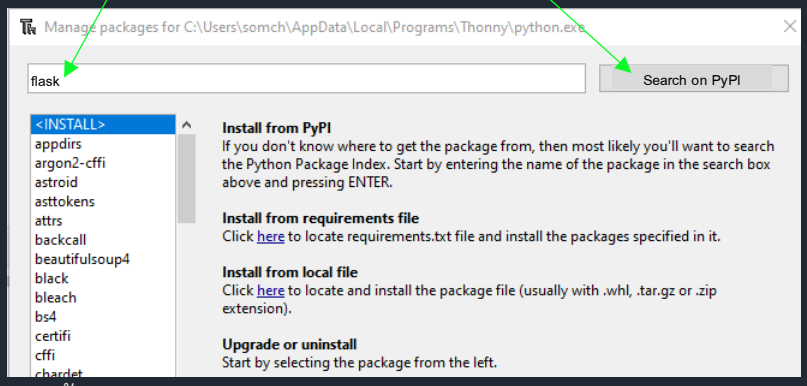
วิธีติดตั้ง flask ใน Thonny

โปรแกรมในการบ้านนี้ หากต้องการ `run webapp.py` ต้องติดตั้ง `Flask` ใน `Thonny` ก่อน ดังนี้

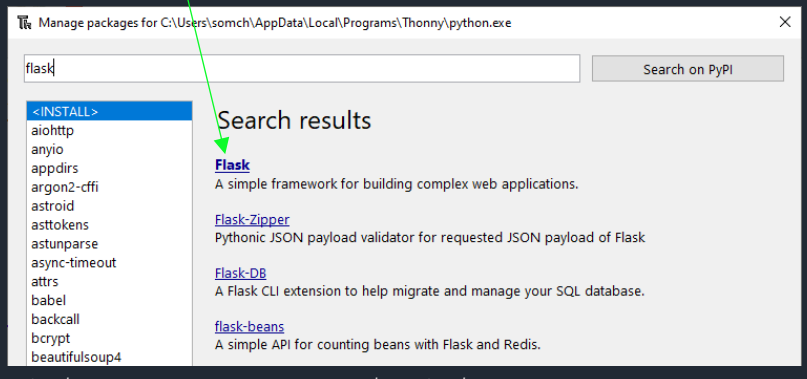
- ใน `Thonny` เลือกเมนู `Tools -> Manage packages...`



- ใส่คำว่า **`flask`** และกดปุ่ม **`Search on PyPI`**



- จากนั้นคลิกเลือก



- แล้วก็กดปุ่ม `Install` รอจนเสร็จ แล้วก็กดปุ่ม `Close`