**工讀金計算系統**

**專案報告書**

****

**組員：B11109003李湘菱**

**B11109004張鈺淋**

**B11109014周詠熙**

**中華民國112年6月**

**摘要**

“打工”幾乎是所有沒錢大學生賺錢的一個主要選項，但打工錢被算錯我們也不知道，因此我們設計了這個工讀金系統，希望可以減少工讀金被算錯的機率，也可以讓使用這系統的人們更方便計算工讀金。

我們使用Python和html設立了一個工讀金系統，目前系統登入後主要有Administrator跟Employee兩種身分。

Admin：

1. 下載目前班表：管理員可以下載目前班表來查看排班。
2. 上傳新的班表：管理員可以在網站中下載模板，更改模板資訊後，重新上傳上去。
3. 下載彙整後的工讀金：管理員可以下載某年某月總員工的工讀金檔案。
4. 設定：管理員可以在內部設定基本薪資和國定假日是否上班。

Employee：

1. 查看班表：員工可以下載自己的班表，查看每週上的斑。
2. 查看薪資：員工可以查看某年某月自己的總薪資。

目前的功能只需要管理員上傳每週班表，就可以計算2023年到2024年所有月份的員工總薪資，的確使工讀金計算變得簡易許多，也可避免多計或少計的狀況，只不過受限於時間，目前這系統只是個半完成品，還可以增加與修改許多功能，未來我們會改進此系統，使其變得更加完善。

關鍵詞：工讀金計算、薪資計算系統、工時記錄、自動化、用戶界面

**目錄**

[一、 緒論 1](#_Toc136390700)

[1、 動機 1](#_Toc136390701)

[2、 目的 1](#_Toc136390702)

[3、 主題探討 1](#_Toc136390703)

[二、 設計過程 2](#_Toc136390704)

[1、 方法 2](#_Toc136390705)

[2、 難題 2](#_Toc136390706)

[3、 工作分配 2](#_Toc136390707)

[三、 執行結果 3](#_Toc136390708)

[1、 系統流程圖 3](#_Toc136390709)

[2、 系統介紹與說明 3](#_Toc136390710)

[四、 結論與建議 4](#_Toc136390711)

[1、 結論 4](#_Toc136390712)

[2、 未來規劃 4](#_Toc136390713)

[五、 個人心得與如何使用ChatGPT 5](#_Toc136390714)

[1、 李湘菱 5](#_Toc136390715)

[2、 張鈺淋 6](#_Toc136390716)

[3、 周詠熙 7](#_Toc136390717)

**圖目錄**

**表目錄**

1. 緒論
   1. 動機

現今許多大學生都有打工，可是經常發生工讀金被多計或少計的情形，導致許多大學生或店家有損失的情形，因此我們打算設計一個工讀金系統，此系統可以幫助店家自動計算每年每月的工讀金，也可以讓員工加班代班的工讀金計算更加方便，以杜絕打工人或店家工讀金被計算錯誤的情形，也可以幫助店家節省算工讀金的時間成本。

* 1. 目的
     1. 杜絕工讀金被計算錯誤的情況。
     2. 節省店家計算工讀金的時間成本。
     3. 使加班代班的工讀金計算更加方便。
  2. 主題探討

我們組別大部分都有打過工，也曾經發生過工讀金被少算的情況，如果不是本身就有在登記，可能就會因此損失一大筆錢，但自己登記實在是太麻煩了，所以我們想設計一個工讀金系統來自動計算每年每月的工讀金，如果遇到加班代班也可以自動幫我們計算加班代班後的總薪資，我們只需要登入自己帳號去查看自己班表和薪資就好了。

1. 設計過程
   1. 方法

此系統配合課程所學，主要使用Python製作，同時使用了html建立使用者介面，方便使用者直接使用，

* + 1. Python

撰寫途中使用到Flask套件、Pandas套件、csv套件與datetime套件。

Flask套件：連結所有html介面，秀出每個介面的格式並抓取介面使用者的輸入與上傳的檔案，傳回Python做計算和編成新的csv檔案，再回傳回html介面供使用者查看與下載。

pandas套件：讀取csv檔案，並將csv檔案內容傳回前端介面。

csv套件：進行csv檔案的讀取與輸入。

datetime套件：抓取日期並對日期做處理。

* + 1. Html

建立前端使用者介面，方便使用者進行操作，使用者可以在此介面輸入資訊傳到後端進行計算，並在此介面上傳與下載檔案。

* 1. 難題

|  |  |
| --- | --- |
| 難題 | 解決方式 |
| 意見分歧、架構一直變更 | 花時間坐下來重新討論。 |
| 自我要求 | 自我要求。 |
| 程式底子不佳，容易卡住 | 組員們相互幫助。 |
| 結合現實時間下去計算 | 使用chatgpt作為輔助，學習datetime套件。 |
| 網頁介面過於簡陋 | bootstrap |

* 1. 工作分配

|  |  |
| --- | --- |
| 學號+人名 | 工作內容 |
| B11109003李湘菱 | schedule.py + 專案報告書 |
| B11109004張鈺淋 | money.py + 簡報 |
| B11109014周詠熙 | main.py + html設計 |

1. 執行結果
   1. 系統流程圖
   2. 系統介紹與說明

登入：登入介面，管理員帳號：admin，密碼：1234，員工帳號為員工姓名，密碼固定為5678。

管理員：有四個功能，分別為下載和上傳班表、下載彙整後的工讀金和設定。

下載目前班表：可下載目前的每週班表的csv檔案。

上傳新班表：可下載班表模板下去更改，之後再上傳新的班表。

下載彙整後工讀金：可下載全部員工某年某月的總薪資csv檔案。

設定：可進去裡面更改基本薪資和國定假日是否上班(如上班，薪水乘二)。

員工：有兩個功能，可察看班表與薪資。

查看班表：可查看員工自己的上班班表(csv檔案)。

查看薪資：可查看員工某年某月的總薪資。

1. 結論與建議
   1. 結論

目前的功能需要由管理員上傳每週班表(通常是固定的)，就可以計算2023年到2024年所有月份的員工總薪資(因為如果要處理到國定假日，需要使用到網路爬蟲，但因為時間問題，我們暫時沒做此功能)，此系統使工讀金計算變得簡易許多，節省了店家或部門計算工讀金的時間成本，也避免多計或少計的狀況，只不過受限於時間，目前這系統只是個半完成品，未來可以增加與修改許多功能，我們會持續改進此系統，使其變得更加完善。

* 1. 未來規劃

受限於時間，我們並沒有將功能全部做出來，以下功能是我們的想法，期許未來可以將這些功能增加到系統上，使系統更加完善。

|  |  |
| --- | --- |
| 新增加代班功能 | 只要是工讀，常常會遇到加班或代班，我們希望能在系統增加加班與代班功能，讓工讀金計算可以變得更加便利，不需要花費額外時間去計算加班和代班。 |
| 新增部門 | 目前管理員只有一個，希望未來可以增加許多部門的管理員，讓整個系統可以變得更加龐大。 |
| 簽名比對 | 希望未來可以增加簽名比對功能，使抓取員工出缺席狀況變得更加便利。 |
| 新增資料庫功能 | 可以讓管理員看歷年薪資，也對管理員工更加便利。 |

1. 個人心得與如何使用ChatGPT
   1. 李湘菱

此次專案我主要負責schedule.py，老實說其實也沒有遇到很大的困難，大部分都是抓取csv檔案，然後下去做修改後上傳，唯一讓我卡比較久的是要抓取現實時間下去計算總時數。

大部分程式碼都是自己寫，而chatgpt則是輔助，主要使用到chatgpt的地方就是抓取現實時間，schedule.py需要抓取現實時間下去做每年每月每人的總時數，但抓取現實時間一開始的確困擾我許久，後來利用chatgpt學習到了datetime套件後，也成功做出了此功能。

類別schedule最主要是要讓人可以從此類別拿到班表與時數，這能使工讀金計算變得更加簡易，schedule有讀檔功能可以讀取csv檔案，也有輸出功能，可以輸出修改好的班表，方便未來使用者調用，同時可以輸出每位員工總時數，讓工讀金計算更加方便。

在這次專案過程中，其實我也學習到許多東西，除了平時課程學到的功能外，還學到了其他套件的功能與排版，這次專案時間並不充裕，又遇到期末週，因此其實沒有什麼時間可以製作，老實說，如果有足夠時間，我覺得我們可以將此系統做得更加完善與龐大，但我還是很滿意這次的製作，雖然這只是個小小系統，但他容納了許多課堂學習的功能，也一次又一次的突破我們自己的原本知識。

這次專案我真的要感謝我的組員們，如果沒有他們的幫助，我覺得並沒有辦法做出此系統，而且很感謝他們包容很忙的我，也願意在我有問題時給予幫助，這次專案的成就感滿滿，除了學習很多知識外，還增進了我和組員們的感情，也讓我想到之前高中寫專題的時候，真是滿滿感慨呀(><)。

* 1. 張鈺淋

這次期末專案我主要是負責money.py檔案，由於組內分成三個主類別，而計算薪資的部分需要大量取得其他檔案的方法來操作，因此，如何正確引入類別並取得相應的參數，是最常用到的技巧。

類別MoneyCalculator的功能是用來計算薪資，需要設定幾項基本變數及用途，如基本薪資、計算本月總薪資的公式，有了這些設定後，下一步便是傳入所需的參數（接收來自scheduleclass實例的資料），以計算出最終結果。

以上取得的都是「個別員工」的資料，因此需要用到Python中的字典，方便將「所有員工」的資訊整合起來。而在取得全體員工的資訊後，藉由引入csv模組，先使用迴圈遍歷資料內的所有員工，並排除掉薪資為0的資料，後將統整資料寫入csv檔，輸出成具實用性的報表。

因為這次設計的系統可供使用者輸入，因此在最後新增了一個方法employee\_salary，用來比對使用者輸入的與資料內的姓名，達到查詢特定員工薪資的目的。

在寫程式的過程中，由於負責的這個類別不太會運用到複雜的邏輯，所以寫起來還算順利。但本身的程式底子並不是很好，尤其剛開始寫的時候如何傳入其他類別的實例、參數要放在哪裡等，這些會很容易混淆，所以碰到問題時會請教程式能力較好的組員，解決問題的同時、一邊將用法記下來，更能加深印象。

程式完成後，參考了教授提供的風格指南，又回頭去看自己的程式碼，確實有些地方寫得太長或註解雜亂。後來一邊修改編排，一邊新增註解，發現這樣一來，不但可以重新確認及更加瞭解自己所撰寫的程式碼，也可以使程式看起來更美觀，方便所有人閱讀。學會這些編排技巧，對日後程式碼的管理相當有幫助。

最後，我想謝謝可靠的組員們，原先聽到期末要做專案報告，其實很擔心自己的能力不好，沒能盡到一份力，討論時也因為自己對於Python的瞭解不夠透徹，很多時候是聆聽組員們的想法。後來分配好每個人所負責的類別，我也開始著手寫程式，這期間不管遇到了什麼問題，組員們總是很有耐心地教我、給我建議，告訴我哪邊可以增加功能、哪邊可以寫得更好，在她們身上真的學到了很多。雖然money.py這個檔案並不是特別困難，但寫出來所得到的成就感卻是最大的收穫，在之後面對其他的問題或專案時，才會更有信心去處理及學習，一步一步提升自我的程式能力。

* 1. 周詠熙