

# 機械設計工程系

## Bg6-1 期末報告

### 專題製作與課程學習成果

#### 組員：

40723219 林承治

40723223 洪緯辰

40723225 高永霖

40723232 陳鈺凱

40723239 黃學聖

40723249 顏士勛

# 摘要

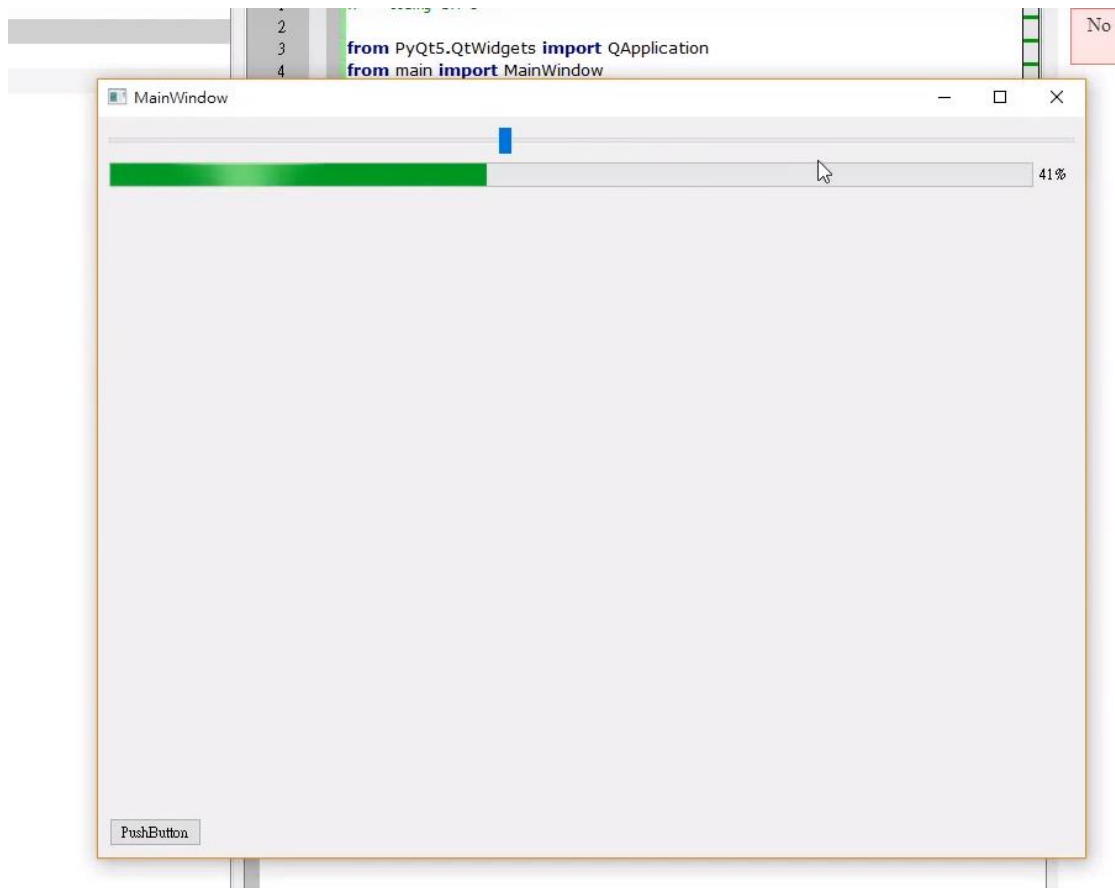
期末老師要我們訂出一個專題並給我們範  
例去參考做出，因此我們做出了一個由  
`python` 語法所做出的小遊戲

# 目錄

摘要.....	
課程內容.....	1
文一.....	2
文二.....	3
文三.....	4
專題內容.....	5
文一.....	6
文二.....	7
文三.....	8
文三.....	9

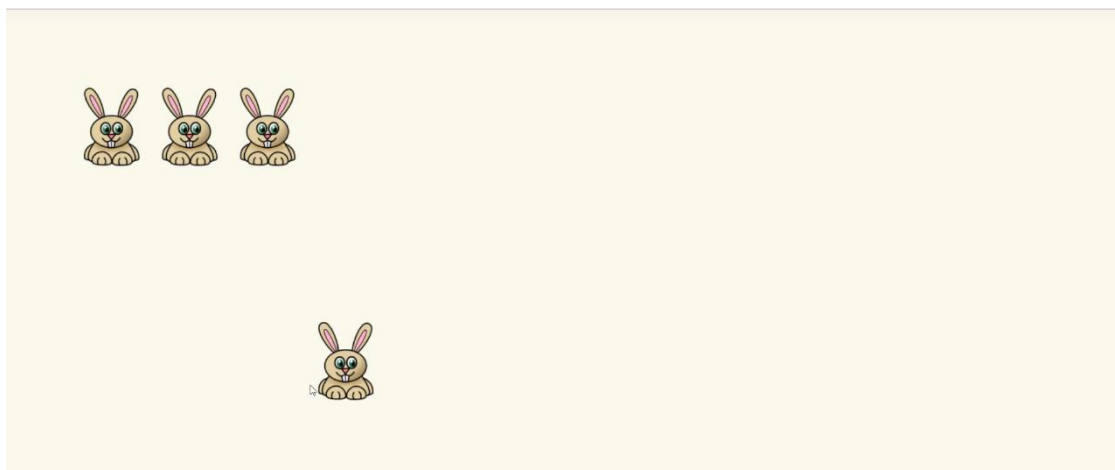
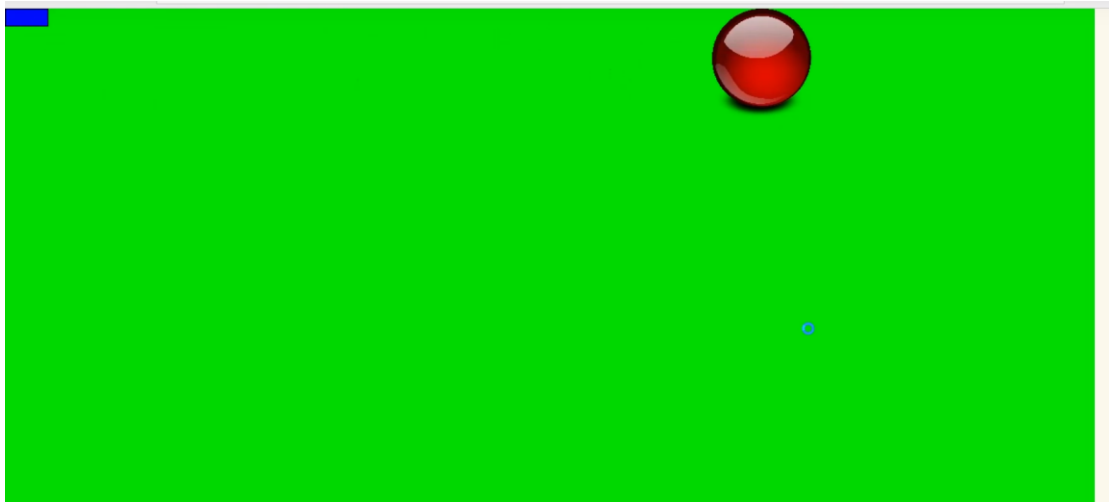
## 課程內容

助教利用新的 p37 裡的 eric6 教我們運用裡面的模組來創建一個頁面，學長教了我們如何利用拉桿及彈簧的運用來製造出進度條的連動





老師介紹了一個由 brython 寫的小遊戲，教我們如何在 leoeditor 裡面做一些簡單的變動，讓程式改變，我們目前只有學一些簡單 python 語法，如果要大改變，勢必要花很多時間。



## 專題

### 利用老師給的範例中而加以更改

```
@language python
# 導入模組
from ggame import App, ImageAsset, Sprite, MouseEvent
from ggame import Color, Sound, LineStyle, RectangleAsset, CircleAsset, PolygonAsset, SoundAsset
from random import random, randint

up = 0
down = 0
right = 0
life = 0
go1 = 1
back1 = 0
go2 = 1
back2 = 0
```

### 將上下左右定義出來

```
def w(event):
    global up
    global down
    global right
    global life
    up = 1
    down = 0
    right = 0
    life = 0
    .....
def s(event):
    global up
    global down
    global right
    global life
    up = 0
    down = 1
    right = 0
    life = 0
    .....
def d(event):
    global up
    global down
    global right
    global life
    up = 0
    down = 0
    right = 1
    life = 0
    .....
def c(event):
    global up
    global down
    global right
```

## 放置物件圖片，圖片大小

```
.....  
class Bunny(Sprite):  
  
    asset = ImageAsset("images/1.png")  
  
    def __init__(self, position):  
        super().__init__(Bunny.asset, position)  
        # register mouse events  
        App.listenKeyEvent('keydown', 'w', w)  
        App.listenKeyEvent('keydown', 's', s)  
        App.listenKeyEvent('keydown', 'd', d)  
        App.listenKeyEvent('keydown', 'a', c)  
        self.scale = 0.4
```

## 為上下左右移動的 x, y 方向以及距離

```
.....  
def step(self):  
    global up  
    global down  
    global right  
    global life  
    if up and self.y > 0:  
        self.y -= 50  
        up -= 1  
    if down and self.y < 770:  
        self.y += 50  
        down -= 1  
    if right and self.x < 1710:  
        self.x += 50  
        right -= 1  
    if life and self.x > 0:  
        self.x -= 50  
        life -= 1
```



## 背景圖片及球的圖片

```
class Bg(Sprite):
    asset = ImageAsset("images/bg.png")
    .....
    def __init__(self, position):
        super().__init__(Bg.asset, position)
    .....
    def step(self):
        self.x = 0
        self.y = 0
    .....
class ball(Sprite):
    asset = ImageAsset("images/ball.png")
    width = 80
    height = 76
    .....
    def __init__(self, position):
        super().__init__(ball.asset, position)
        self.scale = 0.1
```

## 球的方向 反彈方向 球的速度

```
def step(self):
    global gol
    global back1
    if gol:
        self.x += 10
        if self.x > 2000:
            gol = 0
            back1 = 1
    if back1:
        self.x -= 10
        if self.x < -50:
            gol = 1
            back1 = 0
```

## 第二顆球的圖片 方向 大小 反彈方向

```
class ball2(Sprite):  
  
    asset = ImageAsset("images/ball.png")  
    width = 80  
    height = 76  
  
    .....  
  
    def __init__(self, position):  
        super().__init__(ball2.asset, position)  
        self.scale = 0.1  
  
    .....  
  
    def step(self):  
        global go2  
        global back2  
        if go2:  
            self.x += 50  
            if self.x > 1700:  
                go2 = 0  
                back2 = 1  
        if back2:  
            self.x -= 50  
            if self.x < 10:  
                go2 = 1  
                back2 = 0
```

## 執行判段式

```
class DemoApp(App):  
  
    .....  
  
    def __init__(self):  
        super().__init__()  
        Bg((self.width/2, self.height/2))  
        Bunny((800,700))  
        ball1((0,0))  
        ball2((0,100))  
  
    def step(self):  
        """  
        Override step to perform action on each frame update  
        """  
        for bunny in self.spritelist:  
            bunny.step()  
  
# Create the app  
app = DemoApp()  
# Run the app  
app.run()
```