

순장훈련학교 순편성 프로그램

Date: 2022.08.31

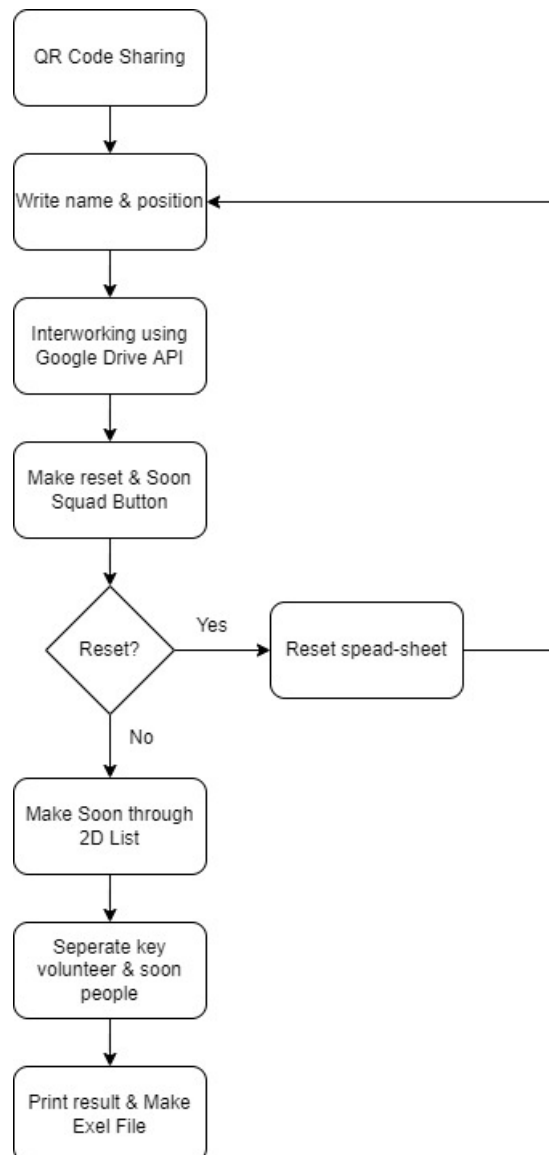
Name: ChangMin An

Github: [Link](#)

I. Introduction

CCC 순장훈련학교 조편성할 때 많은 인원으로 인해 현장에서 조를 편성하는 데에 있어서 어려움이 있었다. 이를 보완하고 순장 3명에서 5명, 순원 1명에서 2명으로 편성하는 알고리즘을 구현하였다.

II. Flow-Chart



III. QR Code



III. Result

```
전체인원 : 34명
순장인원 : 26명
예비순장인원 : 7명
1 조 : ['유형주', '배민석', '박성준', '박희선', '곽찬영', '김선교']
2 조 : ['안풍삼', '조예술', '김민서', '박범', '김건하', '백남경']
3 조 : ['송희영', '안효건', '고나현', '이예은', '정성구']
4 조 : ['안창민', '이채린', '전민성', '박정배', '노형진']
5 조 : ['강주안', '테스트', '전영식', '변이경', '정경구']
6 조 : ['윤한비', '김서영', '이지은', '정석민', '강성민', '권하민']
```

각 순에 6명씩 편성하였으며, 남는 인원의 경우 기존의 조에 추가적으로 넣었다. 기본적으로 순장 4명, 순원 1~2명으로 편성되어 있으며, 이는 사용자가 원하는 만큼 상황에 맞게 변경할 수 있다.

조	이름
1조	['유형주', '박희선', '박정배', '조예술', '곽찬영', '정경구']
2조	['윤한비', '배민석', '안효건', '김서영', '강주안', '정성구']
3조	['송희영', '김건하', '전민성', '이지은', '백남경']
4조	['테스트', '김민서', '박범', '이채린', '권하민']
5조	['안창민', '안풍삼', '고나현', '변이경', '강성민']
6조	['이예은', '박성준', '정석민', '전영식', '노형진', '김선교']

또한 출력으로 엑셀파일을 자동으로 생성하여 CCC 공지방에 공지하기 수월하게 데이터를 모았다. 엑셀 파일명은 'Soon_List.xlsx'이다.

IV. Code

```
'''
* ****
* @author    ChangMin An
* @Mod       2022 - 08 - 31
```

```

* @brief      Soon Generate Program
* *****

'''

import gspread
from oauth2client.service_account import ServiceAccountCredentials
import random as rm
import tkinter as tk
import pandas as pd

# ===== Google Drive Environment ===== #
scope = [
    'https://spreadsheets.google.com/feeds',
    'https://www.googleapis.com/auth/drive',
]
json_file_name = 'soon-361013-a1bfe708a386.json'
credentials = ServiceAccountCredentials.from_json_keyfile_name(json_file_name,
scope)
gc = gspread.authorize(credentials)
spreadsheet_url = 'https://docs.google.com/spreadsheets/d/1ODASwz-
Ym1wPUZAIysxSaCD5prJqMQN07Ci9tBFtJ_o/edit#gid=1121346661'

# 문서 불러오기
doc = gc.open_by_url(spreadsheet_url)

# a 시트 불러오기
worksheet = doc.worksheet('sheet1')

# 순 편성 Flag
soon_flag = False

# ===== Tkinter Environment ===== #
class Test():
    def __init__(self):
        self.root = tk.Tk()
        self.root.geometry("200x100")
        self.buttonA = tk.Button(self.root,
                                text = "Clear",
                                command=self.Clear_Sheet)

        self.buttonB = tk.Button(self.root,
                                text="Click to make Soon",
                                command=self.Make_Soon)

        self.buttonA.pack(side=tk.LEFT)
        self.buttonB.pack(side=tk.RIGHT)
        self.root.mainloop()

    def Clear_Sheet(self):
        worksheet.clear()

    def Make_Soon(self):
        global soon_flag
        soon_flag = True

app=Test()

```

```

# ===== Make Soon Squad ===== #
if soon_flag == True:
    # Make Name & 순장/예비순장 List
    names = worksheet.col_values(2)
    positions = worksheet.col_values(3)

    values = [[0 for col in range(len(names))] for row in range(2)]
    key_volunteer_sort = []
    soon_people_sort = []
    key_volunteer = []
    soon_people = []

    for i in range(1, len(names)):
        values[0][i] = names[i]
        values[1][i] = positions[i]

    for i in range(2):
        values[i] = list(filter(None, values[i]))

    # 전체 인원 / 순장수
    whole_num = len(values[0])
    squad_key = 4 # 순장 수 순원 수 비교하면서 적절하게 조절하면 됨

    print(f"전체인원 : {whole_num}명")

    # 1. 순장, 예비순장 분리
    for i in range(len(values[1])):
        if values[1][i] == '순장':
            key_volunteer_sort.append(values[0][i])
        else:
            soon_people_sort.append(values[0][i])

    # 이름 중복 제거
    for value in key_volunteer_sort:
        if value not in key_volunteer:
            key_volunteer.append(value)

    for value in soon_people_sort:
        if value not in soon_people:
            soon_people.append(value)

    whole_key = len(key_volunteer)
    whole_soon = len(soon_people)

    # 순장, 예비순장 인원 출력
    print(f"순장인원 : {whole_key}명")
    print(f"예비순장인원 : {whole_soon}명")

    # 2. 순 배치 한 조 당 6명 (4 / 2)
    squad_buf = []
    # 순장 배치
    for i in range(whole_key//squad_key):
        squad = []

        while len(squad) < squad_key:
            if len(key_volunteer) != 0:

```

```

        student = rm.choice(key_volunteer)
        if student not in squad:
            squad.append(student)
            key_volunteer.remove(student)

    # print(f"{i+1}", '조 :', squad)
    squad_buf.append(squad)

for i in range(len(key_volunteer)):
    squad_buf[i].append(key_volunteer[i])

# 예비순장 배치
for i in range(len(squad_buf)):
    if len(soonest_people) != 0:
        student = rm.choice(soonest_people)
        if student not in squad:
            squad_buf[i].append(student)
            soonest_people.remove(student)

for i in range((whole_key//squad_key) - 1, 0, -1):
    if len(soonest_people) != 0:
        student = rm.choice(soonest_people)
        if student not in squad:
            squad_buf[i].append(student)
            soonest_people.remove(student)
    else:
        break

# 조 편성 결과 출력
for i in range(len(squad_buf)):
    print(f"{i+1}", '조 :', squad_buf[i])

# 엑셀파일로 Export
formation = []
for i in range(len(squad_buf)):
    formation.append(f"{i+1}"+'조')

Soon = pd.DataFrame()
Soon['조'] = formation
Soon['이름'] = squad_buf

Soon.to_excel(r"C:\Users\AnChangMin\Desktop\21-2\CCC\Soon_List.xlsx",
index=False)

```