

Kivy #4 Begin Game

1. ติดตั้ง Kivy ให้เรียบร้อย

```
pip install docutils pygments kivy_deps.sdl2 kivy_deps.glew kivy_deps.angle \
kivy
```

** ถ้าไม่สามารถติดตั้งได้ ทดลองใช้ `python -m pip install` หรือ `py -m pip install` แทน

2. ศึกษาการทำงานของ Pong Game จาก

<https://kivy.org/doc/stable/tutorials/pong.html>

Checkpoint #1

2.1 เปลี่ยนสีลูกปิงปองและเปลี่ยนสีไม้ปิงปองให้มีสีต่างกันได้อย่างไร

2.2 เปลี่ยน ลูกปิงปองให้เป็นรูปภาพ เช่น รูปโปเกมอน ได้อย่างไร

2.3 การทำให้วัตถุทุกชนิดที่อยู่ในเกมส์ สามารถเคลื่อนที่ได้ ต้องใช้โมดูลอะไร และ มีการทำงานอย่างไร

2.4 ทดลองเปลี่ยน `Clock.schedule_interval(game.update, 1.0 / 60.0)` เป็น `1/40.0` และ `1/20.0` เกิดอะไรขึ้น

3. สร้างเกมสโตร์โดยใช้คีย์บอร์ด

3.1 เขียนโค้ดดังต่อไปนี้ แล้วสังเกตผล

```
from kivy.app import App
from kivy.uix.widget import Widget

class MyApp(App):
    def build(self):
        return Widget()

if __name__ == '__main__':
    app = MyApp()
    app.run()
```

3.2 เพิ่ม widget ให้กับโปรแกรมดังโค้ดด้านล่าง ทดลองรัน

```
from kivy.app import App
from kivy.uix.widget import Widget
from kivy.graphics import Rectangle

class GameWidget(Widget):
    def __init__(self, **kwargs):
        super().__init__(**kwargs)

        with self.canvas:
            Rectangle(pos=(0, 0), size=(100, 100))

class MyApp(App):
    def build(self):
        return GameWidget()

if __name__ == '__main__':
    app = MyApp()
    app.run()
```

Checkpoint #2

ทดลองเปลี่ยน พารามิเตอร์ pos และ size เป็นค่าอื่นแล้วสังเกตผล

ทำให้ Rectangle เปลี่ยนเป็นรูปภาพอะไรก็ได้ที่ต้องการ

3.3 ทำให้โปรแกรมสามารถรับค่าจากคีย์บอร์ดได้

```
from kivy.app import App
from kivy.uix.widget import Widget
from kivy.graphics import Rectangle
from kivy.core.window import Window

class GameWidget(Widget):
    def __init__(self, **kwargs):
        super().__init__(**kwargs)

        self._keyboard = Window.request_keyboard(
            self._on_keyboard_closed, self)
        self._keyboard.bind(on_key_down=self._on_key_down)

        with self.canvas:
            Rectangle(source='hero.png', pos=(0, 0), size=(100, 100))

    def _on_keyboard_closed(self):
        self._keyboard.unbind(on_key_down=self._on_key_down)
        self._keyboard = None

    def _on_key_down(self, keyboard, keycode, text, modifiers):
        print('key down', text)

class MyApp(App):
    def build(self):
        return GameWidget()

if __name__ == '__main__':
    app = MyApp()
    app.run()
```

3.4 ปังคับตัวละครด้วยคีย์บอร์ด

```
from kivy.app import App
from kivy.uix.widget import Widget
from kivy.graphics import Rectangle
from kivy.core.window import Window

class GameWidget(Widget):
    def __init__(self, **kwargs):
        super().__init__(**kwargs)

        self._keyboard = Window.request_keyboard(
            self._on_keyboard_closed, self)
        self._keyboard.bind(on_key_down=self._on_key_down)

        with self.canvas:
            self.hero = Rectangle(source='hero.png', pos=(0, 0), size=(100,
100))

    def _on_keyboard_closed(self):
        self._keyboard.unbind(on_key_down=self._on_key_down)
        self._keyboard = None

    def _on_key_down(self, keyboard, keycode, text, modifiers):
        print('key down', text)

        cur_x = self.hero.pos[0]
        cur_y = self.hero.pos[1]

        if text == 'w':
            cur_y += 1
        elif text == 'd':
            cur_x += 1

        self.hero.pos = (cur_x, cur_y)

class MyApp(App):
    def build(self):
        return GameWidget()

if __name__ == '__main__':
    app = MyApp()
    app.run()
```

Checkpoint #3

เพิ่มโค้ดให้สามารถบังคับตัวละครให้ไปด้านซ้าย และลงด้านล่าง

3.4 บังคับตัวละครด้วยคีย์บอร์ดให้เลื่อนไหล

```
from kivy.app import App
from kivy.uix.widget import Widget
from kivy.graphics import Rectangle
from kivy.core.window import Window
from kivy.clock import Clock

class GameWidget(Widget):
    def __init__(self, **kwargs):
        super().__init__(**kwargs)

        self._keyboard = Window.request_keyboard(
            self._on_keyboard_closed, self)
        self._keyboard.bind(on_key_down=self._on_key_down)
        self._keyboard.bind(on_key_up=self._on_key_up)

        self.pressed_keys = set()
        Clock.schedule_interval(self.move_step, 0)

        with self.canvas:
            self.hero = Rectangle(source='hero.png', pos=(0, 0), size=(100,
100))

    def _on_keyboard_closed(self):
        self._keyboard.unbind(on_key_down=self._on_key_down)
        self._keyboard.unbind(on_key_up=self._on_key_up)
        self._keyboard = None

    def _on_key_down(self, keyboard, keycode, text, modifiers):
        print('down', text)
        self.pressed_keys.add(text)

    def _on_key_up(self, keyboard, keycode):
        text = keycode[1]
        print('up', text)

        if text in self.pressed_keys:
```

```

        self.pressed_keys.remove(text)

    def move_step(self, dt):
        cur_x = self.hero.pos[0]
        cur_y = self.hero.pos[1]

        step = 100 * dt

        if 'w' in self.pressed_keys:
            cur_y += step
        if 's' in self.pressed_keys:
            cur_y -= step
        if 'a' in self.pressed_keys:
            cur_x -= step
        if 'd' in self.pressed_keys:
            cur_x += step

        self.hero.pos = (cur_x, cur_y)

class MyApp(App):
    def build(self):
        return GameWidget()

if __name__ == '__main__':
    app = MyApp()
    app.run()

```

Checkpoint #4

ถ้าต้องการเพิ่มความเร็วในการเคลื่อนที่ของ hero ต้องทำอย่างไร

โค้ดข้อ 3.3 กับ ข้อ 3.4 การใช้งานคีย์บอร์ดต่างกันอย่างไร
