

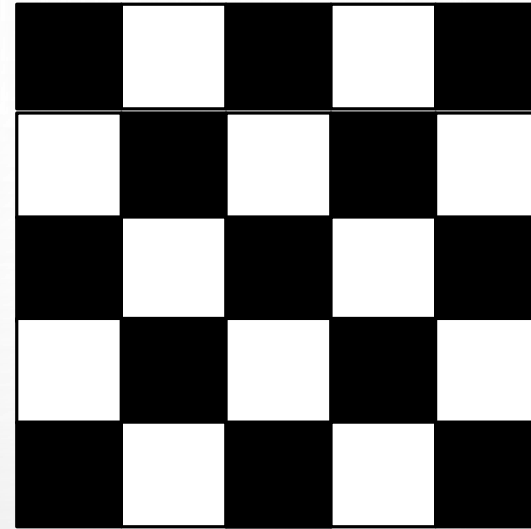
The background of the slide is a light gray gradient. It is decorated with numerous realistic water droplets of various sizes. Some droplets are large and prominent, while others are small and scattered. They are primarily located in the top-left and bottom-right corners, with a few smaller ones in the center and along the edges. The droplets have a glossy, 3D appearance with highlights and shadows.

從零開始的影像處理

CHAP0: PYTHON進階程式設計

PRACTICE

1. 生出500*500的黑白影像如右圖：



2. 生出500*500的黑白影像並繪製出 $Y = x^{2/3}$ 的曲線圖

(提示：計算出Y之後在陣列中(x,Y)的位置填入1 OR 255)

(提示：超過範圍可以不畫，或者是調整Y軸的比例)

(提示：想要讓線變粗可以把附近的幾格也填入顏色)

再多一點點NUMPY-1

1. 執行速度>>[LIST]
2. 可以利用NP.ONES, NP.ZEROS以及NP.EMPTY來初始值
3. 可以利用RESHAPE來改變陣列形狀(元素的數目需要一致)

EX1: 執行下面程式並比較結果

```
import numpy as np
a = np.arange(10000)
%time for i in range(100): a2= a * 2

b = range(10000)
%time for i in range(100): b2 = [x * 2 for x in b]
```

再多一點點NUMPY-1

EX2: 執行下面程式並觀察結果

```
import numpy as np
import cv2

empty = np.empty(10)
print(empty)

x = np.array([[1,2,3,4,5,6],[6,7,8,9,10,11]])
print(x.reshape(3,4))
print(x.T)
print(x[0])
print(x[1,2:5])
```

$$x = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 6 & 7 & 8 & 9 & 10 & 11 \end{bmatrix}$$

```
[[ 1  2  3  4]
 [ 5  6  6  7]
 [ 8  9 10 11]]
[[ 1  6]
 [ 2  7]
 [ 3  8]
 [ 4  9]
 [ 5 10]
 [ 6 11]]
[1 2 3 4 5 6]
[ 8  9 10]
```

再多一點點NUMPY-1

- [2:5] 這種指定陣列內元素的使用方式稱為切片(SLICE) ,
 - : 冒號表示頭尾, 例如: [1:4] 表示編號1到編號3元素 [:] 則是全部
 - , 表示隔開維度

```
a = np.array([[0,1,2,3,4],[5,6,7,8,9],[10,11,12,13,14]])
print(a[1])
print(a[1:])
print(a[1,2:])
print(a[:2,2:])
print(a[:,2:4])
```

$$a = \begin{bmatrix} 0 & 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 & 9 \\ 10 & 11 & 12 & 13 & 14 \end{bmatrix}$$

- -1 表示最後一個元素,-2表示倒數第二個元素, ...
- [START:END:STEP]

```
print(a[:,2,::-2]) = ?
```

```
print(a[-1])
print(a[:, -1])
print(a[-2, -1])
```

```
print(a[:, ::2])
print(a[:, ::3])
print(a[:, ::-1])
```

再多一點點NUMPY-1

RESHAPE是照著元素順序重新排序，所以...

EX4: 執行下面程式並觀察結果

```
y = np.zeros([500,500],dtype="uint8")
for i in range(5):
    for j in range(5):
        if((i+j) % 2 == 0):
            y[i*100:i*100+100,j*100:j*100+100] = 255;

cv2.imshow("test",y);
cv2.waitKey();

z = y.reshape(100,2500);
cv2.imshow("test_reshape",z);
cv2.waitKey();
```

再多一點點NUMPY-1

- RESHAPE是照著元素順序重新排序，所以...
- RESHAPE沒辦法對影像作想像中的“改變形狀”
- 元素的總數目固定，所以也不能“改變大小”
- THINK: 如何改變影像的大小?
 - 下一堂影像處理課會教，不過可以先看看OPENCV的函數

```
cv2.imshow("test_resize",cv2.resize(y,(500,100)))  
cv2.waitKey();
```

再多一點點NUMPY-2

- 亂數RANDOM
- 排序SORT
- 刪除重複的元素UNIQUE

```
import numpy as np
a = np.random.randint(1, 100, 20)
print(a)
a.sort()
print(a)
print(np.unique(a))

b = np.array([[2,1,-1,3,0],[4,2,3,4,1]])
print(b)
b.sort()
print(b)
print(np.unique(b))
```


關於OPENCV

- OPENCV(**OPEN** SOURCE **C**OMPUTER **V**ISION LIBRARY)是一個跨平臺的電腦視覺資料庫，由INTEL發起並參與開發，可以執行在WINDOWS、ANDROID、IOS、OPENBSD和MAC OS作業系統上。
- OPENCV用C++語言編寫，它的主要介面也是C++語言，但是依然保留了大量的C語言介面。該庫也有大量的PYTHON, JAVA AND MATLAB/OCTAVE 的介面。現在也提供對於C#, RUBY等的支援。

載入OPENCV

- 在ANACONDA PROMPT 或是 CMD畫面下輸入

```
>pip install opencv-python
```

- 之後在PYTHON程式中輸入 `IMPORT CV2` 即可使用
- 也可以利用`CV2.__VERSION__` 確認目前的OPENCV版本
- 指令可參考
 - [HTTPS://DOCS.OPENCV.ORG/4.4.0/](https://docs.opencv.org/4.4.0/)

OPENCV繪圖

- OPENCV可用來繪製直線、矩形、圓形、橢圓形、多邊形等圖形
- CV2.LINE
- CV2.RECTANGLE
- CV2.CIRCLE
- CV2.ELLIPSE
- CV2.POLYLINES
- CV2.FILLPOLY
- CV2.PUTTEXT

參考課本的P2-23~P2-27

CV2.RECTANGLE

- CV2.RECTANGLE(IMG,PT1,PT2,COLOR,[THICKNESS])

要繪製上去的目標圖片(ndarray) 矩形的第一個點座標 矩形的第二個點座標 顏色BGR順序 邊框的粗細。-1 = 填滿

- EX: 執行以下程式並觀察結果

```
import numpy as np
import cv2

img = np.zeros([500,500],dtype = "uint8")
cv2.rectangle(img,(100,100),(200,200),(255,255,255),-1)

cv2.imshow("img",img)
cv2.waitKey()

cv2.rectangle(img,(100,200),(300,400),(255,0,0),-1)

cv2.imshow("img",img)
cv2.waitKey()
```

PRACTICE

1. 繪製出之前的5*5棋盤，但利用RECTANGLE讓程式碼更精簡
2. 學校ABC三班的學生成績分別為

A班: [80,97,89,65,90]

B班: [50,77,100,91,20]

C班: [78,60,56,75,79]

請將ABC三班的成績繪製成長條圖
並標記出標準差如右圖:

