

# Introduction to Database Systems

## Final Project Report

### 第 4 組

110652021 龔大承

110652042 賴宥瑜

110652034 賴羿茗

- Main idea

我們的設計理念是設計一個列出所有股票、股價分析圖、各種指數資訊的網頁，讓想要踏入股市的股票初學者可以自由的篩選想要的股票。

希望大家能在投入金錢之前，先透過觀察股價趨勢圖學會各種 K 線代表的意義，並利用資料庫中的資料預測並自行核對股票漲跌。

- Data

### Introduction to the data:

我們的資料是來自臺灣證券交易所官方網站中的每日收盤行情，利用程式從所有資料中篩選出一般股票的大盤統計資訊、寶島股價指數、寶島報酬指數、每日收盤行情的所有股價資訊。

我們的網站透過呼叫 Python 進行爬蟲，每日收盤後便會抓取當天資料，並匯入資料庫中。

### Source API:

[https://www.twse.com.tw/exchangeReport/MI\\_INDEX?response=json&type=ALLBUT0999](https://www.twse.com.tw/exchangeReport/MI_INDEX?response=json&type=ALLBUT0999)

### Tables and columns information:

close\_quotation （股價收盤資訊）

primary key: (date, stock\_no)

欄位名稱	資料型態	代表意義
Date	datetime	資料日期
stock_no	varchar(6)	股票代號
stock_name	varchar(10)	發行公司
tot_volume	bigint	單日總交易股數
tot_num	bigint	單日總交易筆數
tot_money	bigint	單日總交易金額
open_price	float	開盤價
max_price	float	當日最高價
min_price	float	當日最低價
close_price	float	收盤價
up_down	char(1)	漲跌（以+,-表示）
up_diff	float	漲跌幅度（開盤價-收盤價）

market\_statistics (大盤統計資訊)

primary key: date

欄位名稱	資料型態	代表意義
date	datetime	資料日期
tot_money	bigint	單日總交易金額(整體市場)
tot_volume	bigint	單日總交易股數(整體市場)
tot_num	bigint	單日總交易筆數(整體市場)

T\_price (價格指數)

primary key: date

欄位名稱	資料型態	代表意義
date	datetime	資料日期
close_point	float	當日的股票價格指數
up_down	char(1)	漲跌 (以+,-表示)
up_point	float	漲跌幅度 (與昨日比)
up_per	float	漲跌百分比

T\_remuneration (報酬指數)

primary key: date

欄位名稱	資料型態	代表意義
date	datetime	資料日期
close_point	float	當日的股票報酬指數
up_down	char(1)	漲跌 (以+,-表示)
up_point	float	漲跌幅度 (與昨日比)
up_per	float	漲跌百分比

stock (股票代號與對應公司) (解決中文顯示問題而設)

primary key: stock\_no , default charset: utf8mb4

欄位名稱	資料型態	代表意義
stock_no	varchar(6)	股票代號
stock_name	varchar(10)	發行公司

account (紀錄帳號與對應的密碼)

primary key: Account

欄位名稱	資料型態	代表意義
Account	varchar(15)	帳號
Password	varchar(15)	密碼

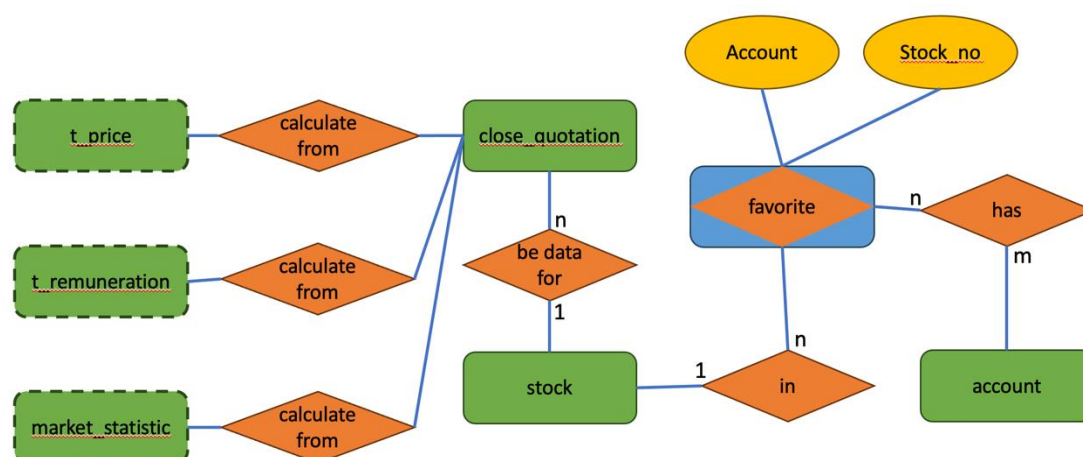
favorite (紀錄帳號的最愛股票代號)

primary key (Account, stock\_no)

foreign key (Account) references account(Account)

欄位名稱	資料型態	代表意義
Account	varchar(15)	帳號
stock_no	varchar(6)	股票代號

ER model:

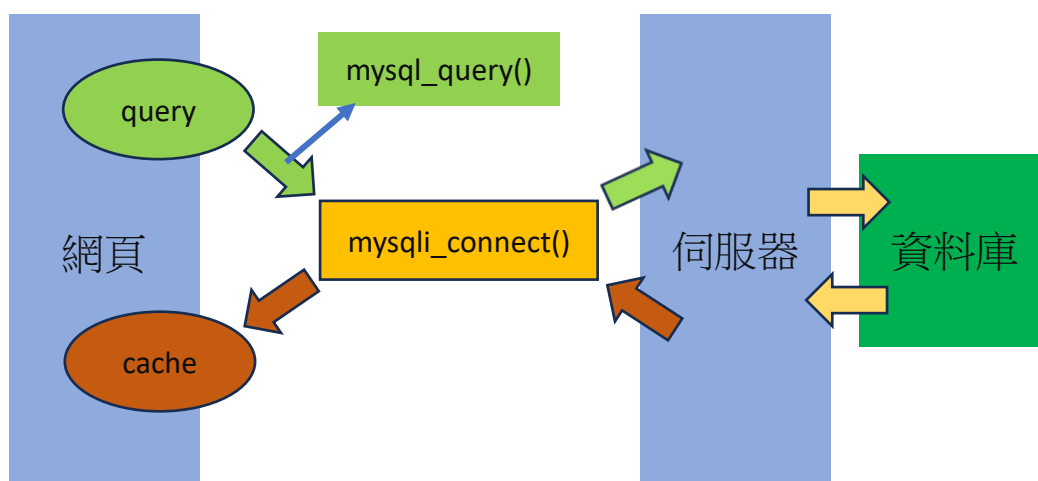


## ● Database

我們使用 MySQL 作為此 project 的 database，並且寫了一個 Python 程式搭配 Windows 排程器每天自動擷取、更新資料。而網頁端的資料則是直接用 php 連結資料庫以更新資料。

how we connect our database to the application:

我們使用了 php 提供的函數 `mysqli_connect()` 從網頁端連結資料庫，並使用 `mysqli_query()` 來執行 query 以取得網頁端需要的資料。



在主頁面中網頁會在使用者按下查詢按鈕後，對 selectbox 選擇的資料集和 textbox 中填入的數值進行分析以建構 where clause 與 from 語句，最後再與 checkbox 勾選的 column 組合出需要查詢的 query。

```
foreach ($selected_columns as $column) {  
    if (isset($_POST["column_text"][$column]))  
        $column_text = $_POST["column_text"][$column];  
    else  
        $column_text = "";  
    if (isset($_POST["column_min"][$column]))  
        $min_text = $_POST["column_min"][$column];  
    else  
        $min_text = "";  
    if (isset($_POST["column_max"][$column]))  
        $max_text = $_POST["column_max"][$column];  
    else  
        $max_text = "";
```

上圖的 code 是將 selectbox 中填入的數值存下來的部分。

```
if ($column_text != '') {  
    if (empty($whereClause)) {  
        $whereClause = " WHERE ";  
    } else {  
        $whereClause .= " AND ";  
    }  
    $whereClause .= $column . " = '" . mysqli_real_escape_string($link, $column_text) . "'";  
} elseif ($min_text != "" && $max_text != "") {  
    if (empty($whereClause)) {  
        $whereClause = " WHERE ";  
    } else {  
        $whereClause .= " AND ";  
    }  
    $whereClause .= $column . " BETWEEN '" . mysqli_real_escape_string($link, $min_text) .  
        "' AND '" . mysqli_real_escape_string($link, $max_text) . "'";  
}
```

上圖的 code 是構造 where clause 的部分。

先判斷一個 column 是否有指定查詢的值：

如果有，則判斷現在的 where clause 是否為空的：

如果是空的則在開頭加入 WHERE。

反之，在 where clause 後加上 AND。

接著再加上指定查詢的條件。

如果沒有，則進入下一個判斷。

檢查一個 column 是否有指定查詢的上下界：

如果有，則判斷現在的 where clause 是否為空的：

如果是空的則在開頭加入 WHERE。

反之，在 where clause 後加上 AND。

接著再加上指定查詢的上下界(BETWEEN 語句)。

```

$query = "SELECT";
if (isset($_POST["selected_columns"])) {
    $selected_columns = $_POST["selected_columns"];
    $first = true;
    foreach ($selected_columns as $column) {
        if ($first) {
            $query .= " ".$column;
            $first = false;
        } else {
            $query .= ", ".$column;
        }
    }
    if ($table == "close_quotation")
        $query .= ", stock_no";
} else {
    $selected_columns = $columns;
    $query .= " * ";
}

```

上圖的 code 是構造要查詢的 column 的部分。

如果有指定顯示的 column，則將其放到 SELECT 後；如果沒有指定顯示的 column，則再 SELECT 後加入「\*」以查詢所有 column。

```

$query .= $whereClause;
$table = $_POST["table"];
if ($table === "close_quotation") {
    $query .= " ORDER BY stock_no";
}
if (empty($_POST["limit"])) {
    $_POST["limit"] = 100;
}
$query .= " LIMIT ".$_POST["limit"];
$result = mysqli_query($link, $query);

```

上圖的 code 是組合整個 query 並將查詢 query 的結果存下來的部分。

在主頁面中還有實時更新的「關注的股票」，網頁會在使用者按下加入或刪除按鈕後將對應的股票代碼與帳號的這筆資料從資料庫加入或刪除。

```
// Process adding to favorites
if (isset($_POST['add_to_favorite'])) {
    $stock_no = mysqli_real_escape_string($link, $_POST['stock_no']);
    $account = $_SESSION["name"]; // Replace with the desired account number
    $query = "INSERT INTO favorite (Account, stock_no) VALUES ('$account', '$stock_no')";
    if (mysqli_query($link, $query)) {
        echo " ";
    } else {
        echo "Error adding stock number to favorites: " . mysqli_error($link);
    }
}
```

上圖的 code 是將股票代碼加入此帳號所關注股票的資料中的部分。

```
$Account = $_SESSION["name"]; // Replace with your account value
// Process delete from favorites
if (isset($_POST['remove_from_favorite'])) {
    $stockNo = mysqli_real_escape_string($link, $_POST['stock_no']);
    $deleteQuery = "DELETE FROM favorite WHERE Account = '$Account' AND stock_no = '$stockNo' ";
    $deleteResult = mysqli_query($link, $deleteQuery);
    if ($deleteResult) {
        echo "";
    } else {
        echo "<p>Error removing data from favorites.</p>";
    }
}
```

上圖的 code 是將股票代碼從此帳號所關注股票的資料中刪除的部分。

```
$query = "SELECT cq.*, f.Account, s.stock_name FROM close_quotation cq
INNER JOIN favorite f ON cq.stock_no = f.stock_no
INNER JOIN stock s ON s.stock_no = f.stock_no
WHERE f.Account = '$Account' ORDER BY cq.stock_no";
$result = mysqli_query($link, $query);
```

上圖的 code 是查詢此帳號關注的股票的資料並存下來的部分。

在登入頁面中網頁會在使用者按下登入按鈕後根據填入的帳號向資料庫查詢此帳號的密碼：

```
SELECT Password FROM account WHERE Account='帳號';
```

```
// 檢查是否輸入使用者名稱和密碼
if ($name != "" && $password != "") {
    // 建立MySQL的資料庫連接
    $link = mysqli_connect("127.0.0.1", "root", "googleking110652042")
        or die("無法開啟MySQL資料庫連接!<br/>");
    $db = mysqli_select_db($link, "finalproject");
    // 送出UTF8編碼的MySQL指令
    mysqli_query($link, 'SET NAMES utf8');
    // 建立SQL指令字串
    $sql = "SELECT * FROM account WHERE Password='";
    $sql.= $password.'" AND Account="'. $name.'"';
    // 執行SQL查詢
    $result = mysqli_query($link, $sql);
    $total_records = mysqli_num_rows($result);
    // 是否有查詢到使用者記錄
    if ( $total_records > 0 ) {
        // 成功登入, 指定Session變數
        $_SESSION["login_session"] = true;
        $_SESSION["name"] = $_POST["name"];
        $_SESSION["password"] = $_POST["password"];
        header("Location: index.php");
    } else { // 登入失敗
        echo "<center><font color='red'>";
        echo "Account或password錯誤!<br/>";
        echo "</font>";
        $_SESSION["login_session"] = false;
    }
    mysqli_query($link, "commit;");
    mysqli_close($link); // 關閉資料庫連接
}
```



在修改密碼頁面中網頁會根據已登入的帳號和填入的新密碼對資料庫進行 update：

**UPDATE account SET Password=`新密碼` WHERE Account=`已登入的帳號`;**

```
//檢查是否輸入密碼
if ($password != "") {
    // 建立MySQL的資料庫連接
    $link = mysqli_connect("127.0.0.1","root", "googleking110652042")
    or die("無法開啟MySQL資料庫連接!<br/>");
    $db = mysqli_select_db($link, "finalproject");
    // 送出UTF8編碼的MySQL指令
    mysqli_query($link, 'SET NAMES utf8');
    // 建立SQL指令字串
    $sql = "UPDATE account SET Password='".$password."' WHERE Account='".$name."'";
    // 執行SQL查詢
    $result = mysqli_query($link, $sql);
    if ($result) {
        echo "<center><font color='red'>";
        echo "修改成功!<br/>";
        echo "</font>";
        $_SESSION["modified"] = true;
        unset($_SESSION["login_session"]);
        header("Location: index.php");
    } else { // 修改失敗
        echo "<center><font color='red'>";
        echo "密碼修改失敗!<br/>";
        echo "</font>";
    }
    mysqli_query($link, "commit;");
    mysqli_close($link);
}
```

在註冊頁面中網頁會先查詢註冊的帳號是否已經存在了，如果不存在則將註冊的這組帳號和密碼加入資料庫，反之則告訴使用者帳號已被註冊過了：

```
// 檢查是否輸入帳號和密碼
if ($name != "" && $password != "") {
    // 建立MySQL的資料庫連接
    $link = mysqli_connect("127.0.0.1","root", "googleking110652042")
    or die("無法開啟MySQL資料庫連接!<br/>");
    $db = mysqli_select_db($link, "finalproject");
    // 送出UTF8編碼的MySQL指令
    mysqli_query($link, 'SET NAMES utf8');
    // 建立SQL指令字串
    $sql = "SELECT Account FROM account WHERE Account='".$name."'";
    // 執行SQL查詢
    $result = mysqli_query($link, $sql);
    $total_records = mysqli_num_rows($result);
    // 是否有查詢到使用者記錄
    if ( $total_records == 0 ) {
        // 帳號未註冊過,註冊帳號
        $sql = "INSERT INTO account VALUES('".$name."', '".$password."')";
        mysqli_query($link, $sql);
        echo "<center><font color='red'>";
        echo "註冊成功!<br/>";
        echo "</font>";
        $_SESSION["registered"] = true;
        unset($_SESSION["login_session"]);
        header("Location: index.php");
    } else { // 註冊失敗
        echo "<center><font color='red'>";
        echo "使用者名稱已註冊過!<br/>";
        echo "</font>";
    }
    mysqli_query($link, "commit;");
    mysqli_close($link);
}
```

## Exception:

我們只有做很簡單的 exception，如果有 error 發生就告訴使用者這項操作失敗了，並且不做出其他改變或更新。

# ● Application

## Interface:

我們的應用最主要包含幾個網頁，其中有：

1. main.php:用來做資料的查詢以及圖表的繪製
2. index.php:所有網頁的進入點，可以跳轉到 login.php 或 main.php(不會顯示出來)
3. login.php:使用者登入的網頁
4. register.php:讓使用者註冊的網頁
5. modify\_password:讓使用者更改密碼的網頁

### 1. main.php:

Table close quotation change

☐ date Find for date Min

Max

☐ tot\_volume Find for tot\_volume Min

Max

☐ open\_price Find for open\_price Min

Max

☐ close\_price Find for close\_price Min

Max

select

rows per page:

☐ stock\_no Find for stock\_no Min

Max

☐ tot\_num Find for tot\_num Min

Max

☐ max\_price Find for max\_price Min

Max

☐ up\_down Find for up\_down Min

Max

☐ stock\_name Find for stock\_name Min

Max

☐ tot\_money Find for tot\_money Min

Max

☐ min\_price Find for min\_price Min

Max

☐ up\_diff Find for up\_diff Min

Max

Account option

最上方可以選擇要顯示哪個 table 中的資訊，接下來可以選擇要顯示的 column 即設定搜尋的條件

stock_no	date	close_price	max_price	min_price	open_price	tot_money	tot_num	tot_volume	up_diff	up_down	Favorite
0050	2023-06-06 00:00:00	126.7	127.25	126.15	126.35	872451559	8845	6877372	0.35	+	<button>Add to Favorites</button>
0051	2023-06-06 00:00:00	63.5	63.55	63.05	63.25	2934793	201	46327	0.2	-	<button>Add to Favorites</button>
0052	2023-06-06 00:00:00	111.6	111.9	111	111	18562462	338	166544	0.45	+	<button>Add to Favorites</button>
0053	2023-06-06 00:00:00	64.3	64.65	64.2	64.65	448561	896	6985	0.25	+	<button>Add to Favorites</button>
0055	2023-06-06 00:00:00	23.25	23.4	23.24	23.31	11298966	456	485082	0.06	-	<button>Add to Favorites</button>
0056	2023-06-06 00:00:00	31.15	31.21	31.01	31.1	804246585	17864	25845612	0.05	+	<button>Add to Favorites</button>
0057	2023-06-06 00:00:00	92.15	92.15	91.85	91.9	2565722	869	27938	0.25	+	<button>Add to Favorites</button>

接下來的區塊則會顯示根據搜尋條件所列出的 query 結果，可以選擇是否要將股票增加到 favorite 中

Favorite Stock												
date	stock_no	stock_name	tot_volume	tot_num	tot_money	open_price	max_price	min_price	close_price	up_down	up_diff	Actions
2023-06-06 00:00:00	0051		46327	201	2934793	63.25	63.55	63.05	63.5	-	0.2	<button>Remove</button>
2023-06-06 00:00:00	0053		6985	896	448561	64.65	64.65	64.2	64.3	+	0.25	<button>Remove</button>
2023-06-06 00:00:00	00673R		23431750	1943	214792387	9.15	9.2	9.14	9.17	+	0.09	<button>Remove</button>

這段則是使用者關注的股票의 相關資訊，可以在這邊移除 favorite 的股票



網頁的最下方則是搜尋特定股票的走勢圖

2. **index.php**:這個網頁是作為進入點並判斷是否已經登入來作為使用，所以並沒有 **html** 的頁面，不過會根據當前的登入狀態來決定要跳轉到的頁面(login 或 main)
3. **login.php**:

這個頁面是讓使用者登入的地方，可以輸入帳號密碼或點擊註冊的連結去註冊新的帳號，登入成功後則會跳到 **main.php**

4. **register.php**:

這邊則是讓使用者註冊帳號的頁面，註冊成功就會跳轉到 **login.php**，讓使用者用註冊好的帳戶重新登入

5. **modify\_password.php**:

這是讓使用者更新密碼的頁面，因為帳號會從 **session** 中取得所以不需要重新輸入帳號，更改密碼成功後也會跳回 **login.php** 重新登入

## Functions:

### 1. 顯示某個 table 中滿足特定條件的資訊:

利用 form 中 post 的方式取得 dropdown list 中被選擇的 table，從 `$_POST["table"]` 中去得 table 名稱放在 query 的尾端。

```
$query = "SELECT COLUMN_NAME
FROM INFORMATION_SCHEMA.COLUMNS
WHERE TABLE_SCHEMA = 'finalproject'
AND TABLE_NAME = '". $table.'" ORDER BY ordinal_position;";
```

接下來利用 mysql 中的 information\_schema 取出被選擇的 table 的所有 column 並且將結果存入一個 array 中

```
foreach ($columns as $column) {
    $formattedColumn = str_pad($column, 10, ' ', STR_PAD_LEFT);
    echo '<div>';
    echo '<input type="checkbox" name="selected_columns[]" value="'. $column.'">';
    echo '<label>'. $formattedColumn. '</label>';
    echo '<input type="text" name="column_text['. $column. ']" placeholder="Find for '. $column.'">';
    echo '<input type="text" name="column_min['. $column. ']" placeholder="Min">';
    echo '<input type="text" name="column_max['. $column. ']" placeholder="Max">';
    echo '</div>';
}
```

列印出所有 column 並且有四個 input，分別是選擇是否顯示、找尋等於某個 value、找尋值在某兩個 value 中的輸入格，並且用四個 array 來儲存輸入的結果。

```
$query = "SELECT";
if (isset($_POST["selected_columns"])) {
    $selected_columns = $_POST["selected_columns"];
    $first = true;
    foreach ($selected_columns as $column) {
        if ($first) {
            $query .= " " . $column;
            $first = false;
        } else {
            $query .= ", " . $column;
        }
    }
} else {
    $selected_columns = $columns;
    $query .= " *";
}
$query .= " FROM " . $table;
```

處理 select 語句中需要顯示出來的 column，若沒有選擇的 column 則選擇全部的 columns

```
$whereClause = '';
foreach ($selected_columns as $column) {
    $column_text = $_POST["column_text"][$column];
    $min_text = $_POST["min_text"][$column];
    $max_text = $_POST["max_text"][$column];
    if ($column_text != '') {
        if (empty($whereClause)) {
            $whereClause = " WHERE ";
        } else {
            $whereClause .= " AND ";
        }
        $whereClause .= $column . " = '" . mysqli_real_escape_string($link, $column_text) . "'";
    } elseif ($min_text != '' && $max_text != '') {
        if (empty($whereClause)) {
            $whereClause = " WHERE ";
        } else {
            $whereClause .= " AND ";
        }
        $whereClause .= $column . " BETWEEN '" . mysqli_real_escape_string($link, $min_text) . "' AND '" . mysqli_real_e
    }
}
$query .= $whereClause;
$table = $_POST["table"];
if ($table === "close_quotation") {
    $query .= " ORDER BY stock_no";
}
if (empty($_POST["limit"])) {
    $_POST["limit"] = 100;
}
$query .= " LIMIT " . $_POST["limit"];
```

利用\$\_POST[]中的輸入來決定 where 中要放的條件，若輸入為空則不加入這個條件，最後補上使用人數入的 rows per page 設為 limit

```
foreach ($selected_columns as $column) {
    echo "<td style='border: 1px solid black; padding: 8px;'>" . $row[$column] . "</td>";
}
// add favorite button
if ($table === "close_quotation") {
    echo "<td style='border: 1px solid black; padding: 8px;'>";
    echo "<form method='post'>";
    echo "<input type='hidden' name='stock_no' value='" . $row['stock_no'] . "'>";
    echo "<input type='hidden' name='table' value='" . $table . "'>";
    echo "<button type='submit' name='add_to_favorite'>Add to Favorites</button>";
    echo "</form>";
    echo "</td>";
}
```

印出資料並利用一個 button 讓使用者選擇是否要將這筆資料加到 favorite 中

EX:

若我們打勾及輸入的內容是如下：

Table <span>close quotation</span> <span>change</span>				Account option	
<input checked="" type="checkbox"/> date	Find for date	<input type="text"/>	Min	<input checked="" type="checkbox"/> stock_no	Find for stock_no
<input type="checkbox"/> Max				<input type="text"/>	Min
<input checked="" type="checkbox"/> tot_volume	Find for tot_volume	<input type="text"/>	Min	<input checked="" type="checkbox"/> tot_num	Find for tot_num
<input type="checkbox"/> Max				<input type="text"/>	Min
<input type="checkbox"/> open_price	Find for open_price	<input type="text"/>	Min	<input type="checkbox"/> max_price	Find for max_price
<input type="checkbox"/> Max				<input type="text"/>	Min
<input type="checkbox"/> close_price	Find for close_price	<input type="text"/>	Min	<input type="checkbox"/> up_down	Find for up_down
<input type="checkbox"/> Max				<input type="text"/>	Min
<input type="button" value="select"/>					
rows per page: 20					

則輸出的內容會是下面的表格：

date	stock_no	tot_volume	tot_num	tot_money	Favorite
2023-06-06 00:00:00	0050	6877372	8845	872451559	<input type="button" value="Add to Favorites"/>
2023-06-06 00:00:00	0051	46327	201	2934793	<input type="button" value="Add to Favorites"/>
2023-06-06 00:00:00	0052	166544	338	18562462	<input type="button" value="Add to Favorites"/>
2023-06-06 00:00:00	0053	6985	896	448561	<input type="button" value="Add to Favorites"/>
2023-06-06 00:00:00	0055	485082	456	11298966	<input type="button" value="Add to Favorites"/>
2023-06-06 00:00:00	0056	25845612	17864	804246585	<input type="button" value="Add to Favorites"/>
2023-06-06 00:00:00	0057	27938	869	2565722	<input type="button" value="Add to Favorites"/>

```
// Process adding to favorites
if (isset($_POST['add_to_favorite'])) {
    $stock_no = mysqli_real_escape_string($link, $_POST['stock_no']);
    $account = $_SESSION["name"]; // Replace with the desired account number
    $query = "INSERT INTO favorite (Account, stock_no) VALUES ('$account', '$stock_no')";
    if (mysqli_query($link, $query)) {
        echo " ";
    } else {
        echo "Error adding stock number to favorites: " . mysqli_error($link);
    }
}
?>
```

利用 insert 的功能將被選擇的 stock\_no 加入使用者 account 的 favorite 中，利用這個功能可以讓使用者在不同地方都可以登入並且查看自己的 favorite

```
$query = "SELECT cq.*, f.Account FROM close_quotation cq
          INNER JOIN favorite f ON cq.stock_no = f.stock_no
          WHERE f.Account = '$Account' ORDER BY cq.stock_no";
$result = mysqli_query($link, $query);
```

從 close\_quotation 及 favorite 中取出當前的帳號的 favorite 並且印出這些 favorite 的資料在表格中

EX:若 0051 0053 006208 被 add 進 favorite 後，favorite 表格的內容會如下面的這樣

### Favorite Stock

date	stock_no	stock_name	tot_volume	tot_num	tot_money	open_price	max_price	min_price	close_price	up_down	up_diff	Actions
2023-06-06 00:00:00	0051		46327	201	2934793	63.25	63.55	63.05	63.5	-	0.2	<button>Remove</button>
2023-06-06 00:00:00	0053		6985	896	448561	64.65	64.65	64.2	64.3	+	0.25	<button>Remove</button>
2023-06-06 00:00:00	006208		5005576	5239	366371723	72.9	73.45	72.85	73.15	+	0.25	<button>Remove</button>

```
$Account = $_SESSION["name"];
if (isset($_POST['remove_from_favorite'])) {
    $stockNo = mysqli_real_escape_string($link, $_POST['stock_no']);
    $deleteQuery = "DELETE FROM favorite WHERE Account = '$Account' AND stock_no = '$stockNo' ";
    $deleteResult = mysqli_query($link, $deleteQuery);
    if ($deleteResult) {
        echo "";
    } else {
        echo "<p>Error removing data from favorites.</p>";
    }
}
```

在列出的表格中讓使用者選擇是否要從 favorite 中移除這個 stock\_num，若要刪除的話就用 delete from favorite 這個語句來清除資料

```
<?php
if ($ _SERVER["REQUEST_METHOD"] === "POST" && isset($_POST["stock_no"])) {
    $stock_no = $_POST["stock_no"];
    $query = "SELECT date, open_price, close_price, min_price, max_price FROM close_quotation WHERE stock_no = '$stock_no' ";
    $result = mysqli_query($link, $query);

    $dbData = array();
    while ($row = mysqli_fetch_assoc($result)) {
        $dbData[] = $row;
    }
    if (empty($dbData)) {
        echo "<p>The stock_no '$stock_no' does not exist.</p>";
    }
}
?>
```

這段則是最下方畫圖的功能，可以根據使用者輸入的股票代號來尋找 close\_quotation 中的資訊，並用 temp.js 的繪圖功能畫出，畫出來的圖表會在 main.php 的最下方

EX:

輸入股票代碼為 0050

Stock Number:

輸出結果如下:





1. index.php:  
這邊沒有下 query，主要是根據當前 login session 來判斷是要跳到 login 還是 main
2. login.php:

```
$sql = "SELECT * FROM Account WHERE Password='";  
$sql.= $password."' AND Account='".$name."'";  
// 執行SQL查詢  
$result = mysqli_query($link, $sql);  
$total_records = mysqli_num_rows($result);  
// 是否有查詢到使用者記錄  
if ( $total_records > 0 ) {  
    // 成功登入，指定Session變數  
    $_SESSION["login_session"] = true;  
    $_SESSION["name"] = $_POST["name"];  
    $_SESSION["password"] = $_POST["password"];  
    header("Location: index.php");  
} else { // 登入失敗
```

Login.php 處理登入的部分，會根據使用者輸入的帳號密碼來進行 select，若存在滿足條件的結果則登入成功，若否則失敗，登入成功後在交談期(session)結束或被清除前都不需要重新登入。

3. register.php:

```
$sql = "SELECT Account FROM Account WHERE Account='".$name."'";  
// 執行SQL查詢  
$result = mysqli_query($link, $sql);  
$total_records = mysqli_num_rows($result);  
// 是否有查詢到使用者記錄  
if ( $total_records == 0 ) {  
    // 帳號未註冊過，註冊帳號  
    $sql = "INSERT INTO Account VALUES('".$name."', ".$password.")";  
    mysqli_query($link, $sql);  
    echo "<center><font color='red'>";  
    echo "註冊成功!<br/>";  
    echo "</font>";  
    $_SESSION["registered"] = true;  
    unset($_SESSION["login_session"]);  
    header("Location: index.php");  
}
```

這個頁面會讀取使用者輸入的帳號密碼，利用 select 語句尋找 account 中這個帳號是否已經存在，若已經被註冊則跳出錯誤訊息要求註冊新帳號，若否則 insert 新的帳號密碼進 account 中並且跳轉到登入的頁面

#### 4. modify\_password.php:

```
$sql = "UPDATE Account SET Password='".$password.'" WHERE Account='".$name.'";  
// 執行SQL查詢  
$result = mysqli_query($link, $sql);  
if ($result) {  
    echo "<center><font color='red'>";  
    echo "修改成功!<br/>";  
    echo "</font>";  
    $_SESSION["modified"] = true;  
    unset($_SESSION["login_session"]);  
    header("Location: index.php");  
} else { // 修改失敗
```

這個頁面會讀取使用者輸入的新密碼，並且利用 update account 中的 record 來更新密碼，更新密碼之後一樣也需要用更新後的密碼重新進行登入，才可以進入 main 的頁面中

#### ● 功能統整:

1. 根據特定條件找出符合要求的資料並以表格列出
2. 讓使用者可以設定自己最愛的股票以便觀察走勢
3. 利用圖表觀察股票的走勢
4. 每天自動更新最新的資料進入資料庫
5. 建立帳號登入的功能，讓使用者可以在任何地方新增、更改、觀察自己喜歡的股票

## • Others

Progress:

時間	Expected Progress	Actual Progress
4/25	網頁架構設計完成	網頁架構設計完成
5/2	建立資料庫	建立資料庫+主頁面完成
5/9	主頁面完成	登入 and 註冊頁面完成
5/16	登入 and 註冊頁面完成	畫圖的 script(額外)完成
6/5	最初版網頁完成	最初版網頁完成

Problems we met:

1. 不會網頁設計。

Solving:

還能怎麼辦？只能去網路上自學 =3=

2. 證券交易所提供的 API 實際能取得的資料與預想的不同的。

Solving:

重新設計資料庫和呈現的內容。(網頁架構不用變)

Contribution:

- 賴宥瑜：

1. Data 的匯入及每日 update
2. Login 頁面的網頁架構
3. 讓刷新頁面時被選擇的資料不會被洗掉
4. Hackmd 更新每周進度
5. Report 中的 database 部分

- 龔大承：

1. 設計網頁中根據條件找出資料並以表格列出的相關功能
2. Session 變數的維持及頁面的跳轉
3. 登入、登出、註冊、更改密碼等功能的實現
4. Report 中 application 的部分

- 賴羿茗：

1. 設計用資料進行繪圖的功能
2. 加入與移除最愛的股票
3. 頁面的排版與美化
4. Report 中的 main idea 及 data 兩部分