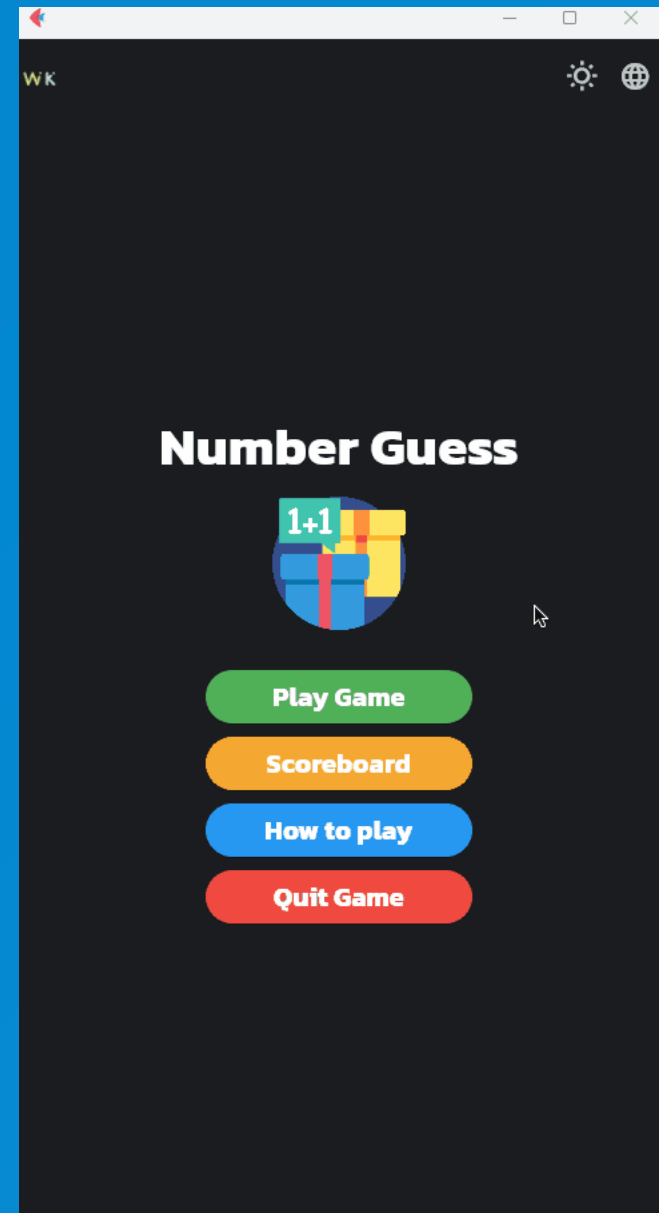


เกมทายเลข

Number Guess Game

เกมทายเลข (Number Guess Game) วัชร บุตร์ดีวงษ์



ติดตั้งโมดูลต่างๆ

```
pip install -r requirements.txt
```

หรือกรณีมี virtualenv

```
virtualenv game
```

บน Windows เรียกใช้ virtualenv game

```
game\Scripts\activate  
pip install -r requirements.txt
```

เรียกใช้งานเกม

```
flet num*.py
```

สร้างไฟล์ EXE

```
pyinstaller number_guessing_game_by_wk18k.spec
```

เปิดเกมที่ Build เรียบร้อยแล้ว

1. เปิดที่โฟลเดอร์ `Number guessing game`
2. เปิดตัวเกมชื่อว่า `Number Guessing Game By wk18k.exe`

วิธีเล่นเกมทายเลข

การเล่นเกมนี้นิยํากมาก แค่ทาย
ตัวเลขให้ถูก จะได้รับคะแนนเก็บ
สะสมไปเรื่อยๆได้ โดยจะมีเวลาเป็น
กำหนด หากเวลาหมดจะเป็นการ
จบเกมโดยปริยาย โดยตัวเลขจะ
สุ่มทุกครั้งทีผู้เล่นเริ่มเกม หรือเริ่ม
เกมใหม่ ทุกการกด เริ่มใหม่จะ
ทำการสุ่มตัวใหม่และลดคะแนน
ครั้งของคะแนนที่จะได้รับ

เกมทายเลข



เล่นเกม

ตารางคะแนน

ช่วยเหลือ

ออกเกม

ระดับความยาก

มีอยู่ 3 โหมด คือ

1. ง่าย
2. ปานกลาง
3. ยาก

เกมทายเลข



เล่นเกม

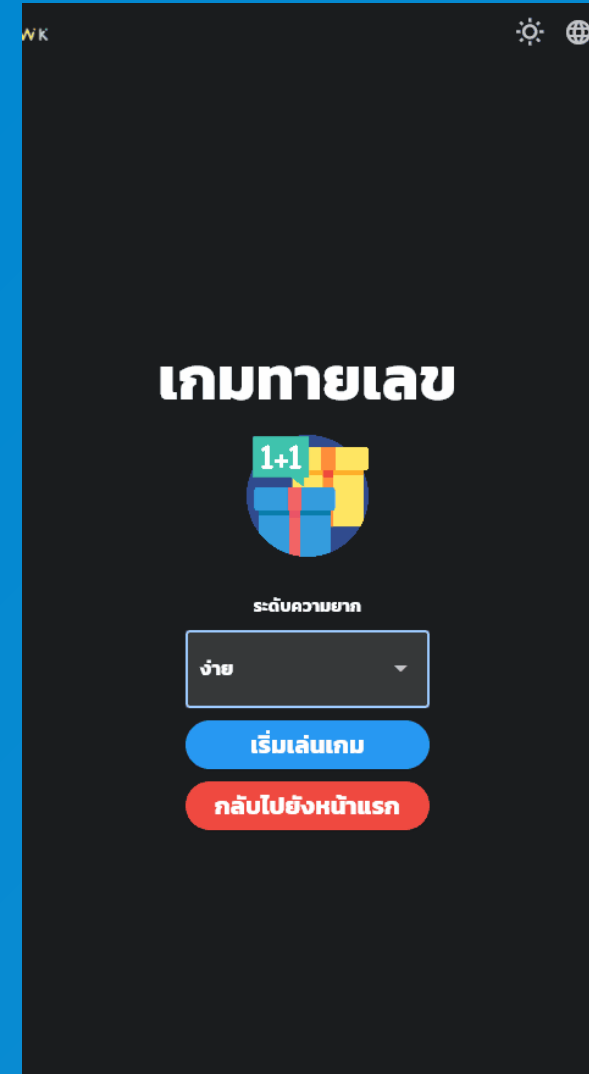
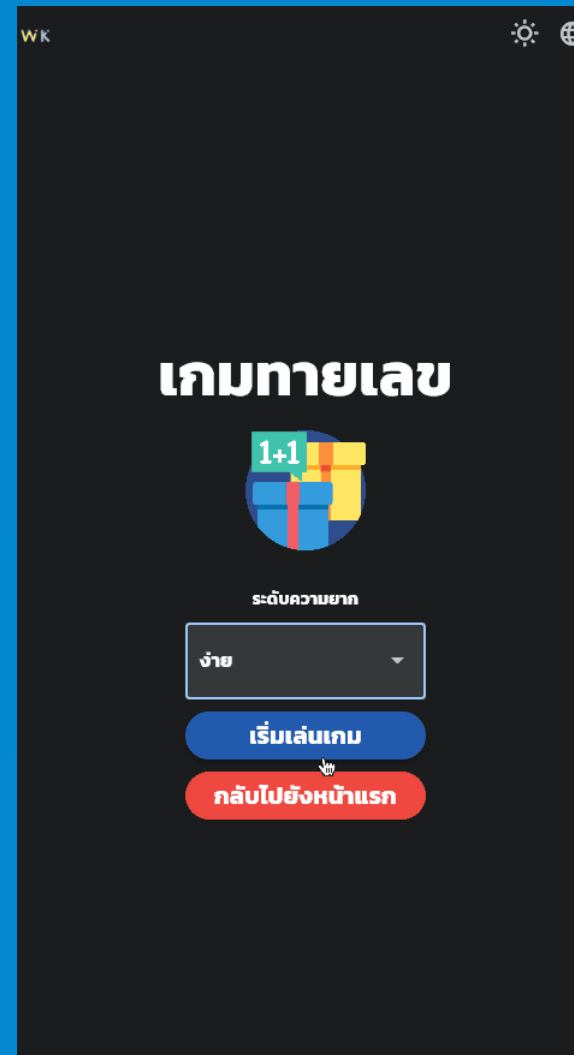
ตารางคะแนน

ช่วยเหลือ

ออกเกม

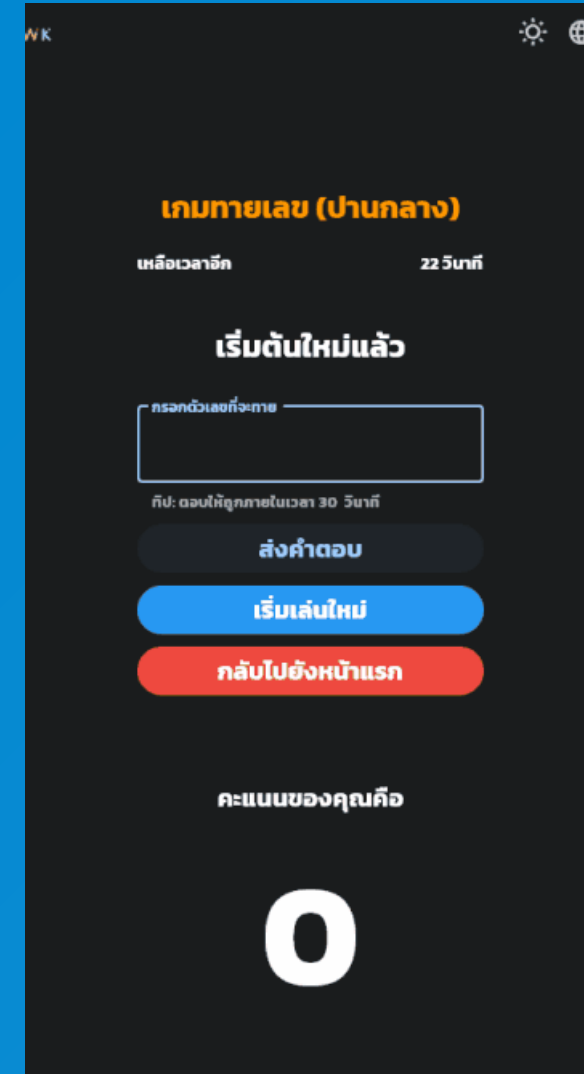
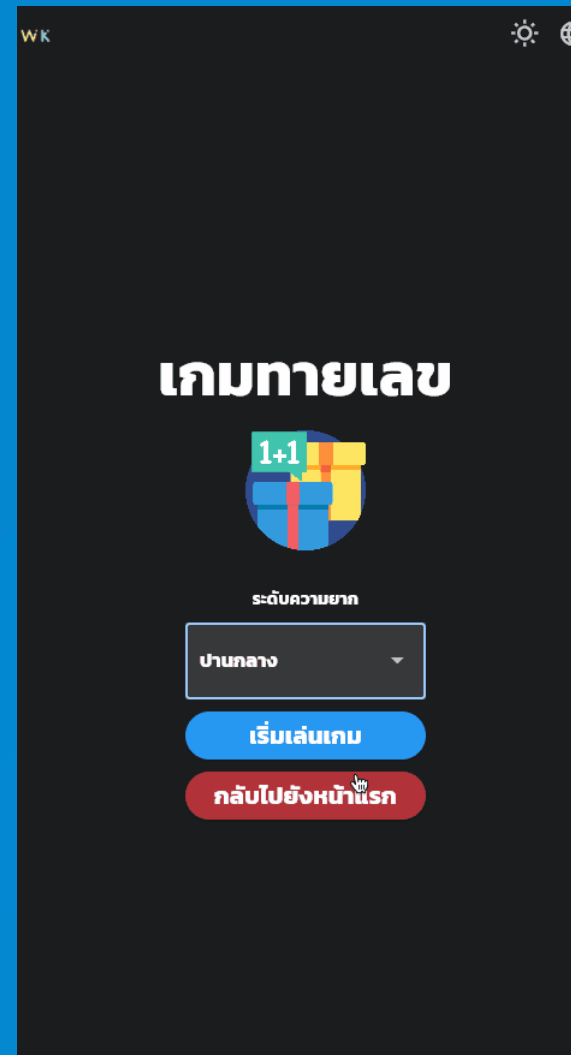
ระดับง่าย

โหมดนี้จะมีเวลาให้ผู้เล่นทั้งหมด 30 วินาที โดยผู้เล่นจะต้องตอบให้ถูกก่อนที่เวลาจะหมด หากตอบถูกจะได้คะแนนข้อละ 1 คะแนน สุ่มตัวเลขตั้งแต่ 1 – 10 โดยหากผู้ใช้กดปุ่ม **เริ่มเกมใหม่** หากคะแนนไม่เท่ากับ 0 คะแนนจะถูกลบออกไป ครั้งละ **1 / 2** หรือ **0.5**



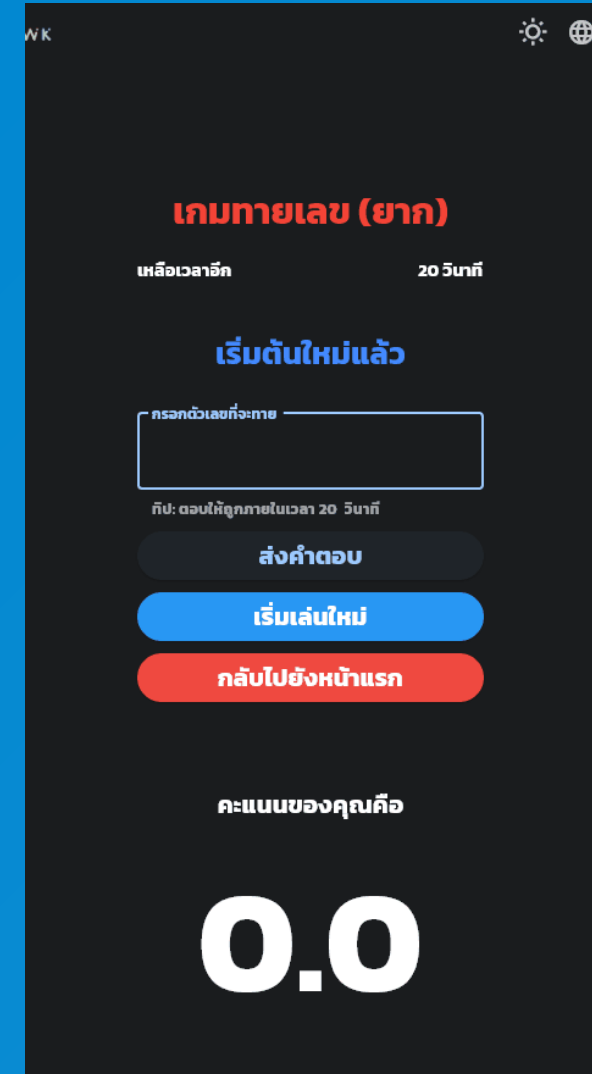
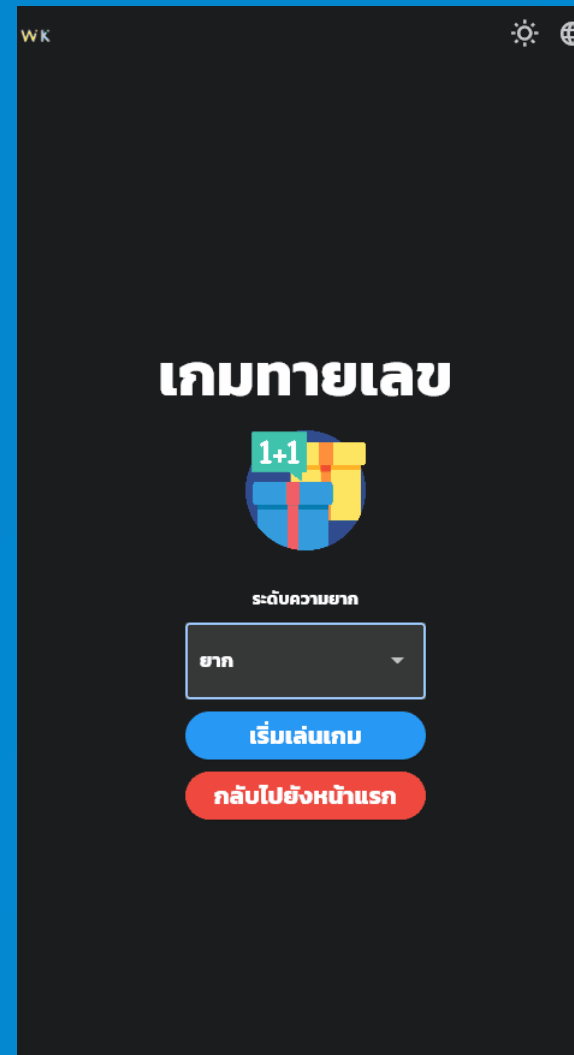
ระดับปานกลาง

โหมดนี้จะมีเวลาให้ผู้เล่นทั้งหมด 30 วินาที โดยผู้เล่นจะต้องตอบให้ถูกต้องก่อนที่เวลาจะหมด หากตอบถูกต้องจะได้คะแนนข้อละ 5 คะแนน สุ่มตัวเลขตั้งแต่ 1 – 100 โดยหากผู้ใช้กดปุ่ม **เริ่มเกมใหม่** หากคะแนนไม่เท่ากับ 0 คะแนนจะถูกลบออกไปครั้งละ **5 / 2** หรือ **2.5**

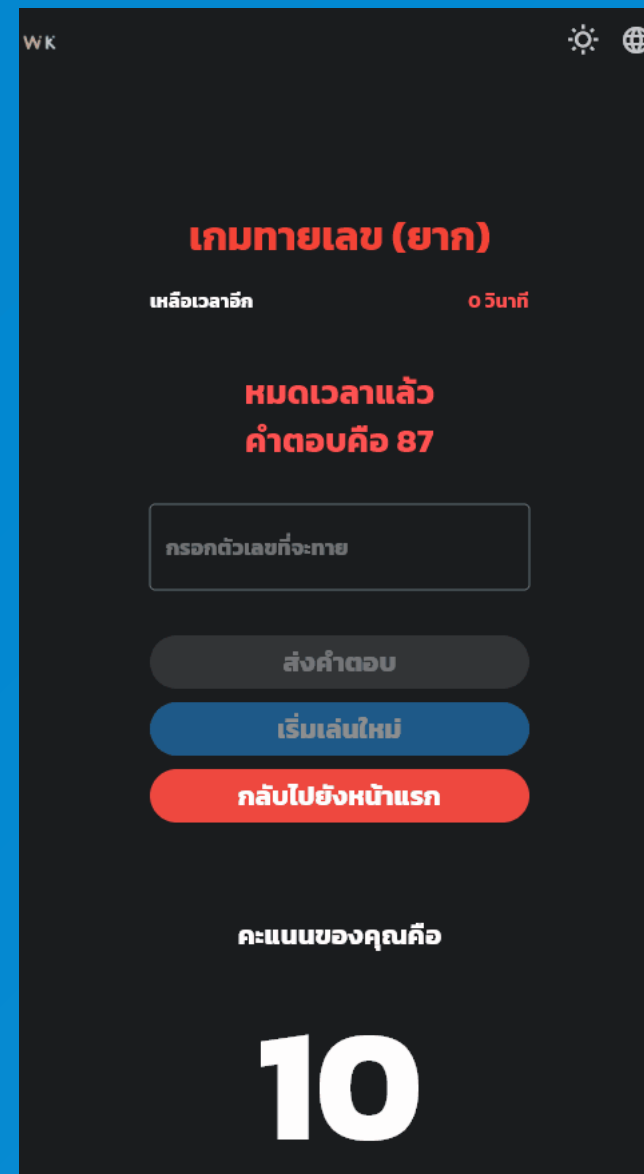


ระดับยาก

โหมดนี้จะมีเวลาให้ผู้เล่นทั้งหมด 20 วินาที โดยผู้เล่นจะต้องตอบให้ถูกก่อนที่เวลาจะหมด หากตอบถูกจะได้คะแนนข้อละ 10 คะแนน สุ่มตัวเลขตั้งแต่ 1 - 100 โดยหากผู้ใช้กดปุ่ม **เริ่มเกมใหม่** หากคะแนนไม่เท่ากับ 0 คะแนนจะถูกลบออกไปครั้งละ **10 / 2** หรือ **5**



หลังจากเวลาหมดลงผู้เล่นจะ
บังคับให้กดปุ่ม **กลับไปหน้าหลัก** แต่
หลังจากกดปุ่มตัวเกมจะให้ผู้เล่น
กรอกชื่อของผู้เล่นเพื่อบันทึก
คะแนนลงในตารางคะแนนแล้วกด
ปุ่ม **ส่งชื่อ** หากผู้เล่นไม่ต้องการ
สามารถกดปุ่ม **กลับไปหน้าหลัก** ระบบ
ก็จะส่งผู้เล่นไปหน้าแรกของเกม



ส่วนต่างๆของ หน้าแรก



ปุ่มเปลี่ยนธีม

ใช้ในการสลับธีมสว่างกับ ธีมมืด

```
def theme_changed(e):  
    """  
    เปลี่ยนธีม สว่าง กับ มืด  
    """  
    page.theme_mode = (  
        ft.ThemeMode.DARK  
        if page.theme_mode == ft.ThemeMode.LIGHT  
        else ft.ThemeMode.LIGHT  
    )  
    page.update()
```



ปุ่มตารางคะแนน

จะส่งไปยังหน้าแสดงคะแนนของผู้
เล่นทั้งหมด

```
def go_to_scoreboard_Scene(e):  
    """  
    change scene to scoreboard scene  
    """  
    page.remove(menu_game_all)  
    page.add(scoreboard_scene_all)  
    query_database()  
    scoreboard_scene.list_player_rank.content = ft.Column(  
        scoreboard_scene.resutl_data_list_player_rank()  
    )  
    scoreboard_scene.rank_1_name.value = query_database()[0][0]  
    scoreboard_scene.rank_1_score.value = query_database()[0][1]  
    page.update()
```



ปุ่มวิธีการเล่น

จะส่งไปยังหน้าบอกวิธีการเล่นและ
วิธีการคิดคะแนนในแต่ละระดับ
ความยาก

```
def go_to_help_Scene(e):  
    """  
    change scene to help scene  
    """  
    page.scroll = True  
    page.add(help_scene_all)  
    page.remove(menu_game_all)
```



ปุ่มออก

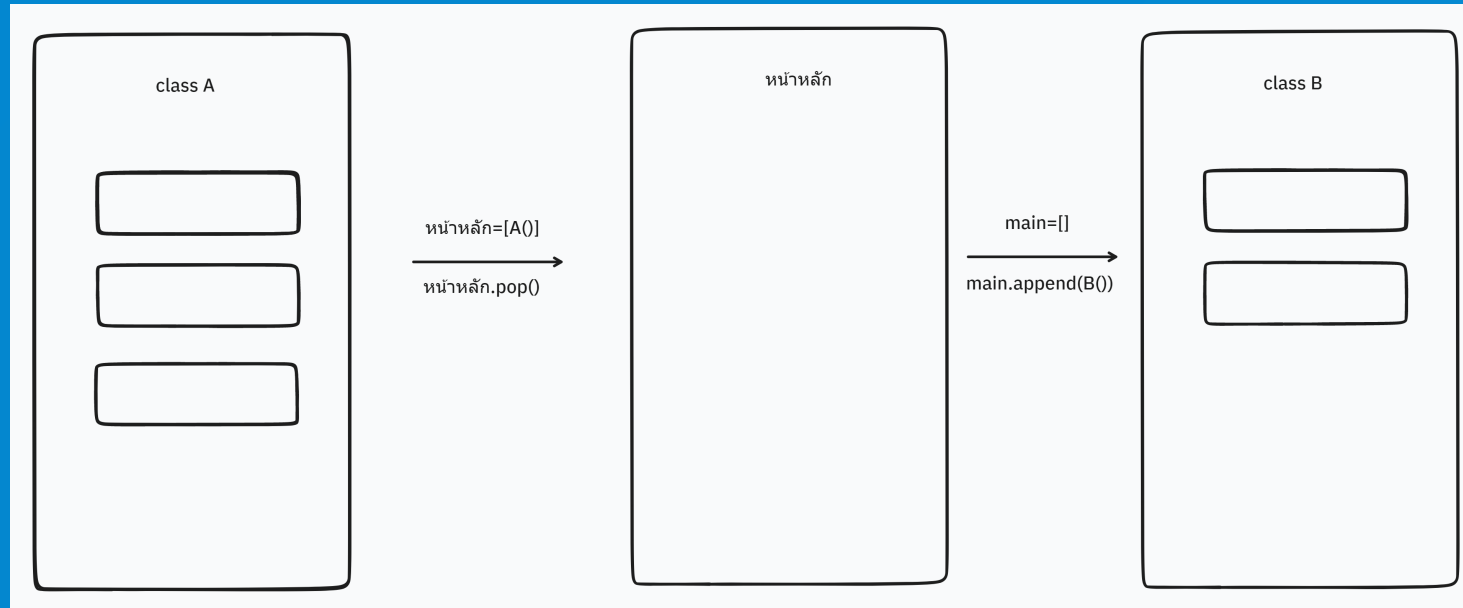
สำหรับผู้เล่นที่ต้องการออกจาก
กดปุ่มนี้ได้เลย

```
def exit_game(e):  
    """  
    Close App  
    """  
    page.window_destroy()
```



อุปสรรคและความท้าทาย

คือการเปลี่ยนหน้าไปยังอีกหน้านึง ซึ่งเป็นความท้าทายมากเนื่องจากโมดูล flet ในการเพิ่มแต่ละ widget จะทำการเก็บเป็น list โดยหากจะเปลี่ยนต้องทำการลบข้อมูลใน list ผู้พัฒนาเลยเลือกวิธีการทำให้ทุกหน้าแยกเป็น class เวลาลบมันออกจะได้ลบทุกอย่างที่อยู่ใน class นั้นๆทิ้งไป



อุปสรรคและความท้าทาย

ระบบการเปลี่ยนภาษาก็เป็นความท้าทายหนึ่งอย่าง ครั้งแรกผู้พัฒนาจะทำให้สามารถเปลี่ยนได้แบบเรียลไทม์แต่พอมาคำนวณระยะเวลาแล้วจึงตัดสินใจว่าหลังจากเปลี่ยนภาษาแล้วจะทำการรีสตาร์ทเกมเพื่อโหลดค่าตัวแปรที่เก็บไว้ในไฟล์ `config.json` แทน เพื่อความรวดเร็วในการเขียนโค้ด โดยไฟล์ `config.json` จะถูกสร้างขึ้นเองเมื่อเปิดเกมหลังจากผู้ใช้เลือกภาษาได้แล้วจะเก็บข้อมูลไปยังไฟล์นี้เพื่อบันทึกค่า จากนั้นจะทำการรีสตาร์ทเพื่อโหลดข้อมูลในไฟล์ออกมาใช้ถือว่าเป็นภาษาอะไร รูปแบบภายในไฟล์ `config.json` คือ

```
{  
  "language": "thai"  
}
```

อุปสรรคและความท้าทาย

ระบบ Scoreboard ปัญหาที่เจอคือระบบไม่อัปเดตข้อมูลในหน้า **ตารางคะแนน** หลังจากผู้เล่นบันทึกข้อมูลเสร็จเรียบร้อยแล้ว แต่เปิดเกมปิดใหม่ข้อมูลมาปกติ ทางผู้พัฒนาเลยทำการ query ข้อมูลหลังจากผู้ใช้งานกดปุ่ม **ตารางคะแนน** เพื่อโหลดข้อมูลจากฐานข้อมูลออกมาแล้วแสดงข้อมูล หลังจากนั้นทำการใช้คำสั่ง `page.update()` เพื่ออัปเดต widget ต่างๆของ flet ให้แสดงข้อมูลออกมา

```
scoreboard_scene.list_player_rank.content = ft.Column(scoreboard_scene.resutl_data_list_player_rank())
scoreboard_scene.rank_1_name.value = query_database()[0][0] #เอาข้อมูลชื่อออกมาฐานข้อมูล
scoreboard_scene.rank_1_score.value = query_database()[0][1] #เอาข้อมูลคะแนนออกมาฐานข้อมูล
score_add_scene.input_name_player.value = "" #ล้างข้อมูลที่อยู่ของป้อนชื่อผู้เล่นออก
score_add_scene.text_show.value = "" #ล้างข้อมูลคะแนนที่ได้ออกไป
page.update() #อัปเดต widget
```