

Piano遊戲設計



大學部人工智慧應用


數位系三年級

吳智鴻

教學網站：[HTTP://120.108.221.55/PROFCHWU/DCTEC](http://120.108.221.55/PROFCHWU/DCTEC)

FB社團： 106 數位系人工智慧

2017/10/11



鋼琴遊戲設計

目標

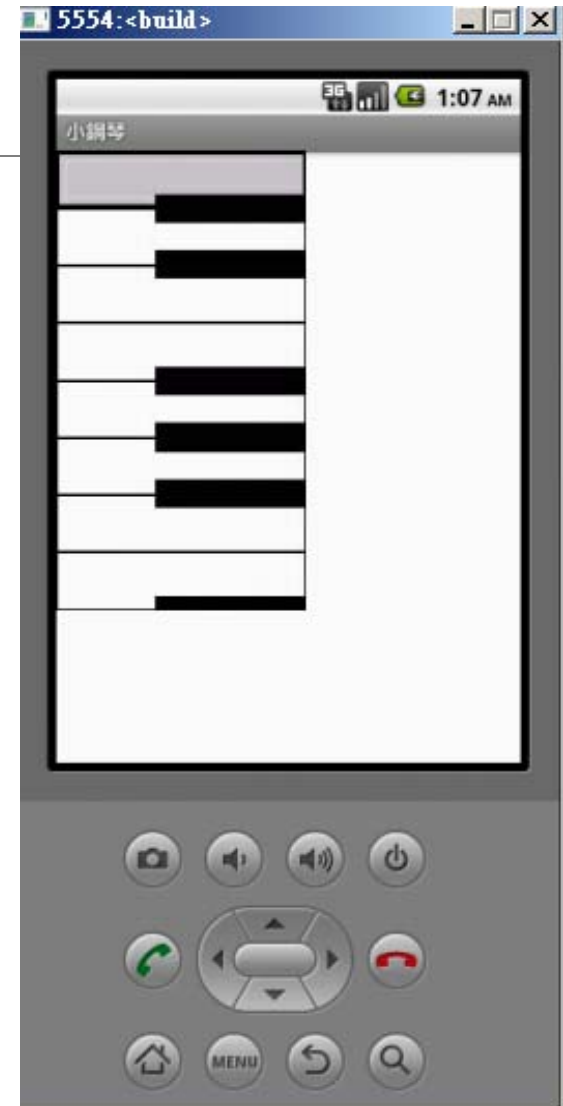
- 設計一個鋼琴模擬遊戲

資料結構

清單（陣列）

演算法

- 依照所按琴鍵，設定對應的圖片。
- 依照所按琴鍵，播放對應的音效。



所需素材（音樂與圖片）



AI2 專案檔



a.mid



b.mid



c.mid



c1.mid



d.mid



e.mid



f.mid



g.mid



piano-1.JPG



piano-2.JPG



piano-3.JPG



piano-4.JPG



piano-5.JPG



piano-6.JPG



需要瞭解的程式技巧

按鈕控制

畫布控制 (TouchDown, TouchUp)

圖片控制

If else判斷

變數 (變數與陣列)

音樂播放

螢幕設計

圖片

ADT105001 小鋼琴桌布板

1
2
3
1
2
2
3
1

Non-visible components

Sound1 Sound2 Sound3 Sound4 Sound5 Sound6 Sound7 Sound8

音樂

c
d
e
f
g
a
b
c1

Components

- Screen1
 - Label1
 - Canvas1
 - Canvas2
 - Canvas3
 - Canvas4
 - Canvas5
 - Canvas6
 - Canvas7
 - Canvas8
 - Sound1
 - Sound2
 - Sound3
 - Sound4
 - Sound5
 - Sound6
 - Sound7
 - Sound8

Properties

Label1

BackgroundColor
 None

FontBold

FontItalic

FontSize
20

FontTypeface
default ▾

HTMLFormat

HasMargins

Height
Automatic...

Width
Automatic...

Text
ADT105001 小鋼琴桌布板

元件設定

元件類別	元件名稱	屬性設定
使用者介面	Screen1	標題->小鋼琴
	按鈕1	圖片->piano-1.jpg 文字->空白
	按鈕2	圖片->piano-2.jpg 文字->空白
	按鈕3	圖片->piano-3.jpg 文字->空白
	按鈕4	圖片->piano-1.jpg 文字->空白
	按鈕5	圖片->piano-2.jpg 文字->空白
	按鈕6	圖片->piano-2.jpg 文字->空白
	按鈕7	圖片->piano-3.jpg 文字->空白
按鈕8	圖片->piano-1.jpg 文字->空白	
多媒體	音效1	來源文件->c.mid
	音效2	來源文件->d.mid
	音效3	來源文件->e.mid
	音效4	來源文件->f.mid
	音效5	來源文件->g.mid
	音效6	來源文件->a.mid
	音效7	來源文件->b.mid
	音效8	來源文件->c1.mid

程式碼

```
當 畫布1 被按壓  
x座標 y座標  
執行 設 畫布1 背景圖片 為 "piano-4.jpg"  
呼叫 音效1 播放
```

```
當 畫布1 被鬆開  
x座標 y座標  
執行 設 畫布1 背景圖片 為 "piano-1.jpg"
```

```
當 畫布2 被按壓  
x座標 y座標  
執行 設 畫布2 背景圖片 為 "piano-5.jpg"  
呼叫 音效2 播放
```

```
當 畫布2 被鬆開  
x座標 y座標  
執行 設 畫布2 背景圖片 為 "piano-2.jpg"
```

```
當 畫布3 被按壓  
x座標 y座標  
執行 設 畫布3 背景圖片 為 "piano-6.jpg"  
呼叫 音效3 播放
```

```
當 畫布3 被鬆開  
x座標 y座標  
執行 設 畫布3 背景圖片 為 "piano-3.jpg"
```

```
當 畫布4 被按壓  
x座標 y座標  
執行 設 畫布4 背景圖片 為 "piano-4.jpg"  
呼叫 音效4 播放
```

```
當 畫布4 被鬆開  
x座標 y座標  
執行 設 畫布4 背景圖片 為 "piano-1.jpg"
```

```
當 畫布5 被按壓  
x座標 y座標  
執行 設 畫布5 背景圖片 為 "piano-5.jpg"  
呼叫 音效5 播放
```

```
當 畫布5 被鬆開  
x座標 y座標  
執行 設 畫布5 背景圖片 為 "piano-2.jpg"
```

```
當 畫布6 被按壓  
x座標 y座標  
執行 設 畫布6 背景圖片 為 "piano-5.jpg"  
呼叫 音效6 播放
```

```
當 畫布6 被鬆開  
x座標 y座標  
執行 設 畫布6 背景圖片 為 "piano-2.jpg"
```

```
當 畫布7 被按壓  
x座標 y座標  
執行 設 畫布7 背景圖片 為 "piano-6.jpg"  
呼叫 音效7 播放
```

```
當 畫布7 被鬆開  
x座標 y座標  
執行 設 畫布7 背景圖片 為 "piano-3.jpg"
```

```
當 畫布8 被按壓  
x座標 y座標  
執行 設 畫布8 背景圖片 為 "piano-4.jpg"  
呼叫 音效8 播放
```

```
當 畫布8 被鬆開  
x座標 y座標  
執行 設 畫布8 背景圖片 為 "piano-1.jpg"
```

加分題

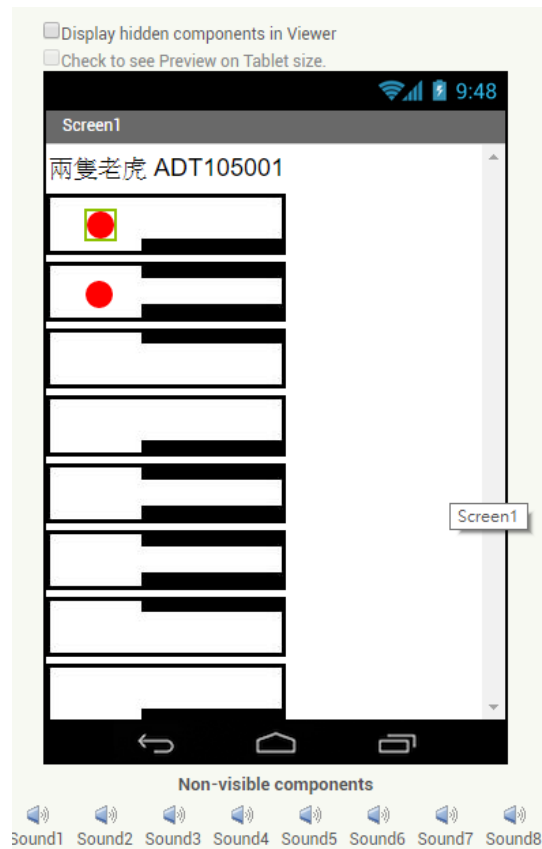
能夠提示使用者琴鍵位置。

能夠提示好幾首曲子。

要如何加上琴鍵
提示呢？
自己練習一下吧。



螢幕與屬性設計



Properties

Ball1

Enabled

Heading 0

Interval 100

PaintColor ■ Red

Radius 10

Speed 0.0

Visible

X 26

Y 9

Z 1.0

The Properties panel shows the configuration for a Ball1 component. It includes a list of properties with their current values: Enabled (checked), Heading (0), Interval (100), PaintColor (Red), Radius (10), Speed (0.0), Visible (checked), X (26), Y (9), and Z (1.0).

琴鍵提示

```
initialize global music_key to create empty list
initialize global max_music to 3
to init_list
do
  add items to list list get global music_key
  item 1
  add items to list list get global music_key
  item 2
  add items to list list get global music_key
  item 3
```

1. 初始化一些變數
2. Music_key為清單，儲存琴鍵的提示
3. 把琴鍵位置依序存入清單中

系統初始化

```
initialize global seq to 0
when Screen1.Initialize
do
  call init_list
  call show_hint
initialize global hint_key to 0
initialize global last_key to 0
```

1. 初始化一些變數
2. seq為變數，每次使用者按下琴鍵就會+1。以便能從Music_key清單中取出下一個使用者需要彈的琴鍵的位置。
3. hint_key為變數，代表目前使用者需要彈的琴鍵的位置。
4. Last_key為變數，代表使用者上一個彈的琴鍵位置。

琴鍵提示主要程式

```
to show_hint
do
  set global seq to (get global seq) + 1
  if (get global seq) > (get global max_music)
  then call reset_music
  set global hint_key to select list item list (get global music_key)
  index (get global seq)
  if (get global hint_key) = 1
  then set Ball1 . Visible to true
  if (get global hint_key) = 2
  then set Ball2 . Visible to true
  if (get global hint_key) = 3
  then set Ball3 . Visible to true
```

```
to clear_hint
do
  if (get global hint_key) = 1
  then set Ball1 . Visible to false
  if (get global hint_key) = 2
  then set Ball2 . Visible to false
  if (get global hint_key) = 3
  then set Ball3 . Visible to false
```

```
to reset_music
do
  set global seq to 1
```

1. Show_hint 副程式：顯示對應的琴鍵提示（ball元件 visible=true）
2. Clear_hint副程式：清除對應的琴鍵提示（ball元件 visible=false）
3. Reset_music副程式：當已經彈完曲子，要重設狀態，避免錯誤。


簡譜對應

簡譜

	1	2	3	4	5	6	7
簡譜	1	2	3	4	5	6	7
So- Fa- name	do	ra	mi	fa	so	la	ti

小蜜蜂簡譜

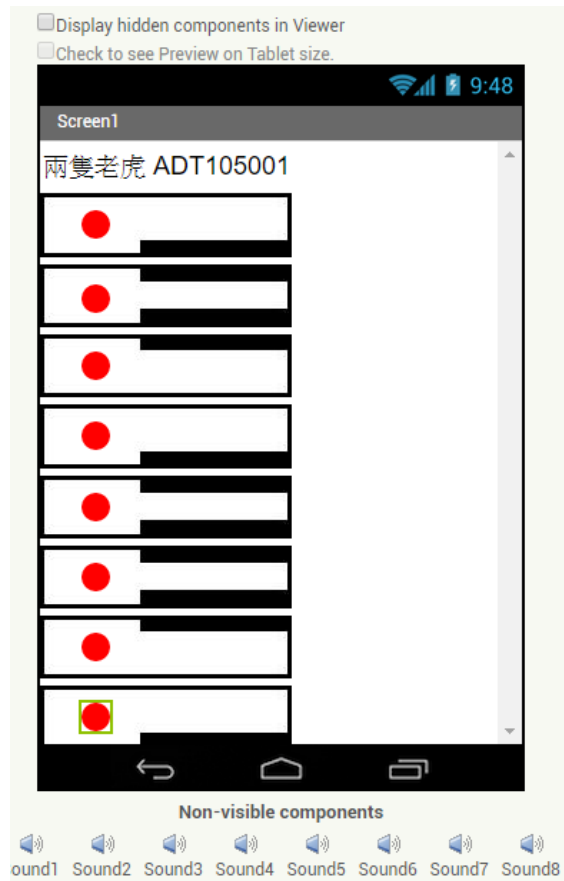
| 5 3 3 - | 4 2 2 - | 1 2 3 4 | 5 5 5 - |
 嗡嗡嗡 嗡嗡嗡 大家一起 勤作工 

| 5 3 3 - | 4 2 2 - | 1 3 5 5 | 3 - - - |
 嗡嗡嗡 嗡嗡嗡 做工趣味 濃  *looking ...*

| 2 2 2 2 | 2 3 4 - | 3 3 3 3 | 3 4 5 - |
 天暖花好 不做工 將來哪裡 好過冬

| 5 3 3 - | 4 2 2 - | 1 3 5 5 | 1 - - - |
 嗡嗡嗡 嗡嗡嗡 別學懶惰 蟲

最後的螢幕設計



BALL元件設計

Properties

Ball8

Enabled

Heading

Interval

PaintColor Red

Radius

Speed

Visible

X

Y

Z

```

to clear_hint
do
  if get global hint_key = 1
  then set Ball1 . Visible to false
  if get global hint_key = 2
  then set Ball2 . Visible to false
  if get global hint_key = 3
  then set Ball3 . Visible to false
  if get global hint_key = 4
  then set Ball4 . Visible to false
  if get global hint_key = 5
  then set Ball5 . Visible to false

```

```

to show_hint
do
  set global seq to get global seq + 1
  if get global seq > get global max_music
  then call reset_music
  set global hint_key to select list item list get global music_key
  index get global seq
  if get global hint_key = 1
  then set Ball1 . Visible to true
  if get global hint_key = 2
  then set Ball2 . Visible to true
  if get global hint_key = 3
  then set Ball3 . Visible to true
  if get global hint_key = 4
  then set Ball4 . Visible to true
  if get global hint_key = 5
  then set Ball5 . Visible to true

```

```

to reset_music
do
  set global seq to 1

```

```

to init_list
do
  add items to list list get global music_key
  item 5
  add items to list list get global music_key
  item 3
  add items to list list get global music_key
  item 3
  add items to list list get global music_key
  item 4
  add items to list list get global music_key
  item 2
  add items to list list get global music_key
  item 2
  add items to list list get global music_key
  item 1
  add items to list list get global music_key
  item 2
  add items to list list get global music_key
  item 3
  add items to list list get global music_key
  item 4
  add items to list list get global music_key
  item 5
  add items to list list get global music_key
  item 5
  add items to list list get global music_key
  item 5
  set global max_music to length of list list get global music_key

```




```
when Canvas1 .TouchDown
  x y
do call clear_hint
  set Canvas1 . BackgroundImage to " piano-4.jpg "
  call Sound1 .Play
```

```
when Canvas2 .TouchDown
  x y
do call clear_hint
  set Canvas2 . BackgroundImage to " piano-5.jpg "
  call Sound2 .Play
```

```
when Canvas3 .TouchDown
  x y
do call clear_hint
  set Canvas3 . BackgroundImage to " piano-6.jpg "
  call Sound3 .Play
```

```
when Canvas4 .TouchDown
  x y
do call clear_hint
  set Canvas4 . BackgroundImage to " piano-4.jpg "
  call Sound4 .Play
```

```
when Canvas5 .TouchDown
  x y
do call clear_hint
  set Canvas5 . BackgroundImage to " piano-5.jpg "
  call Sound5 .Play
```

```
when Canvas6 .TouchDown
  x y
do call clear_hint
```

```
when Canvas1 .TouchUp
  x y
do call show_hint
  set Canvas1 . BackgroundImage to " piano-1.jpg "
```

```
when Canvas2 .TouchUp
  x y
do call show_hint
  set Canvas2 . BackgroundImage to " piano-2.jpg "
```

```
when Canvas3 .TouchUp
  x y
do call show_hint
  set Canvas3 . BackgroundImage to " piano-3.jpg "
```

```
when Canvas4 .TouchUp
  x y
do call show_hint
  set Canvas4 . BackgroundImage to " piano-1.jpg "
```

```
when Canvas5 .TouchUp
  x y
do call show_hint
  set Canvas5 . BackgroundImage to " piano-2.jpg "
```

```
when Canvas6 .TouchUp
  x y
do call show_hint
```



剩下的就讓你自
己發揮了



後續可以延伸

增加更多的曲子提示。

曲子提示可以從資料庫取得

可以記錄使用者正確彈奏/錯誤比率