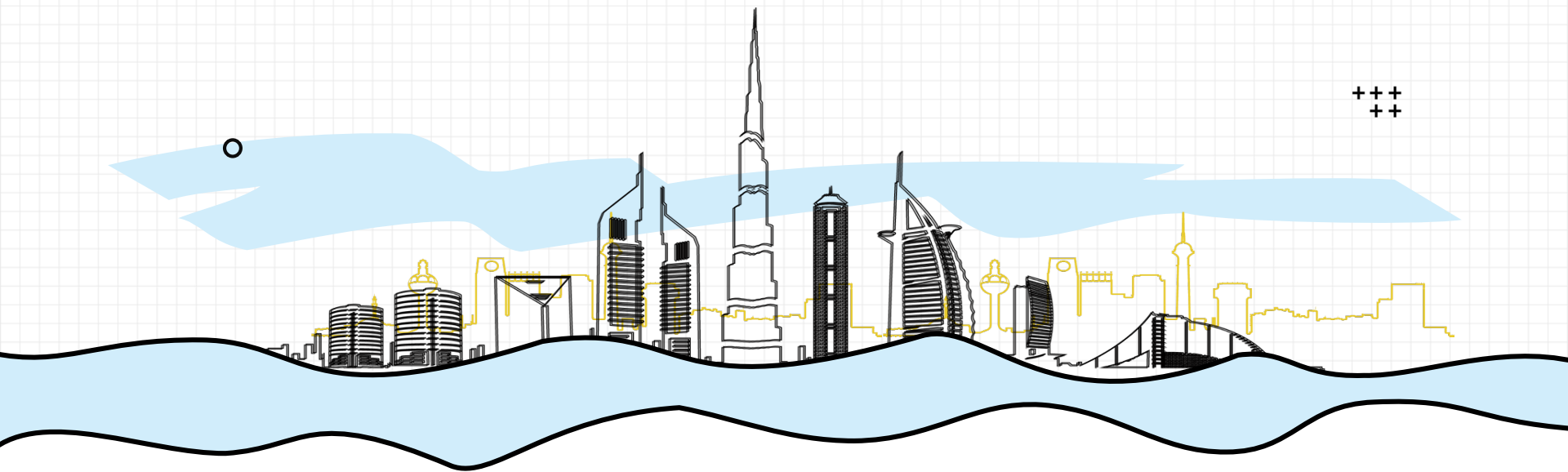




棋 文 共 賞

第七組 資安所 108164504 徐玄昌
資安所 110164515 陳昶安
資工所 110062649 楊煥竣
資應所 110065538 林郁凱

Outline

01

團隊分工

Project Composed & Teamwork

02

作品論述&執行方法

Project Statement & System Architecture & Implementation

03

作品展示

Project Demo

04

Connection & QA

Connection & QA

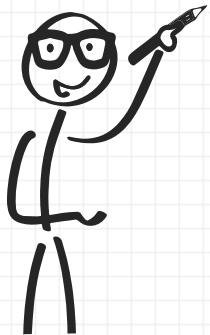


01

PART
ONE

團隊分工

Project Composed & Teamwork

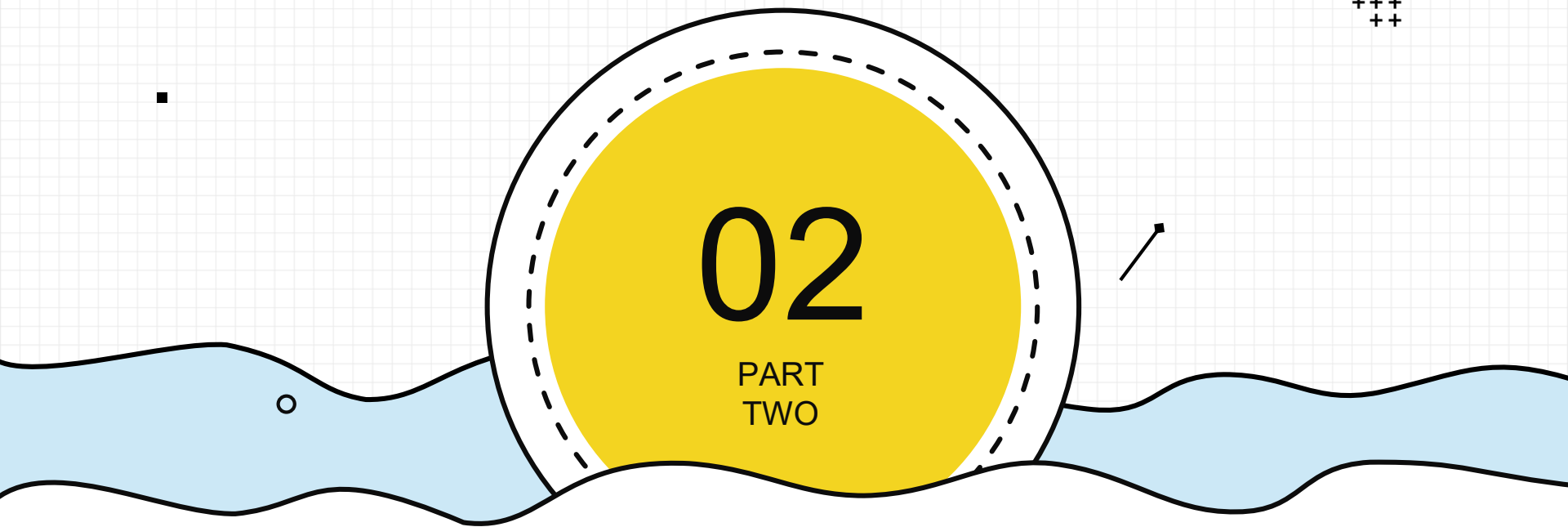


Project Composed & Teamwork

- Environment Setting : ■ ■
 - server and database settings
- Front-end : ■ ■ ■ ■
 - web and script
 - html5
 - css
 - javascript
 - python

- Backend :
 - web crawler ■ ■
 - python django ■
 - bitmap encoder ■ ■
 - data analysis ■ ■ ■ ■
 - jieba NLP tool ■ ■
- Slides & Proposal :
 - slides ■ ■ ■ ■
 - oral presentation ■ ■ ■ ■



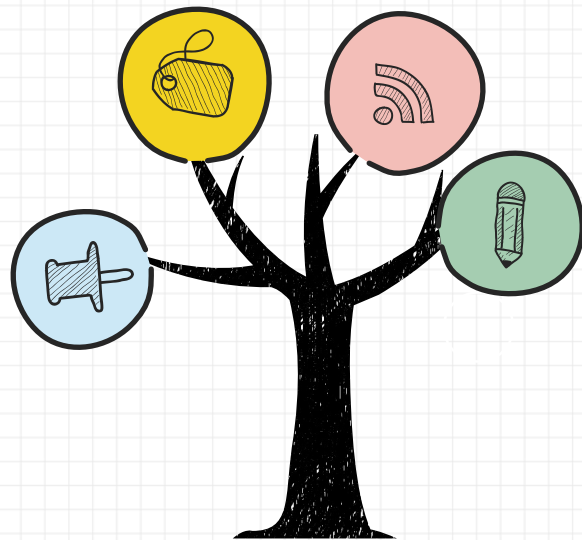


作品論述&執行方法

Project Statement & System Architecture & Implementation

作品論述

- 棋文共賞--在網路上的許多文章中，文章底下的留言往往有人在底下吵架或是有正反論戰，但是這些爭論通常吵來吵去都沒有結果。所以我們想說可以利用把留言變成下五子棋的方式，以一個好奇有趣的心態，來呈現正反方大戰的結果。



System Architecture - Priority Mode

0. Start up Django Server on Oracle Cloud Machine.

1. Pre-processing :

1.1 Input article URL.

1.2 Use URL to drive the web crawler.

1.3 Store the raw data to Database.

1.4 Use NLP model to parsing the raw data.

1.5 Analysis the words' frequency with NLP parsing.

1.6 Assign specific priority to each message by indexing the word' frequency, and other indexes.

2. Runtime :

2.1 Dynamic generate the runtime priority map.

2.2 By the property of the message, deciding that which color it belongs to.

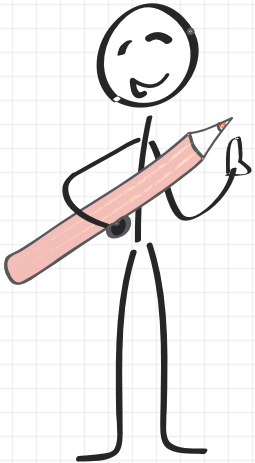
2.3 Due to the message's priority, decide where the chess can be place.

2.4 Simultaneously, print out the chess' message to the bottom of bulletin board.

2.5 Judge the current chess table:

2.5.1 If anyone wins the game, stop and show the winner.

2.5.2 else, continue to back to step 2.1 until end.



Priority Mechanism

棋局開始前：

- 當抓取原始資料時，先針對所有的留言進行關鍵字的擷取

- 依照關鍵詞出現的頻率給定不同的權狀值

(eg. 若抓取的文章關鍵詞有5個，分別為： A_1, A_2, A_3, A_4, A_5

其字詞出現次數分別為： F_1, F_2, F_3, F_4, F_5

給予 A_i 這個關鍵詞的權重值為 $W_i = \frac{F_i}{\sum_j F_j}$)

- 對於每筆留言，進行關鍵詞的檢索，若此筆留言有出現的該關鍵詞，則對此加上權重

(eg. 若此留言 M_1 中出現的關鍵詞有3個，分別為： A_1, A_2, A_5

其字詞出現次數分別為： C_1, C_2, C_5

給予 M_1 這個留言權重： $C_1W_1 + C_2W_2 + C_5W_5 + Bias_{length}$)

對於所有留言權重做重新分配，根據留言的權重分佈分成 5 等分，依據重要程度分為 1 ~ 5 級

Priority Mechanism

當棋局開始時：

先動態產生棋盤地形，產生方法為對於每個棋盤上的點做 $5 * 5$ 的卷積操作，蒐集周邊的棋子狀態決定該點的相對地形值，並區分成對黑白子分別有利的 5 等分

(eg. 初始時每個點皆為空的，所以皆會卷積出相同值，

即在三維空間內產生一個的水平面)

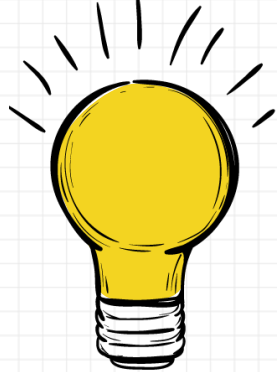
依據棋子本身的重要程度 (1 ~ 5)，蒐集對應的點能下的地圖相對位置，隨機決定下的點

當棋子落下時，會依據其顏色改變周圍地圖的地形，權重值則為其改變地形的程度

(eg. 若以空間比喻，黑色為向上長出山丘，白色為向下挖出低谷；

其權重值則為其改變地形的深度)

Implementation



- 利用爬蟲從ptt抓取特定主題的文章討論串
- 然後根據該篇文章底下的推文進行分析調配權重
- 根據上之結果用網頁動態的譜出一盤棋
- 用棋局的結果來反映留言對該主題的想法

+++
++

03

PART
THREE

作品展示

Project Demo

Demo

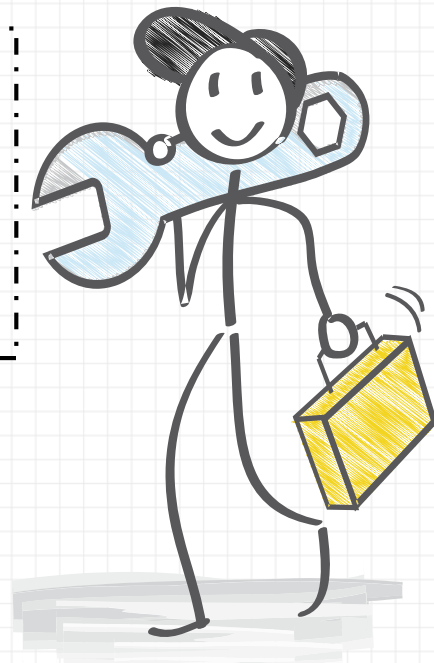
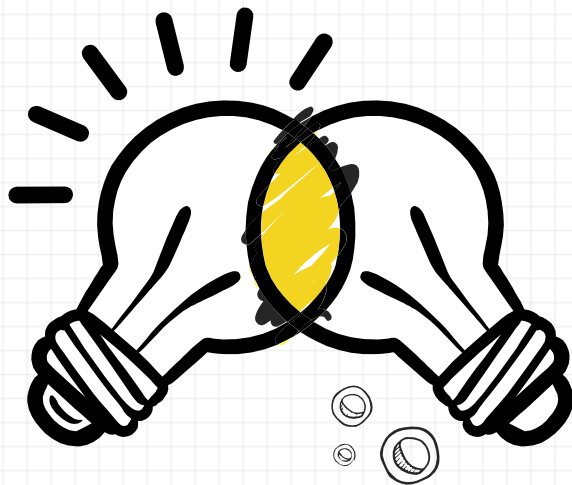
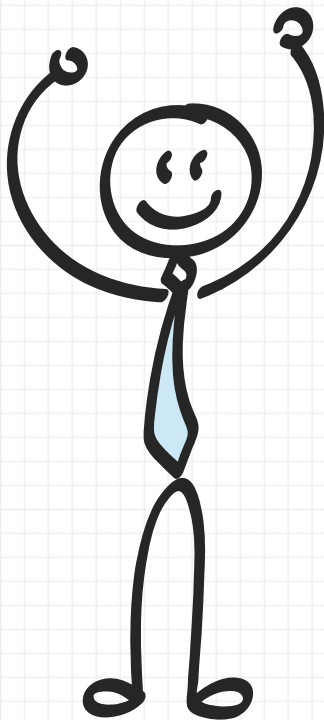
實驗平台：chrome 無痕模式

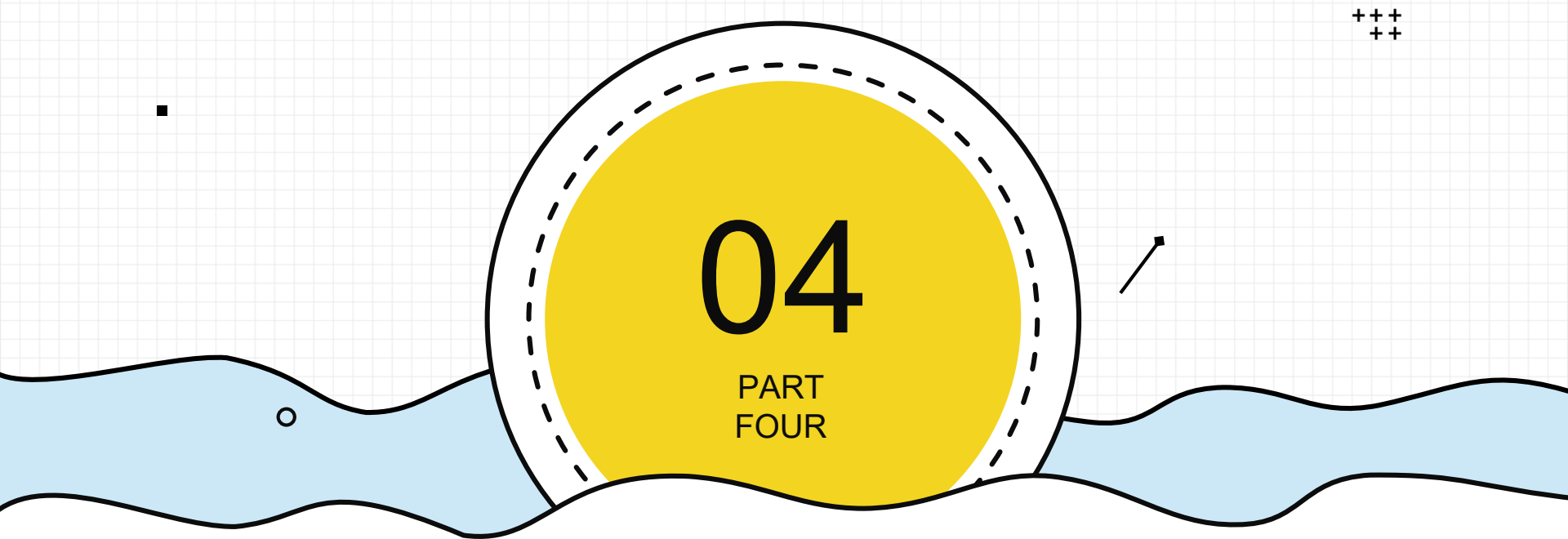
連線網址：168.138.51.50:8080

Submit URL：限定ptt.cc相關文章網站

(有些時候爬蟲會因網路關係造成delay，可重新輸入網址再送出一次request等待成功)

文章限制：需有推 / 噓各100則留言以上方能成功進行模擬





Connection & QA

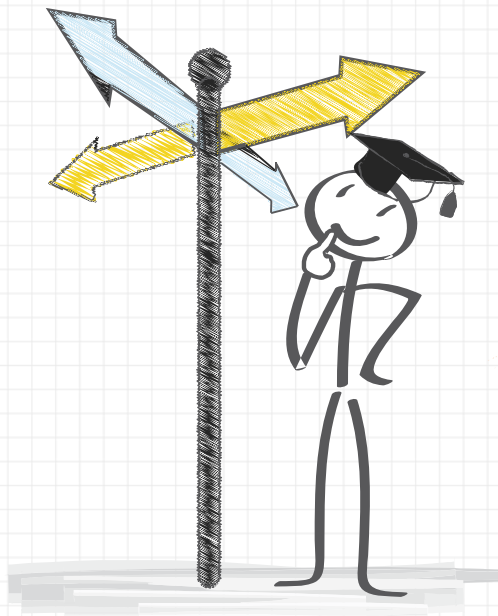
Connection & QA

Connection

第一次的小組報告：年度之字

我們從每篇文章中收集留言，並在每篇文章的留言裡找出有出現最多次的詞彙當作關鍵字，這就有點類似學期初的年度之字的報告。年度之字也是收集該年所發生的種種事情，並從中去找出最能影響當年甚鉅然後可以代表當年的關鍵字；而我們找出該篇文章留言的關鍵詞/字也頗有相同的意味。

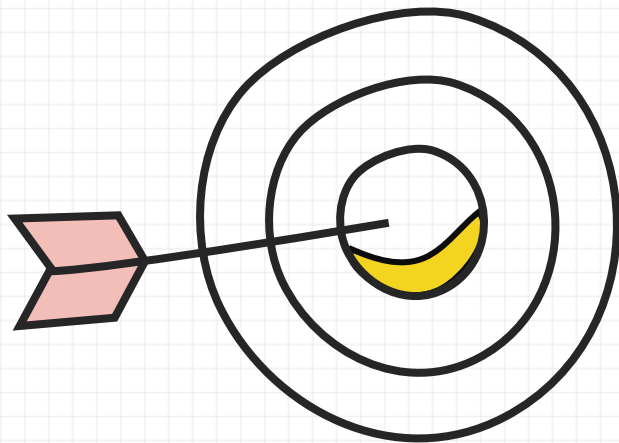
選出來的關鍵詞可以代表說他是該篇文章的留言中具有很重要的權重值，也因為它有很重要的權重值，所以他很有可能是影響整篇文章留言的風向指標，可能許多人都因為看到那則留言而紛紛改變他的想法，從而改變成支持那則留言的聲音；但也有可能那則留言是掀起對立的開端，那篇文章底下的留言都因那則權重值高的留言開始而開始吵架。



心得、啟發、聯想

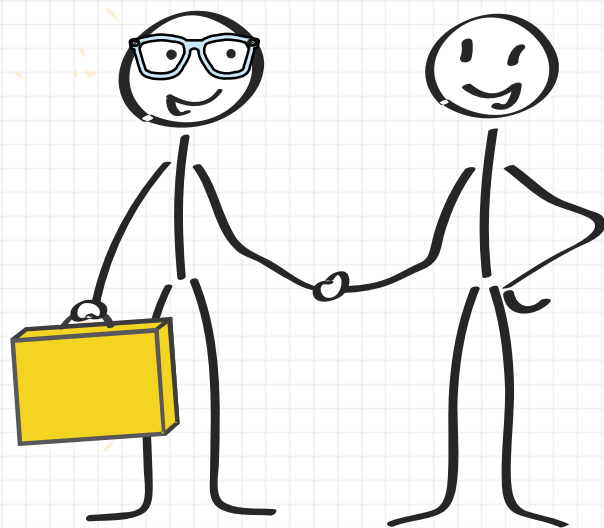
在開發的過程中，我們從中發現每個人的留言，背後都蘊含著那個人的思想、意識型態和精神。對於同一個話題來說，正方、反方的留言都是有意義的，當然也有存在那些不支持也不反對的留言，但是這不代表說支持的留言就一定對，而反對的留言就必錯這樣。

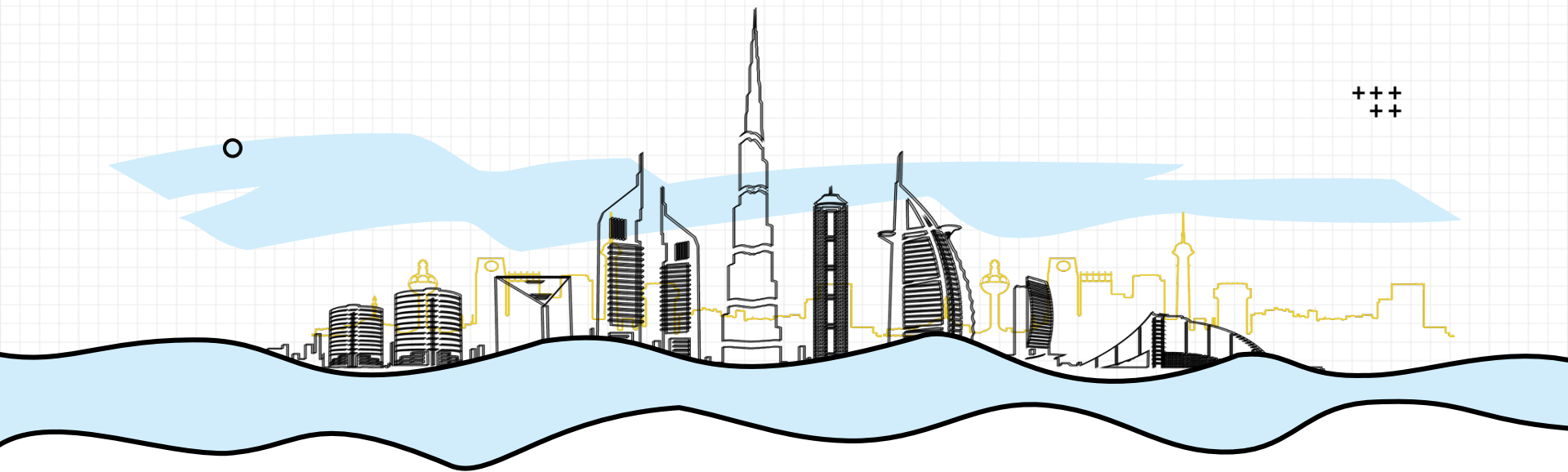
我們應該以平常心去看待每則留言，尊重每個人都有發言的權利。也許別人的留言很偏激、很煽動，但是我們應該秉持著就事論事的精神，理性地去評斷每件事情，這樣才能有效率地進行討論並且達到良性的溝通從而解決問題，這樣才是接受高等教育的我們理想中的景象而非一言不合就翻桌。



Q&A

- 對於網路上的吵架，有沒有人有其他有趣的想法可以更公平或是更有效率的分出勝負？
- 對於利用五子棋分出勝負，大家是不是覺得這樣對於一篇文章的勝負更清楚易懂了呢？





T H A N K S