

11/3 작업 내역 (2)

간호사들의 근무 우선순위를 정하기 위해 필요했던 정보들을 `PriorityRawData` 클래스로 통합 관리합니다.

1. 클래스 설계

```
from collections import deque

class PriorityRawData:

    def __init__(self):
        self.nurse_pk = 0
        self.team_pk = 0
        self.nurse_grade = 0
        self.monthly_shift = 0
        self.monthly_night_shift = 0
        self.off_s = 0
        self.last_shift = 0
        self.off_streaks = 0
        self.shift_streaks = 0
        self.weekly_schedule = deque()

    def fill_last_weeks_schedule(self, last_schedule):
        for day_index in range(-1, -8, -1):
            self.weekly_schedule.append(
                last_schedule[-day_index]
            )

    def find_last_shift(self):
        self.last_shift = self.weekly_schedule[-1]

    def find_off_streaks(self):
        if not self.last_shift and not self.weekly_schedule[-2]:
            self.off_streaks = 2
        elif not self.last_shift:
            self.off_streaks = 1

    def find_shift_streaks(self):
        for i in range(-1, -8, -1):
            if self.weekly_schedule[i]:
                self.shift_streaks += 1
            else:
                break

    def update_self(self, shift):
        # 1. 가장 최근 근무 기록 갱신
        self.last_shift = shift

        # 1) 휴무
        # 만약 오늘 쉬었다면
        # 연속 근무일수 초기화
        if not shift:
            self.shift_streaks = 0
```

```

        self.off_streaks += 1
    # 2) 근무
    else:
        self.monthly_shift += 1
        self.shift_streaks += 1
        self.off_streaks = 0
    # 예외 - 야간 근무
    if shift == 3:
        self.monthly_night_shift += 1

```

2. 개선사항

1) divide_nurse_info

매개변수:

0. team_list: 제작을 원하는 팀 번호가 담긴 리스트.
1. nurse_profile_dict: view 함수 중 get_nurse_info의 출력값.
 - └ key = pk
 - └ value = [pk, level, team, off_cnt]
2. nurse_last_months_schedule_dict: get_last_schedule의 출력값.
 - └ key = pk
 - └ value = [한 달 개인 듀티표]

출력값:

Key: 팀 번호들

value: 딕셔너리.

└ key: 팀별 간호사 pk

└ value: 팀별 간호사 PriorityRawData 객체.

수정 전:

```

def divide_nurse_info_by_team(
    team_list,
    nurse_profile_dict,
    nurse_last_months_schedule_dict
):
    # 1. 선언
    # 출력을 위한 dict 선언.
    divided_nurse_info_dict_by_team = dict()
    nurse_pk_by_team = dict()
    for team_number in team_list:
        divided_nurse_info_dict_by_team[team_number] = dict()
        nurse_pk_by_team[team_number] = list()

    # 2. 연산
    for nurse_detail in nurse_profile_dict.values():

        # 1) 앞부분에 필요한 값들.
        nurse_pk = nurse_detail[0]
        nurse_grade = nurse_detail[1]
        nurse_team = nurse_detail[2]
        nurse_offs = nurse_detail[3]

```

```

# 2) 뒷부분에 필요한 값들.
nurse_last_month_schedule = nurse_last_months_schedule_dict[nurse_pk]
nurse_last_shift = nurse_last_month_schedule[-1]
nurse_shift_streaks = 0
if not nurse_last_shift and not nurse_last_month_schedule[-2]:
    nurse_off_streaks = 2
elif not nurse_last_shift:
    nurse_off_streaks = 1
else:
    nurse_off_streaks = 0
    for i in range(1, 6):
        if nurse_last_month_schedule[-i]:
            nurse_shift_streaks += 1
        else:
            break

# 3) 딕셔너리에 삽입
nurse_pk_by_team[nurse_team].append(nurse_pk)

divided_nurse_info_dict_by_team[nurse_team][nurse_pk] = [
    nurse_pk,
    nurse_grade,
    nurse_team,
    0,
    nurse_shift_streaks,
    nurse_offs,
    0,
    0,
    nurse_off_streaks,
    nurse_last_shift
]

# pprint(nurse_pk_by_team)
return divided_nurse_info_dict_by_team, nurse_pk_by_team

```

수정 후

```

def divide_nurse_info_by_team(
    team_list,
    nurse_profile_dict,
    last_months_schedule_dict
):

    # 1. 선언
    # 출력을 위한 dict 선언.
    nurse_info_by_team = dict()
    nurse_pk_by_team = dict()
    for team_number in team_list:
        nurse_info_by_team[team_number] = dict()
        nurse_pk_by_team[team_number] = list()

    # 2. 연산
    for nurse_detail in nurse_profile_dict.values():
        personal_data = PriorityRawData()

        # 1) profile_dict에서 가져오는 부분

```

```

personal_data.nurse_pk = nurse_pk = nurse_detail[0]
personal_data.nurse_grade = nurse_detail[1]
personal_data.team_pk = nurse_team = nurse_detail[2]
personal_data.off_s = nurse_detail[3]

# 2) schedule_dict에서 가져오는 부분.
last_month_schedule = last_months_schedule_dict[nurse_pk]
personal_data.fill_last_weeks_schedule(last_month_schedule)
personal_data.find_last_shift()
personal_data.find_off_streaks()

# 3) 딕셔너리에 삽입
nurse_pk_by_team[nurse_team].append(nurse_pk)
nurse_info_by_team[nurse_team][nurse_pk] = personal_data

return nurse_info_by_team, nurse_pk_by_team

```

2) update_nurse_infos

매개변수:

nurse_infos: 딕셔너리.

- └ key: nurse_pk
- └ value: PriorityRawData 객체

temporary_schedule:

- └ 임시 스케줄.

반환값: 업데이트된 nurse_info

수정 전

```

def update_nurse_info(nurse_info, temporary_schedule):
    """
    0 NURSE_NUMBER 간호사 일련번호
    1 NURSE_GRADE 간호사 grade
    2 TEAM_NUMBER 팀번호
    3 SHIFTS, 이번 달 근무 일수
    4 SHIFT_STREAKS, 연속 근무일 수
    5 OFFS, 그.. 마크다운에 있는 'OFF' 참조.
    6 MONTHLY_NIGHT_SHIFTS, 한 달에 night 근무한 횟수
    7 VACATION_INFO, 휴가 정보(외부 딕셔너리로 수정 예정)
    8 OFF_STREAKS, 연속 휴무
    9 LAST_SHIFT, 마지막 근무 정보
    """

    # 모든 4개의 시프트를 순회하며
    for shift in range(4):

        # 모든 간호사들 순회.
        for nurse in temporary_schedule[shift]:

            # 1. 가장 최근 근무 기록 갱신
            nurse_info[nurse][9] = shift

            # 1) 휴무

```

```

# 만약 오늘 쉬었다면
# 연속 근무일수 초기화
if not shift:
    nurse_info[nurse][4] = 0
    nurse_info[nurse][8] += 1

# 2) 근무
# (1) DAY 근무
elif shift == 1:
    # a. 한 달 전체 근무일 1 가산
    nurse_info[nurse][3] += 1
    # b. 연속 근무일 수 1 가산
    nurse_info[nurse][4] += 1
    # c. 연속 휴무일 수 0으로 수정
    nurse_info[nurse][8] = 0

# (2) EVENING 근무
elif shift == 2:
    # a. 한 달 근무일 1 가산
    nurse_info[nurse][3] += 1
    # b. 연속 근무일 수 1 가산
    nurse_info[nurse][4] += 1
    # c. 연속 휴무일 수 0으로 수정.
    nurse_info[nurse][8] = 0

# (3) NIGHT 근무
elif shift == 3:
    # a. 한 달 근무일 수 1 가산
    nurse_info[nurse][3] += 1
    # b. 연속 근무일 수 1 가산
    nurse_info[nurse][4] += 1
    # c. 한 달 전체 나이트 근무 1 가산
    nurse_info[nurse][6] += 1
    # d. 연속 휴무일 수 0으로 조정.
    nurse_info[nurse][8] = 0

return nurse_info

```

수정 후

```

def update_nurse_infos(nurse_infos, temporary_schedule):

    # 모든 4개의 시프트를 순회하며
    for shift in range(4):
        # 모든 간호사들 순회.
        for nurse_pk in temporary_schedule[shift]:
            nurse_infos[nurse_pk].update_self(shift)

    return nurse_infos

```

