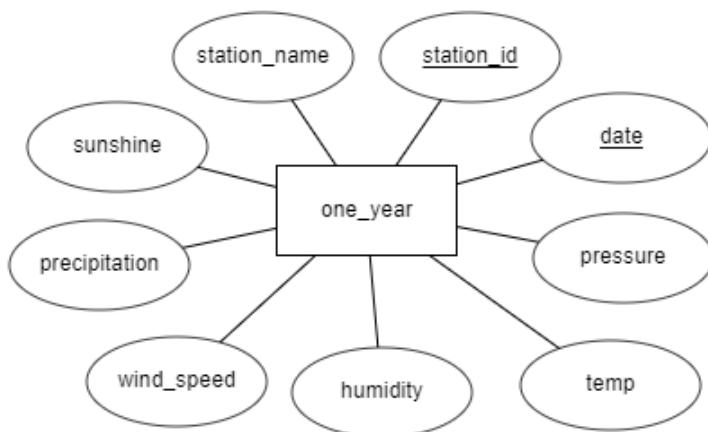


Project_Team15

壹、Data

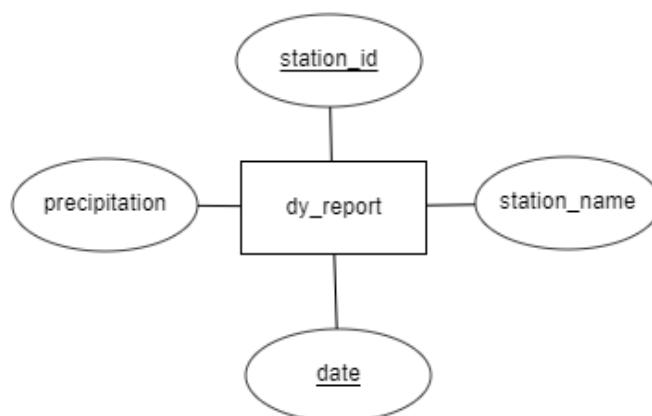
一、一年觀測資料——本局屬地面測站一年觀測資料(one_year)

- 1.介紹：來自氣象局的資料，包含測站名、測站ID、每日詳細的氣象觀測資料。每一小時一筆資料。
- 2.Primary key:{station_id, date}, foreign key:{station_id}(參考station的station_id)
- 3.欄位：station_name, station_id, date, pressure, temp, humidity, wind_speed, wind_dir(風向), precipitation(降水), sunshine(每小時日照時數)
- 4.資料筆數：約二十七萬筆
- 5.補充說明：氣象局只有提供一年份每天的資料，怕資料量太少，不利於統計的準確度，因此還有找另一個資料(每日雨量)來補足。



二、每日雨量——過去9年局屬地面測站每日雨量資料(dy_Report)

- 1.介紹：來自氣象局的資料，包含最近九年每天的資料，但是只有雨量的資料。由於當天是否下雨對於出門遊玩影響很大，我們將雨量特別列出考慮。
- 2.Primary key:{station_id, date}, foreign key:{station_id}(參考station的station_id)
- 3.欄位：station_id, station_name, date, precipitation
- 4.資料筆數：約十萬筆



三、景點 —— 觀光資訊資料庫(attraction)

1.介紹：包含35個column，有景點的詳細資料、營業時間、票價資訊等。總共有5000多個景點。

2.Primary key:{id}

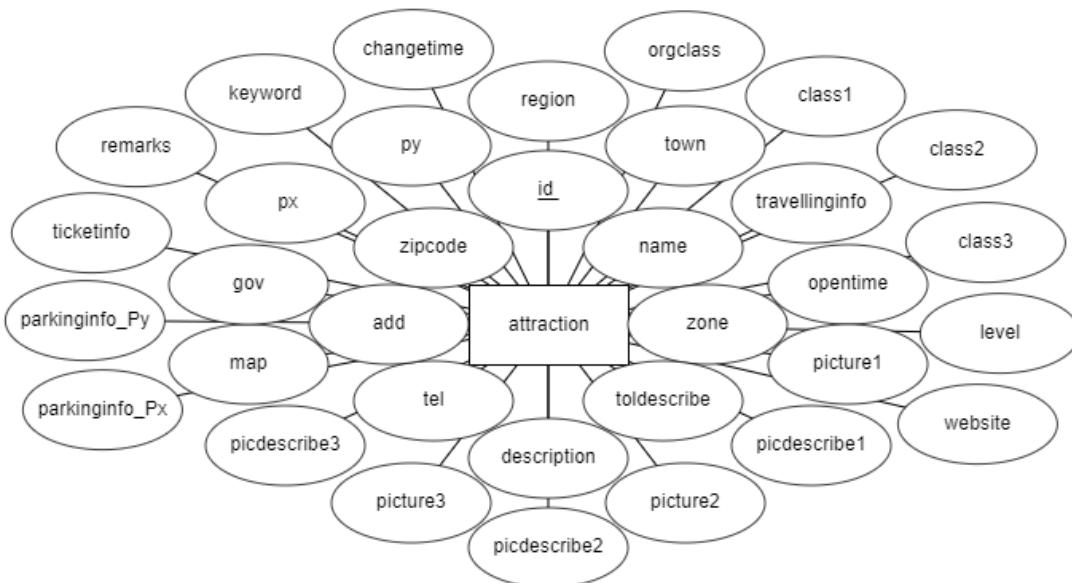
3.欄位：id, name, zone(景區編號), toldescribe(詳細文字介紹), description(精簡文字介紹), tel(電話), add(地址), zipcode, region, town, travellinginfo(交通資訊), opentime, picture1(圖片url), picdescribe1, picture2(圖片url), picdescribe2, picture3(圖片url), picdescribe3, map(地圖url), gov(管理單位代碼), px, py, orgclass(觀光資料生產單位之景點分類), class1, class2, class3, level(古蹟分級), website, parkinginfo, parkinginfo_Px, parkinginfo_Py, ticketinfo, remarks(注意事項), keyword, changetime(資料更新時間)

4.資料筆數：約五千筆

5.補充說明：

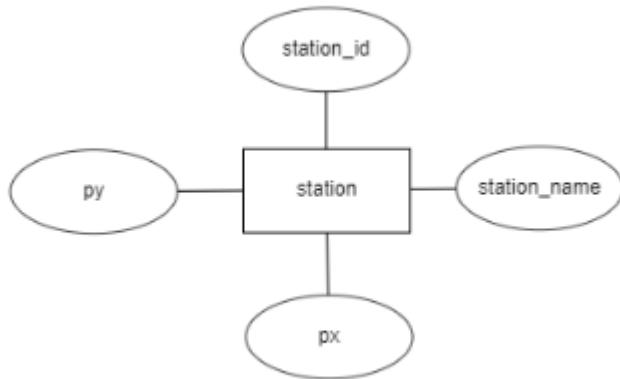
(1)class1, class2, class3為景點分類代碼(複選，選擇下列最接近之三種分類：01.文化類、02.生態類、03.古蹟類、04.廟宇類、05.藝術類、06.小吃/特產類、07.國家公園類、08.國家風景區類、09.休閒農業類、10.溫泉類、11.自然風景類、12.遊憩類、13.體育健身類、14.觀光工廠類、15.都會公園類、16.森林遊樂區類、17.林場類、18.其他)

(2)這次網站沒有讓使用者新增景點是因為這個table的id不是單純的數字，是當初政府使用的特殊編號(中/英文標示+資料類別代碼_資料生產單位代碼_六碼流水號)。因此使用者要新增資料時，很難自動產生一個可用的id。



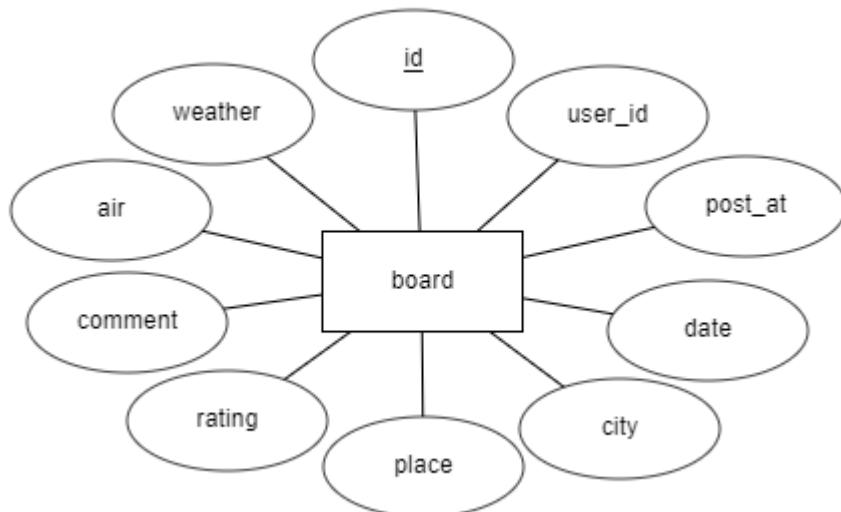
四、氣象站資訊(station)

1. 介紹：提供測站的經緯度座標
2. Primary key : {station_id}
3. 欄位 : station_id, station_name, px, py
4. 資料筆數 : 35筆
5. 補充說明：從網路上找到測站的資料手動輸入，目的是為了連結不同的table。attraction有景點的經緯度資料，one_year和dy_Report有測站名稱及id，沒有經緯度的資訊。這個table可以把前面三個table串聯起來。

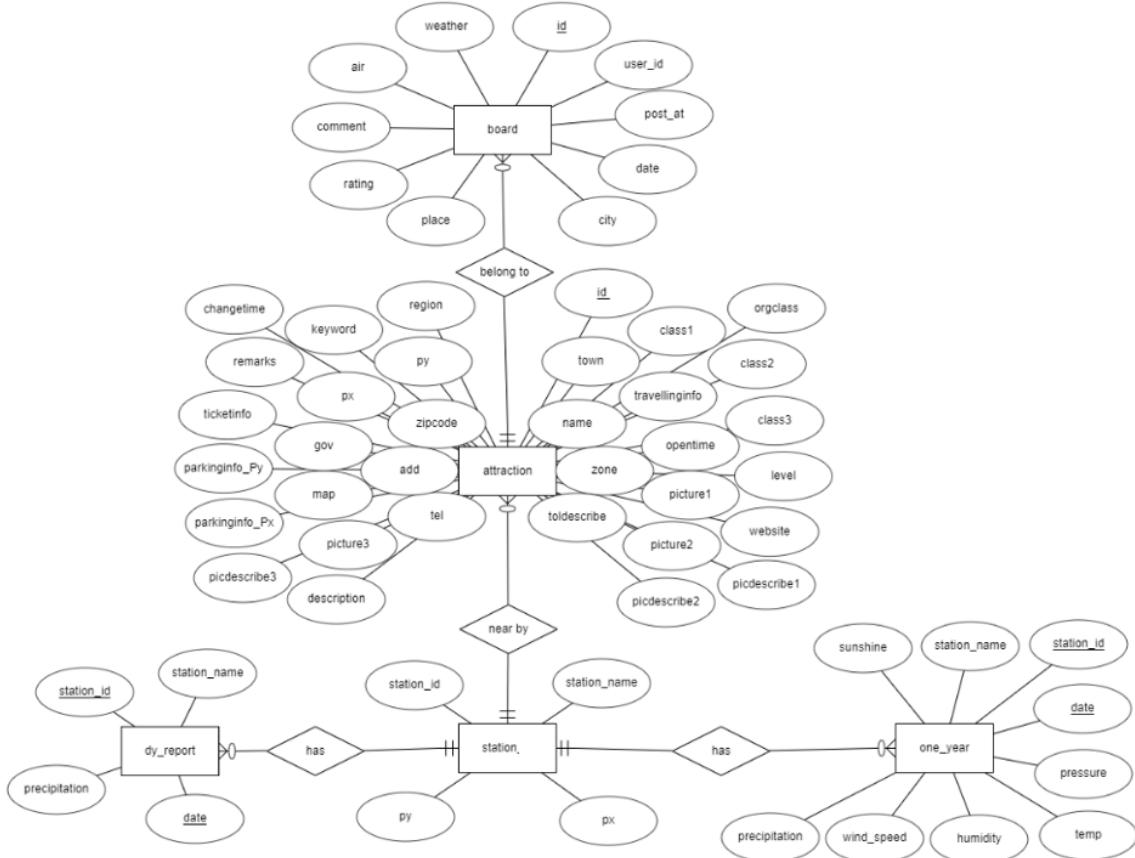


五、留言板(board)

1. 介紹：這是我們自己設計的table，可以記錄用戶的留言。包含景點、旅遊日期及當時的相關天氣資訊。
2. Primary key : {id}
3. 欄位 : id, user_id, post_at(留言發布時間), date(旅遊時間), city, place(景點名稱), rating, comment, air(空氣品質), weather(天氣好壞)
4. 補充說明：
 - (1)place可以用來和attraction的name比對。如果留言的景點名不在attraction中就不能留言，board與attraction為多對一關係。
 - (2)有評分的欄位像是rating、air、weather的平均評分會在搜尋景點時和天氣資訊一起顯示出來。
 - (3)id有設定為流水號，不會出現相同id的留言。



六、整合後的ER model



七、normalization

我們沒有在天氣資料one_year以及dy_Report加上測站的座標，而是開了一個新的table叫做station，用station_id搜尋可以找到該測站的座標。因此我們可將attraction與天氣資料連結，先找到每個景點最鄰近的測站，再透過station_id搜尋天氣，符合正規化的精神，降低了資料的重複儲存，也沒有破壞他們的函數依賴關係。

而天氣資料的部分因為原始資料為非正規化形式，無法匯入資料庫，因此我們用jq這個工具對這部分的資料進行正規化。

貳、Database

我們下載XAMPP利用MySQL處理資料庫的部分，一開始插入開放資料時，是直接用phpMyAdmin新增資料。和網頁相關的部分我們用PDO連接資料庫，以下是留言板部分連接資料庫的寫法。天氣搜地點、地點搜天氣的部分也是類似寫法。

```
public function dbConnect(){
    $host = DB_HOST;
    $dbname = DB_NAME;
    $user = DB_USER;
    $pass = DB_PASS;
    $dsn = "mysql:host=$host;dbname=$dbname;charset=utf8"; //改dbname

    try{
        $dbh = new PDO($dsn, $user, $pass,[ PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION, ]);
    } catch(PDOException $e){
        echo ' fail to connect'. $e->getMessage();
        exit();
    };
    return $dbh;
}
```

若連線失敗，會顯示
erro message。

由網頁對資料庫進行更動的部分，先說明留言板部分。

一、新增留言：

```
1  <?php
2  ini_set('display_errors', "On");
3  require_once('blog.php');
4  $blogs = $_POST;
5  $blog = new Blog();
6  $blog->blogValidate($blogs);
7  $blog->blogCreate($blogs);
8  ?>
9  <p><a href="/index_comment.php">返回留言板</a></p>
10 <p><a href="/">返回首頁</a></p>
```

blogValidate function會對使用者輸入的資料進行檢查，如果有任何一個欄位沒有填寫，送出資料後會有提醒，例如：

```
if(empty($blogs['user_id'])){
    exit('請輸入使用者ID');
}
```

另外，景點的名稱我們有限制不能超過20字(景點資料庫中皆 ≤ 20 字)。

```
if(mb_strlen($blogs['place']) > 20){
    exit('請輸入景點名稱(20字以內)');
}
```

留言的景點名稱有先和資料庫做比對，若有一模一樣的景點名稱，使用者才能新增留言，如此一來，使用者搜尋景點詳細資料時，顯示的使用者評價一定會是正確的資料，不會有使用者新增留言時對同一個景點的稱呼不同，而導致顯示的景點使用者評價對應不到正確資料的問題。

對於景點的評價(評分、天氣好壞、空氣好壞)在使用者介面用checkbox讓使用者勾選，所以不會有超出範圍、無此選項的問題。

確認輸入的資料沒問題後，會使用blog create function新增留言資料，如下圖：

```

public function blogCreate($blogs){
    $sql = "INSERT INTO
        |     $this->table_name(user_id, date, city, place, rating, comment, air, weather)
        |     VALUES
        |         (:user_id, :date, :city, :place, :rating, :comment, :air, :weather)";
    $dbh = $this->dbConnect();
    $dbh->beginTransaction();
    try{
        $stmt = $dbh->prepare($sql);
        $stmt->bindValue(':user_id', $blogs['user_id'], PDO::PARAM_INT);
        $stmt->bindValue(':date', $blogs['date'], PDO::PARAM_STR);
        $stmt->bindValue(':city', $blogs['city'], PDO::PARAM_STR);
        $stmt->bindValue(':place', $blogs['place'], PDO::PARAM_STR);
        $stmt->bindValue(':rating', $blogs['rating'], PDO::PARAM_INT);
        $stmt->bindValue(':comment', $blogs['comment'], PDO::PARAM_STR);
        $stmt->bindValue(':air', $blogs['air'], PDO::PARAM_INT);
        $stmt->bindValue(':weather', $blogs['weather'], PDO::PARAM_INT);
        $stmt->execute();
        $dbh->commit();
        echo '已新增心得';
    }catch(PDOException $e){
        $dbh->rollBack();
        exit($e);
    }
}

```

這個function會呼叫dbConnect function連接資料庫，將使用者輸入的資料連接成正確的sql。

beginTrancsaction可關閉自動提交，直到出現commit或rollback才會恢復。

commit會將在beginTrancsaction之間的指令送給資料庫，並開啟自動提交。

rollback在出現錯誤時，不會使在beginTrancsaction之間的指令對資料庫造成影響，並開啟自動提交。

以上三個function可以避免錯誤的sql影響資料庫。

若無法正確的新增資料，網頁上會顯示error message。

二、更改留言：

```

public function blogUpdate($blogs){
    $sql = "UPDATE $this->table_name SET
            user_id = :user_id, date = :date, city = :city, place = :place,
            WHERE
            id = :id";
    $dbh = $this->dbConnect();
    $dbh->beginTransaction();
    try{
        $stmt = $dbh->prepare($sql);
        $stmt->bindValue(':id', $blogs['id'], PDO::PARAM_INT);
        $stmt->bindValue(':user_id', $blogs['user_id'], PDO::PARAM_INT);
        $stmt->bindValue(':date', $blogs['date'], PDO::PARAM_STR);
        $stmt->bindValue(':city', $blogs['city'], PDO::PARAM_STR);
        $stmt->bindValue(':place', $blogs['place'], PDO::PARAM_STR);
        $stmt->bindValue(':rating', $blogs['rating'], PDO::PARAM_INT);
        $stmt->bindValue(':comment', $blogs['comment'], PDO::PARAM_STR);
        $stmt->bindValue(':air', $blogs['air'], PDO::PARAM_INT);
        $stmt->bindValue(':weather', $blogs['weather'], PDO::PARAM_INT);
        $stmt->execute();
        $dbh->commit();
        echo '已更新留言';
    }catch(PDOException $e){
        $dbh->rollBack();
        exit($e);
    }
}

```

整體寫法都差不多。但更改留言的介面會保留原先輸入的資料,

```

<?php

require_once('blog.php');

$blog = new Blog();
$result = $blog->getById($_GET['id']);

$id = $result['id'];
$user_id = $result['user_id'];
$date = $result['date'];
$city = $result['city'];
$place = $result['place'];
$rating = (int)$result['rating'];
$comment = $result['comment'];
$air = (int)$result['air'];
$weather = (int)$result['weather'];

?>

```

會使用getById依據留言id(網頁上按下刪除的按鈕時會自動從網頁上抓)拿到那則留言的詳細資訊, 再將留言的詳細資料呈現在更改留言的表格上, 使用者就可以直接更動不須重新輸入資料。一樣的, 欄位有缺少, 送出時會顯示少了哪個欄位。

```

public function getById($id){
    if(empty($id)){
        exit('留言ID錯誤');
    }

    // $dbh = dbConnect();
    $dbh = $this->dbConnect();

    $stmt = $dbh->prepare("SELECT * FROM $this->table_name WHERE id = :id");
    $stmt->bindValue(':id', (int)$id, PDO::PARAM_INT);

    $stmt->execute();
    $result = $stmt->fetch(PDO::FETCH_ASSOC);

    if(!$result){
        exit('沒有這則留言');
    }
    return $result;
}

```

getById會下一個query到資料庫，但如果搜不到那則留言時，會顯示錯誤訊息在網頁上。

三、刪除留言：

```

public function delete($id){
    if(empty($id)){
        exit('留言ID錯誤');
    }

    $dbh = $this->dbConnect();

    $stmt = $dbh->prepare("DELETE FROM $this->table_name WHERE id = :id");
    $stmt->bindValue(':id', (int)$id, PDO::PARAM_INT);

    $stmt->execute();
    echo '已刪除留言';
}

```

留言的id(\$id)會在按下刪除按鈕時直接從網頁上自動抓取，不需要由使用者輸入，仍有做除錯的步驟。

四、搜尋留言：

```

<?php
ini_set('display_errors', "On");
require_once('blog.php');
$key = $_POST;
$blog = new Blog();
$blog->placeValidate($key);
$result = $blog->searchByPlace($key['place']);

function h($s){
    return htmlspecialchars($s, ENT_QUOTES, "UTF-8");
}

```

使用者可以輸入地點名稱，搜尋留言，只要留言的景點名稱有包含使用者輸入的關鍵字就會顯示。

```

public function placeValidate($blogs){
    if(empty($blogs['place'])){
        exit('請輸入景點名稱(20字以內)');
    }
    if(mb_strlen($blogs['place']) > 20){
        exit('請輸入景點名稱(20字以內)');
    }
}

```

placeValidate會檢查使用者有輸入資料，且沒有超過20字。

```

public function searchByPlace($key){
    if(empty($key)){
        exit('留言ID錯誤');
    }

    $dbh = $this->dbConnect();
    $sql = "SELECT * FROM $this->table_name WHERE place LIKE CONCAT( '%', :key, '%' )";
    //var_dump($sql);
    $stmt = $dbh->prepare($sql);
    $stmt->bindValue(':key', $key, PDO::PARAM_STR);
    //var_dump($sql);
    $stmt->execute();
    $result = $stmt->fetchall(PDO::FETCH_ASSOC);

    if(!$result){
        exit('沒有相符資料');
    }
    return $result;
    $dbh = null;
}

```

searchByPlace會比對有包含關鍵字的景點。若無景點符合搜尋條件，會顯示錯誤資訊在網頁上。

接下來是由地點條件搜尋景點詳細資訊、天氣資訊、使用者評價的部分。

五、天氣與景點sql

1. 搜尋符合縣市、鄉鎮市區的景點資料。可以提供使用者參考。

```

try {
    $dbh = new PDO($dsn, $user, $pass, [
        PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,
    ]);
    //$sql = "SELECT * FROM attraction WHERE Name LIKE BINARY '%" . $_GET['site'] . "%'";
    //$sql = "SELECT * FROM attraction WHERE Region LIKE BINARY '新竹%'";
    $sql = "SELECT * FROM attraction WHERE Region LIKE BINARY '%" . $_POST['city'] . "%' AND To
    if ($stmt = $dbh->query($sql)) {
        echo '找到' . $stmt->rowCount() . '筆資料<br><br>';
        $result = $stmt->fetchAll();
        foreach ($result as $datainfo) {
            echo $datainfo['name'] . "<br>" . $datainfo['toldescribe'] . "<br>";
            if ($datainfo[12] != NULL) {
                echo "<img src=$datainfo[12] width=400><br>";
            }
            echo "<br>";
        }
    } else {
        echo 'hello<br>';
    }
    $dbh = null;
} catch (PDOException $e) {
    echo ' fail to connect' . $e->getMessage();
    exit();
};

```

```
$sql = "SELECT * FROM attraction WHERE Region LIKE BINARY '%" .
$_POST['city'] . "%' AND Town LIKE BINARY '%" . $_POST['place'] . "%';
```

有用try catch防止query錯誤。

sql有用like的寫法因此有包含關鍵字的都可以搜尋出來。例如要搜尋「信義區」的景點時，輸入「信義」就可以搜尋的到。

2.以景點關鍵字、日期搜尋景點詳細資料(information.php)

```
if (empty($_POST['place'])) {
    echo '資料有缺，請再次填寫<br>';
    exit();
};
```

一樣有try catch, 若使用者沒有輸入關鍵字，會無法搜尋，並顯示資料有缺，請使用者再次輸入。

```
$sql = "SELECT * FROM attraction WHERE Name LIKE BINARY '%" .
$_POST['place'] . "%' OR Keyword LIKE BINARY '%" . $_POST['place'] . "%';
```

首先會先篩出有符合景點關鍵字和景點名稱的所有景點相關資料。(所以可以用名稱也可以用關鍵字進行查詢)

```
$sql1 = "SELECT station.station_id, station.station_name FROM
attraction, station WHERE attraction.Name = '" . $datainfo['name'] . "'"
ORDER BY
SQRT((attraction.Px-station.px)*(attraction.Px-station.px)+(attraction.
Py-station.py)*(attraction.Py-station.py)) limit 2";
```

搜尋和符合搜尋條件的景點最接近的兩個測站(測站id)

```
$sql = "SELECT SUBDATE('" . $_POST['date'] . "', INTERVAL 4 DAY)";
if ($stmt = $dbh->query($sql)) {
    $result = $stmt->fetchAll();
    foreach ($result as $datainfo) {
        $datestr7 = $datainfo[0];
    }
}
```

用4個含ADDDATE的sql找出使用者指定的日期的前4到前1天的日期
然後指定日期的後1到後4天是利用含SUBDATE的sql得出。

接下來分天氣部分分兩個部分，第一部分是雨量有關的部分：

```
SELECT *
FROM dy_report
WHERE station_id = '" . $stationarr[0] . "'"
AND (
    date LIKE "%" . $datestr1 . "%"
    OR date LIKE "%" . $datestr2 . "%"
    OR date LIKE "%" . $datestr3 . "%"
)
AND precipitation != 0
```

這部分如果使用者有選擇日期，則用日期前後三天的資料去算降雨資訊，如果沒有選擇，則統計測站過去全年的降雨機率及平均降雨量。

第二部分是雨量以外的天氣部分：

```

SELECT ROUND(avg(temp), 2),
       ROUND(AVG(humidity), 2),
       ROUND(AVG(wind_speed), 2)
FROM one_year
WHERE (
    date LIKE "%" . $datestr1 . "%:00"
    OR date LIKE "%" . $datestr2 . "%:00"
    OR date LIKE "%" . $datestr3 . "%:00"
    OR date LIKE "%" . $datestr4 . "%:00"
    OR date LIKE "%" . $datestr5 . "%:00"
    OR date LIKE "%" . $datestr6 . "%:00"
    OR date LIKE "%" . $datestr7 . "%:00"
    OR date LIKE "%" . $datestr8 . "%:00"
    OR date LIKE "%" . $datestr9 . "%:00"
)
AND station_id LIKE "%" . $stationarr[0] . "%"
echo "<td>" . round($v1[0] * 0.7 + $v2[0] * 0.3, 2) . "</td>";
echo "<td>" . round($v1[1] * 0.7 + $v2[1] * 0.3, 2) . "</td>";
echo "<td>" . round($v1[2] * 0.7 + $v2[2] * 0.3, 2) . "</td>";

```

用前面搜尋出來的兩個測站，以最近測站為主(佔平均的七成)，第二近測站為輔，計算出當日前後九天的平均氣溫、濕度和風速，因為有可能有的測站沒有資料，所以如果只有一個測站有資料就直接使用那個測站的平均

```

SELECT ROUND(AVG(O.temp), 2) AS avg_temp,
       ROUND(AVG(O.humidity), 2) AS avg_humidity,
       ROUND(AVG(O.wind_speed), 2) AS avg_wind_speed
FROM one_year O,(
    SELECT DISTINCT O.station_id
    FROM one_year O,
        (SELECT DISTINCT S.station_id
        FROM attraction A,station S
        WHERE Name LIKE '%景點關鍵字%' OR Keyword LIKE BINARY '%景點關鍵字%'
        ORDER BY SQRT((A.px - S.px) *(A.px - S.px) +(A.py - S.py) *(A.py - S.py))
        limit 10
    ) AS stations
    WHERE O.station_id = stations.station_id
    limit 1
) AS the_station,
(SELECT *
FROM
    (SELECT DATE_FORMAT(
        (SELECT DATE_ADD('2021-06-08', INTERVAL 1 DAY)), '%m-%d') AS A1) AS A,
    (SELECT DATE_FORMAT(
        (SELECT DATE_ADD('2021-06-08', INTERVAL 2 DAY)), '%m-%d') AS A2) AS B,
    (SELECT DATE_FORMAT(
        (SELECT DATE_ADD('2021-06-08', INTERVAL 3 DAY)), '%m-%d') AS A3) AS C,
    (SELECT DATE_FORMAT(
        (SELECT DATE_ADD('2021-06-08', INTERVAL 4 DAY)), '%m-%d') AS A4) AS D,
    (SELECT DATE_FORMAT(
        (SELECT DATE('2021-06-08')), '%m-%d') AS N) AS E,
    (SELECT DATE_FORMAT(
        (SELECT DATE_SUB('2021-06-08', INTERVAL 1 DAY)), '%m-%d') AS S1) AS F,
    (SELECT DATE_FORMAT(
        (SELECT DATE_SUB('2021-06-08', INTERVAL 2 DAY)), '%m-%d') AS S2) AS G,
    (SELECT DATE_FORMAT(
        (SELECT DATE_SUB('2021-06-08', INTERVAL 3 DAY)), '%m-%d') AS S3) AS H,
    (SELECT DATE_FORMAT(
        (SELECT DATE_SUB('2021-06-08', INTERVAL 4 DAY)), '%m-%d') AS S4) AS I
) AS X

```

```
WHERE O.station_id = the_station.station_id
    AND (O.date LIKE CONCAT('%', X.N, '%')
        OR O.date LIKE CONCAT('%' + X.A1 + '%')
        OR O.date LIKE CONCAT('%', X.A2, '%')
        OR O.date LIKE CONCAT('%', X.A3, '%')
        OR O.date LIKE CONCAT('%', X.A4, '%')
        OR O.date LIKE CONCAT('%', X.S1, '%')
        OR O.date LIKE CONCAT('%', X.S2, '%')
        OR O.date LIKE CONCAT('%', X.S3, '%')
        OR O.date LIKE CONCAT('%', X.S4, '%'))
    AND O.date NOT LIKE '% 00:00:00'
```

這部分是以上操作最初的sql雛形

```
$sql = "SELECT ROUND(avg(rating), 1) FROM board WHERE place =
'$citename';"
```

篩出景點的使用者評分平均分數。

```
$sql = "SELECT *
    FROM (
        SELECT COUNT(*)
        FROM board
        WHERE air = 5 AND place = '$citename') AS a,
        (SELECT COUNT(*)
        FROM board
        WHERE place = '$citename') AS b";
```

篩出使用者覺得景點空氣品質非常好的有幾則留言。

```
$sql = "SELECT *
    FROM (
        SELECT COUNT(*)
        FROM board
        WHERE weather = 3 AND place = '$citename') AS a,
        (SELECT COUNT(*)
        FROM board
        WHERE place = '$citename') AS b";
```

篩出使用者紀錄景點當日是晴天的有幾則留言。

3. 以天氣條件、日期範圍篩選景點，以及景點的使用者評價(search.php)

```
try {
    $dbh = new PDO($dsn, $user, $pass, [
        PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,
    ]);
    $d1 = '';
    $d2 = '';
    if (empty($_POST['date1'])) {
        $d1 = '2020-05-29 01:00';
    }
    $d2 = '2020-06-01 01:00';
} catch (PDOException $e) {
    echo ' fail to connect' . $e->getMessage();
    exit();
};?>
```

有try catch, 資料庫無法連線，或是query有問題時會顯示error message。

使用者評價(評分、天氣、空氣)的寫法和上一部分的寫法一樣。前半部分，是以使用者輸入的

```
SELECT*
FROM (
    SELECT *
    FROM one_year
    WHERE date >= '" . $d1 . "'  

        AND date <= '" . $d2 . "'  

        AND temp >= '" . $_POST['tem1'] . "'  

        AND temp <= '" . $_POST['tem2'] . "'  

        AND humidity >= '" . $_POST['hum1'] . "'  

        AND humidity <= '" . $_POST['hum2'] . "'  

        AND wind_speed LIKE BINARY '%" . $_POST['spe'] . "%'  

        AND precipitation LIKE BINARY '%" . $_POST['pre'] . "%'  

        AND sunshine LIKE BINARY '%" . $_POST['sun'] . "%'
group by station_id) AS A,  

(SELECT s.name, s.toldescribe,s.description, s.tel,s.add,s.opentime,s.travellinginfo,  

 s.ticketinfo,s.remarks,s.picdescribe1,s.picdescribe2,s.picdescribe3, N.station_id  

FROM (
    SELECT N.name, MIN(N.distance) AS m, N.toldescribe,N.description,N.tel,N.add,  

    N.opentime,N.travellinginfo,N.ticketinfo,N.remarks,N.picdescribe1,N.picdescribe2,N.picdescribe3
    FROM (
        SELECT SQRT((attraction.Px-station.px)*(attraction.Px-station.px)+(attraction.Py-station.py)*(attraction.Py-station.py)) AS distance,  

            attraction.name , station.station_name ,station.station_id , attraction.id,attraction.toldescribe,attraction.description,  

            attraction.tel,attraction.add,attraction.opentime,attraction.travellinginfo,attraction.ticketinfo,attraction.remarks,  

            attraction.picdescribe1,attraction.picdescribe2,attraction.picdescribe3
        FROM attraction, station) AS N
    GROUP BY N.name) AS s,  

(SELECT SQRT((attraction.Px-station.px)*(attraction.Px-station.px)+(attraction.Py-station.py)*(attraction.Py-station.py)) AS distance,  

    attraction.name , station.station_name, station.station_id, attraction.id
    FROM attraction, station) AS N
    WHERE N.distance = s.m AND N.name = s.name) AS B
    WHERE A.station_id =B.station_id
LIMIT 10
```

天氣條件以及日期範圍(月、日)篩出符合條件的測站id(station_id)，後半段是做出每個景點以及距離它最近的測站id的table，距離的算法我們是利用歐幾里得距離，再將兩個table做station_id比對，就可以找出符合搜尋條件景點的詳細資料。

參、Application

一、選擇景點搜尋天氣

畫面1：首頁的搜尋功能為旅遊建議。如果不輸入的話會顯示所有資料，考慮景點的數量龐大，可以輸入縣市或鄉鎮市篩選地點。縣市的部分有做成下拉式選單，頁面的下半部分有所以鄉鎮市區名稱可以提供參考。例如可以只用「臺北市」作為搜尋條件，也可以只用「信義」或是「信義區」搜尋。按下「確定」可以看到一些參考的景點簡介。

點選「用天氣篩選觀想去的地區」可進到另一個搜尋功能的畫面。

點選「進入留言板」可進到和留言板功能相關的頁面。

首頁

請選擇要去遊玩的縣市：

輸入想搜尋的城鎮名稱：

[用天氣篩選觀想去的地區](#)

[進入留言板](#)

以下是各縣市城鎮名稱，可參考

南投縣 中寮鄉
南投縣 仁愛鄉
南投縣 信義鄉
南投縣 南投市
南投縣 名間鄉
南投縣 國姓鄉
南投縣 埔里鎮
南投縣 水里鄉
南投縣 竹山鎮
南投縣 草屯鎮
南投縣 集集鎮
南投縣 魚池鄉
嘉義市 東區
嘉義市 西區
嘉義縣 中埔鄉
嘉義縣 六腳鄉

畫面2：當我們輸入縣市、鄉鎮市區作為篩選條件。如：新竹市東區

首頁

請選擇要去遊玩的縣市：

輸入想搜尋的城鎮名稱：

畫面3：以新竹市東區作為篩選條件得到的旅遊建議。下圖上半部的搜尋欄位可讓使用者輸入想去的地點關鍵字，篩尋出來的結果不會受到之前輸入的縣市、鄉鎮市區影響，例如可以輸入「書店」，獲得資料庫中所有景點名稱中有書店的景點。

下圖下半部文字敘述部分即為符合條件的景點，會列出景點名稱、簡介、圖片，若景點無圖片則不顯示。

以下是此地點中的景點

輸入想去的景點名稱：(ex：海、湖、天后宮、書店，或複製貼上下面的景點)

選擇想要旅遊的日期：

 yyyy / 月 / dd

[返回首頁](#)

找到50筆資料

新竹驛(新竹火車站)

1886年台灣省，劉銘傳基於海防、開發與運輸之需要，經於1887年向清廷奏准建設台灣鐵路。1893年基隆-新竹鐵路竣工，新竹票房(車站)設立於枕頭山脚下。明治初期，日鐵道隊有鑿於清朝期間所建造的鐵路蜿蜒蛇行，開始修改舊有的路線，新建的車站稱為新竹停車場，是一處簡易的日本瓦木造車站。1908年新的鐵路通車，新的車站(新竹驛)於1913年完工，即今日所見新竹火車站。新竹火車站位於新竹市中華路上，是全台唯一沒有招牌的火車站，建築樣式融合了西方建築特色，加上陡斜的複折式屋頂與老虎窗，同時在柱體，牆壁轉角與開口部四周等結構應力集中的地方，加強石造的結實感，使得整體建築物呈現繁複卻不失莊重，嚴肅卻又不失親和的建築風味來。設計者的巧思，使火車站成為新竹的門戶與都市的地標。新竹火車站也是台灣現存最古老的火車大站，因地處位置良好，不僅是交通要地，更是新竹市最繁榮的地區，並於民國87年獲列為國定古蹟。

新竹火車站 - 站前廣場

新竹火車站前廣場經107年景觀改造工程，全新人行步道，以及無障礙、人車分流的設計，提升行人安全，拉近市民與廣場的距離，城市門面煥然一新。站前廣場工程打造視覺穿透廣場、寬敞的平面步道不僅安全，輪椅族也能利用平坦人行道，此外，站前廣場是年輕人表演、聚會聊天熱門場地。站前廣場融入「大公園」及「人本」理念，為容納1,500人的步行廣場，讓人、車分流，行人不與車爭道，營造安全、舒適的休憩候車空間，並串聯護城河、府後街街角廣場、文化局綠廊、新竹綠園道、隆恩圳全長4公里的「新竹之森」。

畫面4：若景點有圖片則結果如下。

蘇格貓底二手書咖啡店

蘇格貓底二手書咖啡屋原在臺北師大商圈裡，如今則進駐在清華大學校園內，是一家仰賴餐飲支撐營業開銷的二手書店。店內空間十分雅緻，最令人讚賞的是數量龐大的音樂CD。經營的貓哥以讀者自居，空間裡一定要有書，是他最大的堅持。



畫面5：根據旅遊建議，輸入關鍵字及旅遊日期。下圖以「蘇格拉底二手書咖啡館」為例。

以下是此地點中的景點

輸入想去的景點名稱：(ex：海、湖、天后宮、書店，或複製貼上下面的景點)

選擇想要旅遊的日期：

[返回首頁](#)

畫面6：顯示有幾筆符合的資料及每個景點的詳細資訊，算出指定日期利用過去天氣資訊得出的推測結果與使用者對此景點的評分。過去天氣資訊的部分，由於我們的資料來源，除了降雨量有過去九年的資料之外，其他詳細的氣象觀測資料氣象局只有一年份的。我們用離景點最近的測站過去九年每天的降雨資料算出某日期前後三天的降雨機率及平均降雨量。其他氣象資料我們找出離景點最近的兩個測站某日期前後九天的資料來計算平均，最靠近的測站佔七成、第二近的測站佔三成。如果某個測站不在一年觀測資料裡面，會只用一個測站資料做平均。下圖由於尚未有人對此景點評分，預設為0/0。

景點詳細資訊

找到1筆資料

蘇格貓底二手書咖啡店

蘇格貓底二手書咖啡屋原在臺北師大商圈裡，如今則進駐在清華大學校園內，是一家仰賴餐飲支撐營業開銷的二手書店。店內空間十分雅緻，最令人讚賞的是數量龐大的音樂CD。經營的貓哥以讀者自居，空間裡一定要有書，是他最大的堅持。

電話：886-3-5736096

地址：新竹市 新竹市光復路2段101號

開放時間：週一至六：11:30 - 20:30，週日休

旅遊資訊：公車站「清華大學」站



根據過去資料，以下為此景點和日期的天氣資訊（如無選擇日期，將顯示全年資料）：

降雨機率 平均雨量 平均溫度 平均相對濕度 平均風速 使用者評分 使用者評空氣品質(非常好/全部資料) 使用者評天氣好壞(晴天/全部資料)

0.37 3.33 30.13 69.67 2.99 尚無人評分 0/0 0/0

[返回首頁](#)

畫面7：根據旅遊建議，輸入關鍵字及旅遊日期。下圖以「風櫃嘴觀景台」為例。

以下是此地點中的景點

輸入想去的景點名稱：(ex：海、湖、天后宮、書店，或複製貼上下面的景點)

選擇想要旅遊的日期：

畫面8：顯示有幾筆符合的資料及每個景點的詳細資訊，以及搜尋旅遊日期在過去幾年的天氣資訊與使用者對此景點的評分等。由於先前有人對此景點評分，會顯示最高分評價(空氣非常好、晴天)佔所有評價的比例。

景點詳細資訊

找到1筆資料

風櫃嘴觀景台

風櫃嘴位於萬里、士林、內湖三條產業道路的三叉點上，從萬里經溪底、靈泉寺，往臺北士林方向前行即可到達，是一處停車歇腳的好地方；此處屬陽明山國家公園管轄，有一步道長6.1公里，可上擎天崗。此處因地型關係，形成一風口，以霧大風強而聞名，在此可感受大自然豐碩的能量，可聆聽大自然的訊息，也看得見大自然幻化的景象，進而意會出大自然生命的智慧。

電話：886-2-28613601

地址：新北市萬里區風櫃嘴

開放時間：全年開放

根據過去資料，以下為此景點和日期的天氣資訊（如無選擇日期，將顯示全年資料）：

降雨機率 平均雨量 平均溫度 平均相對濕度 平均風速 使用者評分 使用者評空氣品質(非常好/全部資料) 使用者評天氣好壞(晴天/全部資料)

0.33	6.33	25.07	81.58	1.14	4.8	3/6	2/6
------	------	-------	-------	------	-----	-----	-----

[返回首頁](#)

二、選擇天氣搜尋景點

畫面1：這是第二個搜尋功能，利用指定日期及指定的天氣條件，篩選出符合條件的景點。只有旅遊日期是一定要輸入的，其餘的選項如溫度、濕度、風速、降雨量、日照時間可以留白。

首頁

[返回首頁](#)

請選擇datetime範圍：

yyyy / 月 / dd -- -- : -- ~ yyyy / 月 / dd -- -- : --

請輸入溫度範圍：（也可以不輸入）

~

請輸入濕度範圍：（也可以不輸入）

~

請輸入風速：（也可以不輸入）

請輸入降雨量：（0為無降雨，也可以不輸入）

請輸入日照時間：（也可以不輸入）

畫面2:選擇旅遊日期及期望的天氣狀況。

首頁

[返回首頁](#)

請選擇datetime範圍:

2021/07/01 上午 08:00 ~ 2021/07/02 下午 06:00

請輸入溫度範圍: (也可以不輸入)

25 ~ 30

請輸入濕度範圍: (也可以不輸入)

~

請輸入風速: (也可以不輸入)

~

請輸入降雨量: (0為無降雨，也可以不輸入)

~

請輸入日照時間: (也可以不輸入)

~

畫面3:符合指定日期及指定天氣條件的景點，會顯示景點的詳細資訊及使用者評價。

[返回搜尋](#)

以下是符合天氣中的景點

找到10筆資料

台灣金屬創意館

台灣金屬創意館為第一座金屬造型觀光工廠，來這裡可讓大家看到並觸摸到金屬柔軟的一面，館區內充滿各種金屬的驚奇與創意，是一個極具觀光、休閒、知性與文化的好地方。台灣金屬創意館為志鋼金屬股份有限公司創立之品牌，志鋼金屬是由六位志同道合的好朋友攜手創立，1995年成立之時，即專注於板金鋼鐵加工上，因而將公司命名為志鋼金屬，十多年來，六人同心協力使公司一路成長茁壯。2010年，本著創新、突破、品牌、回饋之思維，決議推動工廠觀光化，將製造業與服務業結合，打造全國第一座板金觀光工廠。2013年正式通過觀光工廠評鑑，成立台灣金屬創意文化館。志鋼金屬為專業板金製造廠，一貫化生產流程讓參觀者可在志鋼看到完整的流程，由雷射切割、自動沖孔機進行金屬板下料，再由電腦折床進行成形作業，接著由焊接部門已氬鋸、CO₂鋸接或自動鋸接系統進行鋸接加工與研磨，之後由塗裝部門進行產品塗裝與印刷，最後由組立部門以手動或氣動工具進行產品組裝與包裝。

電話 : 886-6-2036735#245

地址 : 台南市永康區永科環路598號

開放時間 :

收費方式 : 收費方式請電洽

註記 : 詳細參觀訊息請逕洽廠商網站說明

使用者評分: 4.3

使用者評空氣品質(非常好/全部資料): 2/3

使用者評天氣好壞(晴天/全部資料): 1/3

臺灣菸酒(股)公司林口觀光酒廠

林口酒廠為擁有近百年歷史的台北酒廠(1921)、板橋酒廠(1937)、樹林酒廠(1906)合併而成的新酒廠，於民國76年春正式啟用量產。原三酒廠各具有不同產品與文化特色，合併後發展為北部地區產能最大、設備最現代化的米糧釀造酒生產工廠。林口酒廠成立以來一直是以生產紹興酒為主，民國85年利用原紹興

三、留言板

畫面1：從首頁進入留言板後，會看到所有的留言及留言內容，如下圖所示。心得只顯示前五字，若要閱讀全文可以點及右側的「全文」查看。

留言板

[新增留言](#)

[搜尋留言](#)

[返回首頁](#)

使用者ID	發文日期	日期	縣市	景點名稱	評價	心得	空氣	天氣
440	2021-06-07 16:23:18	2021-06-06	臺北市	風櫃嘴觀景台	5	從萬里經溪	非常好	晴天
101	2021-06-09 01:57:08	2021-06-01	臺北市	風櫃嘴觀景台	4	此處屬陽明	普通	晴天
433	2021-06-07 16:26:01	2021-06-04	臺北市	風櫃嘴觀景台	5	此處因地型	好	雨天
1111	2021-06-07 17:15:36	2021-06-07	桃園市	水牛書店	1	普	普通	多雲
1213	2021-06-07 17:16:12	2021-06-07	臺北市	女書店	5	棒	非常好	晴天
1130	2021-06-07 23:30:59	2021-06-07	臺北市	風櫃嘴觀景台	5	測試用，景	非常好	雨天
440	2021-06-09 01:35:17	2021-06-07	臺南市	台灣金屬創意館	3	台灣金屬創	非常糟	雨天
433	2021-06-09 01:36:28	2021-06-21	臺南市	台灣金屬創意館	5	來這裡可讓	非常好	晴天
433	2021-06-09 01:37:16	2021-06-21	臺南市	台灣金屬創意館	5	館區內充滿	非常好	多雲
440	2021-06-09 01:46:53	2021-06-05	臺北市	風櫃嘴觀景台	5	好玩	非常好	晴天
433	2021-06-09 02:45:38	2021-06-12	臺北市	風櫃嘴觀景台	5	棒	非常糟	雨天

畫面2：按下畫面1中的全文，可以看到詳細的留言內容。

心得全文

發文者ID: 440

發文日期: 2021-06-07 16:23:18

旅遊日期: 2021-06-06

縣市: 臺北市

景點名稱: 風櫃嘴觀景台

評分: 5

心得: 從萬里經溪底、靈泉寺，往臺北士林方向前行即可到達，是一處停車歇腳的好地方

空氣: 非常好

天氣: 晴天

[返回留言板](#)

[返回首頁](#)

畫面3：如果我們在畫面1點擊「編輯」可以更新留言，如我們更新低一則留言的旅遊日期為2021/06/07。

更新留言

使用者ID:

旅遊日期:

縣市:

景點名稱:

評分:(5為最高)

1分 2分 3分 4分 5分

心得:

從萬里經溪底、靈泉寺，往臺北士林方向前行即可到達，是一處停車歇腳的好地方

空氣品質:

非常糟 糟 普通 好 非常好

天氣:

雨天 多雲 晴天

[返回留言板](#)

[返回首頁](#)

畫面4：更新完內容後按下「更新」，會顯示「已更新留言」。

已更新留言

[返回留言板](#)

[返回首頁](#)

畫面5:當我們返回留言板可以看到第一則留言的日期已被更新。

留言板

[新增留言](#)

[搜尋留言](#)

[返回首頁](#)

使用者ID	發文日期	日期	縣市	景點名稱	評價	心得	空氣	天氣
440	2021-06-17 16:57:53	2021-06-07	臺北市	風櫃嘴觀景台	5	從萬里經溪 非常好	晴天	全文 編輯 刪除
101	2021-06-09 01:57:08	2021-06-01	臺北市	風櫃嘴觀景台	4	此處屬陽明 普通	晴天	全文 編輯 刪除
433	2021-06-07 16:26:01	2021-06-04	臺北市	風櫃嘴觀景台	5	此處因地型 好	雨天	全文 編輯 刪除
1111	2021-06-07 17:15:36	2021-06-07	桃園市	水牛書店	1	普 普通	多雲	全文 編輯 刪除
1213	2021-06-07 17:16:12	2021-06-07	臺北市	女書店	5	棒 非常好	晴天	全文 編輯 刪除
1130	2021-06-07 23:30:59	2021-06-07	臺北市	風櫃嘴觀景台	5	測試用，景 非常好	雨天	全文 編輯 刪除
440	2021-06-09 01:35:17	2021-06-07	臺南市	台灣金屬創意館	3	台灣金屬創 非常糟	雨天	全文 編輯 刪除
433	2021-06-09 01:36:28	2021-06-21	臺南市	台灣金屬創意館	5	來這裡可讓 非常好	晴天	全文 編輯 刪除
433	2021-06-09 01:37:16	2021-06-21	臺南市	台灣金屬創意館	5	館區內充滿 非常好	多雲	全文 編輯 刪除
440	2021-06-09 01:46:53	2021-06-05	臺北市	風櫃嘴觀景台	5	好玩 非常好	晴天	全文 編輯 刪除
433	2021-06-09 02:45:38	2021-06-12	臺北市	風櫃嘴觀景台	5	棒 非常糟	雨天	全文 編輯 刪除

畫面6:更新留言時，所有欄位都必須填入值，如果有任何一欄是空的，如下圖。

更新留言

使用者ID:

旅遊日期:

縣市:

景點名稱:

評分:(5為最高)

1分 2分 3分 4分 5分

心得:

從萬里經溪底、靈泉寺，往臺北士林方向前行即可到達，是一處停車歇腳的好地方

空氣品質:

非常糟 糟 普通 好 非常好

天氣:

雨天 多雲 晴天

[返回留言板](#)

[返回首頁](#)

畫面7:如果有任何一欄是空的，會跳出「請輸入使用者ID」。

| 請輸入使用者ID

畫面8:在留言板我們也可以點擊「刪除」來刪除留言，如刪除第四則景點名稱為水牛書店的留言。

留言板

[新增留言](#)

[搜尋留言](#)

[返回首頁](#)

使用者ID	發文日期	日期	縣市	景點名稱	評價	心得	空氣	天氣
440	2021-06-17 16:57:53	2021-06-07	臺北市	風櫃嘴觀景台	5	從萬里經溪	非常好	晴天
101	2021-06-09 01:57:08	2021-06-01	臺北市	風櫃嘴觀景台	4	此處屬陽明	普通	晴天
433	2021-06-07 16:26:01	2021-06-04	臺北市	風櫃嘴觀景台	5	此處因地型	好	雨天
1111	2021-06-07 17:15:36	2021-06-07	桃園市	水牛書店	1	普	普通	多雲
1213	2021-06-07 17:16:12	2021-06-07	臺北市	女書店	5	棒	非常好	晴天
1130	2021-06-07 23:30:59	2021-06-07	臺北市	風櫃嘴觀景台	5	測試用，景	非常好	雨天
440	2021-06-09 01:35:17	2021-06-07	臺南市	台灣金屬創意館	3	台灣金屬創	非常糟	雨天
433	2021-06-09 01:36:28	2021-06-21	臺南市	台灣金屬創意館	5	來這裡可讓	非常好	晴天
433	2021-06-09 01:37:16	2021-06-21	臺南市	台灣金屬創意館	5	館區內充滿	非常好	多雲
440	2021-06-09 01:46:53	2021-06-05	臺北市	風櫃嘴觀景台	5	好玩	非常好	晴天
433	2021-06-09 02:45:38	2021-06-12	臺北市	風櫃嘴觀景台	5	棒	非常糟	雨天

畫面9:按下「刪除」後會顯示「已刪除留言」。

| 已刪除留言

[返回留言板](#)

[返回首頁](#)

畫面10:如圖所示，景點名稱為水牛書店的留言已被刪除

留言板

[新增留言](#)

[搜尋留言](#)

[返回首頁](#)

使用者ID	發文日期	日期	縣市	景點名稱	評價	心得	空氣	天氣
440	2021-06-17 16:57:53	2021-06-07	臺北市	風櫃嘴觀景台	5	從萬里經溪	非常好	晴天
101	2021-06-09 01:57:08	2021-06-01	臺北市	風櫃嘴觀景台	4	此處屬陽明	普通	晴天
433	2021-06-07 16:26:01	2021-06-04	臺北市	風櫃嘴觀景台	5	此處因地型	好	雨天
1213	2021-06-07 17:16:12	2021-06-07	臺北市	女書店	5	棒	非常好	晴天
1130	2021-06-07 23:30:59	2021-06-07	臺北市	風櫃嘴觀景台	5	測試用，景	非常好	雨天
440	2021-06-09 01:35:17	2021-06-07	臺南市	台灣金屬創意館	3	台灣金屬創	非常糟	雨天
433	2021-06-09 01:36:28	2021-06-21	臺南市	台灣金屬創意館	5	來這裡可讓	非常好	晴天
433	2021-06-09 01:37:16	2021-06-21	臺南市	台灣金屬創意館	5	館區內充滿	非常好	多雲
440	2021-06-09 01:46:53	2021-06-05	臺北市	風櫃嘴觀景台	5	好玩	非常好	晴天
433	2021-06-09 02:45:38	2021-06-12	臺北市	風櫃嘴觀景台	5	棒	非常糟	雨天

畫面11：從留言板頁面中點選「新增留言」會跳出景點名稱搜尋，要輸入景點的全名，以確保留言的景點名稱存在於資料庫中。

搜尋資料庫中有無景點名稱

景點名稱:

[返回留言板](#)

[返回首頁](#)

畫面12：如果只輸入部分名稱，如書店。

搜尋資料庫中有無景點名稱

景點名稱:

[返回留言板](#)

[返回首頁](#)

畫面13：只輸入部分名稱會顯示「景點資料庫中無相符資料」。

景點資料庫中無相符資料

畫面14：如果輸入的景點名稱存在資料庫中，我們就可以新增留言。

搜尋資料庫中有無景點名稱

景點名稱:

[返回留言板](#)

[返回首頁](#)

畫面15:新增留言時欄位不可為空白。

新增留言

使用者ID:

旅遊日期:

 yyyy / 月 / dd

評分:(5為最高)

1分 2分 3分 4分 5分

心得:

空氣品質:

非常糟 糟 普通 好 非常好

天氣:

雨天 多雲 晴天

[返回留言板](#)

[返回首頁](#)

畫面16:若新增留言時有心得為空白, 會顯示「請輸入心得」。

請輸入心得

畫面17:須補齊欄位後才可新增。

已新增心得

[返回留言板](#)

[返回首頁](#)

畫面18:在留言板頁面我們也可以搜尋留言，輸入欲搜尋的地點名稱。

搜尋留言

地點名稱:

[返回留言板](#)

[返回首頁](#)

畫面19:如果景點名稱沒有相符的，會顯示「沒有相符資料」。

沒有相符資料

畫面20:有相符的景點名稱時。

搜尋留言

地點名稱:

[返回留言板](#)

[返回首頁](#)

畫面21:會顯示所有相同景點名稱的留言。

依地點關鍵字搜尋

使用者ID	發文日期	日期	縣市	景點名稱	評價	心得	空氣	天氣
440	2021-06-07 16:23:18	2021-06-06	臺北市	風櫃嘴觀景台	5	從萬里經溪 非常好	晴天	全文 編輯 刪除
101	2021-06-09 01:57:08	2021-06-01	臺北市	風櫃嘴觀景台	4	此處屬陽明 普通	晴天	全文 編輯 刪除
433	2021-06-07 16:26:01	2021-06-04	臺北市	風櫃嘴觀景台	5	此處因地型 好	雨天	全文 編輯 刪除
1130	2021-06-07 23:30:59	2021-06-07	臺北市	風櫃嘴觀景台	5	測試用，景	非常好	雨天 全文 編輯 刪除
440	2021-06-09 01:46:53	2021-06-05	臺北市	風櫃嘴觀景台	5	好玩	非常好	晴天 全文 編輯 刪除
433	2021-06-09 02:45:38	2021-06-12	臺北市	風櫃嘴觀景台	5	棒	非常糟	雨天 全文 編輯 刪除

[返回搜尋留言頁面](#)

[返回留言板](#)

[返回首頁](#)

肆、Others

我們沒有規劃每周進度，因為變化永遠趕不上計劃，大家的課業也都很忙，不如確定有時間時在約討論。

4/17 寫proposal

4/20 第二次寫proposal(因為proposal有缺漏)

5/6 第一次寫project(發現用ubuntu匯入資料時會出現很多問題，試著架網站)

5/26 線上寫project(查詢要用哪種語言寫前端、後段，以及使用的程式)(鄧、呂)

5/28 線上寫project(insert資料、php完成連接資料庫部分)

5/29 線上寫project(insert資料、寫留言板功能)

5/30 寫留言板(鄧)

5/31 完成留言板(鄧)、處理完9年雨量資料和一年天氣資料匯入(這部分花很久時間)

、寫景點搜尋的部分(謝)

6/1 線上討論

6/2 寫景點搜尋天氣的部分(謝)

6/3 第一個sql模板(呂)、約定星期日(6/6)完成

6/5 把sql模板與自己寫的部分融合(謝)

6/6 景點、日期搜尋詳細資料、天氣部分完成(謝)

6/7 寫出天氣搜尋測站網頁(周)、整合不同人網頁(鄧)

6/8 處理網頁還未完成部分、準備簡報、影片

6/9 測試簡報

6/17 寫報告(鄧、呂、謝)

遇到的問題：

在插入資料時，花了非常久的時間，一開始和之前的功課一樣，用ubuntu輸入資料表，但是花一個晚上只能解決一個table，還會發生兩人做法一模一樣，但其中一人會出現error，後來我們放棄用ubuntu寫報告。後來是改用XAMPP完成這次的project。插入資料時，我們有遇到格式不相容無法直接上傳、上傳後資料0行、上傳到一半突然終止等問題。

一、工作分配：

鄧婕妮：分配工作、約討論時間、留言板功能、網頁彙整

周孟謙：連結資料庫，解決資料表上傳bug，用天氣搜尋地點功能

謝曼蓉：對政府網站上的天氣資料進行處理，做景點查詢天氣功能

呂蕊瑄：處裡資料匯入部分及預處理，旅遊日天氣之sql，剪輯影片、簡報製作

助教好，我(鄧婕妮)有幾件事想說明，我們這組有一位組員(周)從頭到尾都沒有在狀況內，例如大家線上討論時，已經說了新增資料表時遇到哪些error時該如何處理，也有寫在hackMD上，但大家結束討論後，半夜又傳訊息來問該如何處理。線上討論時，常常突然說他剛剛都沒聽到聲音、或是正在使用的電腦沒麥克風。(補：還有一開始說要討論架設網頁時都說壓力很大不想討論)6/3就約好星期日完成，但星期日晚上問他寫完沒，他才說他還沒，6/7才上傳他負責的部分，但他寫出的是用天氣搜尋出符合條件的「測站」，和一開始講好的搜尋結果要是景點資料完全不同，我和他解釋了很久他還是沒聽懂，導致大家到了要報告的前天還在熬夜趕project，報告要用的簡報也沒幫忙準備，甚至報告當日人根本不在線上，我幫他報告了他的部分。6/10日他上傳這份報告，並寫「大部分的都完成了」，但今天我們打開時，卻是把我們hackmd、簡報講稿貼上來、當成簡答題在寫，所以剛剛是全部重做。hackmd上9.9成的資料是我們其他三人整理的。還有很多的問題，無法在此詳述。如需求證可以連絡我們其他三個組員(鄧、呂、謝)。謝謝。

二、GitHub連結

https://github.com/sylviacat1222/database_project

三、Hackmd連結

<https://hackmd.io/XbJ6oUjMRIOvdbVeLbt1A?view>