

```
In [ ]: import pandas as pd
```

```
In [ ]: df = pd.read_csv('c:/mid_project/recommend.csv') # 전처리 된 데이터 불러오기( 국민연령별추천운동정보 데이터에서 불필요 컬럼 삭제)
df
```

```
Out[ ]:
```

	AGRDE_FLAG_NM	COAW_FLAG_NM	SPORTS_STEP_NM	RECOMEND_MVM_NM
0	10대	4	본운동	줄넘기 운동
1	10대	4	본운동	자전거타기
2	10대	4	본운동	조깅
3	10대	4	본운동	버피 테스트
4	10대	4	본운동	맨몸운동 루틴프로그램
...	...	...	...	...
5035	70대 이상	3	마무리운동	누워서 전신 뻗기
5036	70대 이상	3	마무리운동	넓다리 안쪽 스트레칭
5037	70대 이상	3	마무리운동	넓다리 앞쪽 스트레칭
5038	70대 이상	3	마무리운동	엉덩이 스트레칭
5039	70대 이상	3	마무리운동	옆구리 스트레칭

5040 rows × 4 columns

## 연령별 데이터 분할

```
In [ ]: child = df[df['AGRDE_FLAG_NM'] == '10대']
teens = df[df['AGRDE_FLAG_NM'] == '10대']
adult = df[df['AGRDE_FLAG_NM'].str.contains('20대|30대|40대|50대')]
elders = df[df['AGRDE_FLAG_NM'].str.contains('60대|70대 이상')]
```

## 등급별로 랜덤하게 추천운동 반환

```
In [ ]: # 등급별로 추천 운동 상위 20개 종목 리스트로 반환하는 함수
def top_recommend_movements(data, coaw_flag_num, top_n=20):
    filtered_data = data[data['COAW_FLAG_NM'] == coaw_flag_num]
    top_movements = filtered_data['RECOMEND_MVM_NM'].value_counts().sort_values(ascending=False).head(top_n)
    return top_movements.index.tolist()

# 데이터에 top_recommend_movements 함수를 적용해 딕셔너리로 저장하는 함수
def grade_top_recommend_movements(data, grade_name):
    grade_data = {}
    for i in range(1, 5):
        grade_data[f'{grade_name}_{i}'] = top_recommend_movements(data, i)
    return grade_data

# 각 등급을 위한 상위 20개 추천운동 딕셔너리(연령 데이터별)
child_grade_movements = grade_top_recommend_movements(child, 'child')
teens_grade_movements = grade_top_recommend_movements(teens, 'teens')
adult_grade_movements = grade_top_recommend_movements(adult, 'adult')
elders_grade_movements = grade_top_recommend_movements(elders, 'elders')
```

```
In [ ]: import random

# 사용자 정보 입력받기
name = input('이름을 입력하세요: ')
age = int(input('나이를 입력하세요: '))
grade = int(input('등급을 입력하세요(참가증은 4를 입력해주세요.): '))

# 연령 및 등급에 따라 적절한 운동 추천 리스트 선택
if 11 <= age <= 12:
    selected_recommendations = child_grade_movements[f'child_{grade}']

if 13 <= age <= 19:
    selected_recommendations = teens_grade_movements[f'teens_{grade}']

if 20 <= age <= 59:
    selected_recommendations = adult_grade_movements[f'adult_{grade}']

if 60 <= age <= 105:
    selected_recommendations = elders_grade_movements[f'elders_{grade}']

# 첫 번째와 두 번째 요소 고정 추출 및 나머지 세 개의 요소 랜덤 선택하는 함수
```

```
def get_random_elements(recommendations):  
    fixed_elements = recommendations[:2]  
    random_elements = random.sample(recommendations[2:], 3)  
    return fixed_elements + random_elements  
  
# 랜덤 추천 생성 및 출력  
random_result = get_random_elements(selected_recommendations)  
print(f'{name}님({age}세/{grade}등급) 추천 운동 리스트')  
print(random_result)
```

김멋사님 (15세/4등급) 추천 운동 리스트

['동적 스트레칭 루틴프로그램', '정적 스트레칭 루틴프로그램', '짝 스트레칭 루틴프로그램', '맨몸운동 루틴프로그램', '줄넘기 운동']

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js