



영진전문대학교

글로벌시스템융합과 - GSC

# 자율학습용 코딩 문제 #3

정영철 교수

글로벌시스템융합과



A.I WITH BETTER LIFE

# 문제 1 : 사전 예약 시스템 시뮬레이터 (1)

여러분은 사전 예약 시스템의 로직을 시뮬레이션하는 프로그램을 작성해야 합니다.  
이 시스템은 사용자가 특정 이벤트의 사전 예약을 진행할 때, 사용자의 입력에 따라  
예약 가능 여부를 판단하고 그에 따른 결과를 출력합니다

## ➤ 시나리오 및 요구사항

### 1. 프로그램은 사용자에게 다음 정보를 입력받습니다:

- 사용자의 나이 (age)
- 사용자가 예약하려는 이벤트 코드 (event\_code) - 이벤트 코드는 'E1', 'E2', 'E3' 중 하나입니다.
- 사용자가 예약을 원하는 날짜 (reservation\_date) - 예약 날짜는 1부터 30까지의 숫자로 입력됩니다.

### 2. 각 이벤트 코드별로 다음과 같은 규칙이 적용됩니다:

- 'E1': 만 18세 이상만 예약 가능합니다.
- 'E2': 모든 연령대가 예약 가능하지만, 날짜는 짝수일에만 예약할 수 있습니다.
- 'E3': 만 16세 이상만 예약할 수 있으며, 7의 배수인 날짜에만 예약 가능합니다.

### 3. 사용자는 각 이벤트 코드에 따라 예약을 시도할 때 다음과 같은 결과를 받게 됩니다:

- 예약 성공: "예약이 완료되었습니다!"
- 나이 미달: "나이 제한으로 인해 예약할 수 없습니다."
- 날짜 제한: "선택하신 날짜에는 예약할 수 없습니다."
- 잘못된 입력: "잘못된 입력입니다. 프로그램을 종료합니다"

# 문제 1 : 사전 예약 시스템 시뮬레이터 (2)

## ➤ 문제 해결을 위한 추가 정보

- 나이 제한이 있는 이벤트는 사용자의 나이를 확인하여 조건에 맞지 않을 경우 "나이 제한으로 인해 예약할 수 없습니다."라고 출력해야 합니다.
- 날짜 제한이 있는 이벤트는 사용자가 입력한 날짜를 검증하여 조건에 맞지 않을 경우 "선택하신 날짜에는 예약할 수 없습니다."라고 출력해야 합니다.
- 사용자가 입력한 event\_code나 reservation\_date가 정해진 형식에 맞지 않을 경우 "잘못된 입력입니다. 프로그램을 종료합니다."라고 출력하고 프로그램을 종료해야 합니다.

✓ 첫 번째 실행: E1 code 예약 성공 예

나이를 입력하세요: 20  
예약하려는 이벤트 코드를 입력하세요: E1  
원하는 예약 날짜를 입력하세요: 3  
예약이 완료되었습니다!

✓ 두 번째 실행: E1 code 예약 실패 → 나이 제한

나이를 입력하세요: 15  
예약하려는 이벤트 코드를 입력하세요: E1  
원하는 예약 날짜를 입력하세요: 6  
나이 제한으로 인해 예약할 수 없습니다.

✓ 세 번째 실행: E2 code 예약 성공 예

나이를 입력하세요: 12  
예약하려는 이벤트 코드를 입력하세요: E2  
원하는 예약 날짜를 입력하세요: 8  
예약이 완료되었습니다!

✓ 네 번째 실행: E2 code 예약 실패 → 예약 날짜 홀수

나이를 입력하세요: 12  
예약하려는 이벤트 코드를 입력하세요: E2  
원하는 예약 날짜를 입력하세요: 3  
선택하신 날짜에는 예약할 수 없습니다.

# 문제 1 : 사전 예약 시스템 시뮬레이터 (3)

✓ 다섯 번째 실행: E3 code 예약 성공 예

나이를 입력하세요: 17  
예약하려는 이벤트 코드를 입력하세요: E3  
원하는 예약 날짜를 입력하세요: 14  
예약이 완료되었습니다!

✓ 일곱 번째 실행: E3 code 예약 실패 → 나이제한

나이를 입력하세요: 15  
예약하려는 이벤트 코드를 입력하세요: E3  
원하는 예약 날짜를 입력하세요: 14  
나이 제한으로 인해 예약할 수 없습니다.

✓ 여덟 번째 실행: E3 code 예약 실패 → 선택한 날짜가 7의 배수 아님

나이를 입력하세요: 17  
예약하려는 이벤트 코드를 입력하세요: E3  
원하는 예약 날짜를 입력하세요: 15  
선택하신 날짜에는 예약할 수 없습니다.

✓ 아홉 번째 실행: 예약 실패 → 예약코드 입력 오류

나이를 입력하세요: 20  
예약하려는 이벤트 코드를 입력하세요: E4  
원하는 예약 날짜를 입력하세요: 20  
잘못된 입력입니다. 프로그램을 종료합니다.

✓ 열 번째 실행: 예약 실패 → 날짜 입력 유효범위 오류

나이를 입력하세요: 20  
예약하려는 이벤트 코드를 입력하세요: E1  
원하는 예약 날짜를 입력하세요: -1  
잘못된 입력입니다. 프로그램을 종료합니다.

나이를 입력하세요: 20  
예약하려는 이벤트 코드를 입력하세요: E1  
원하는 예약 날짜를 입력하세요: 31  
잘못된 입력입니다. 프로그램을 종료합니다.

✓ 열한 번째 실행: 예약 실패 → 날짜 입력 유효범위 오류 4

## 문제 2 : 출석 점수 프로그램 (1)

- 대학에서의 출석 점수는 여러분의 최종 성적 결정에 중요한 역할을 합니다.  
영진전문대학에서는 출석 점수 만점을 20점으로 설정하고 있으며, 다음 기준에 따라 점수를 산정합니다
  - 출석점수 계산법:
    - 20점에서 결석시간에 비례하는 점수를 차감합니다
    - 계산식:  $20점 - (20 \times \text{결석시간 수} / \text{총 수업시간 수})$
  - 총 수업 시간 계산법:
    - $\text{시수/주} \times 15$
  - 지각 처리 규칙:
    - 지각 3회는 결석 1시간으로 처리
  - 결석 처리 규칙:
    - 결석시수가 총 수업시수의 1/4을 초과할 경우 학점 미부여 (F처리)
- 문제 요구 사항
  - 입력: 사용자로부터 주당 수업 시간(시수/주), 결석한 총 시간, 지각 횟수를 입력 받습니다.
  - 출력
    - 위의 기준에 따라 계산된 출석 점수를 출력합니다.
    - 결석 시간이 총 수업 시간의 1/4을 초과하는 경우, "F (학점 미부여)"를 출력합니다.

## 문제 2 : 출석 점수 프로그램 (2)

```
def calculate_attendance_score(hours_per_week, absence_hours, tardy_count):  
    """  
    주당 수업 시간, 결석 시간, 지각 횟수를 기반으로 출석 점수를 계산하는 함수.  
  
    Parameters:  
    - hours_per_week: 주당 수업 시간(시수/주)  
    - absence_hours: 결석한 총 시간  
    - tardy_count: 지각 횟수  
  
    Returns:  
    - attendance_score: 계산된 출석 점수  
    """
```

# 사용자 입력

```
hours_per_week = int(input("주당 수업 시간을 입력하세요: "))  
absence_hours = int(input("결석한 총 시간을 입력하세요: "))  
tardy_count = int(input("지각 횟수를 입력하세요: "))
```

# 출석 점수 계산 및 출력

✓ 첫 번째 실행

```
주당 수업 시간을 입력하세요: 6  
결석한 총 시간을 입력하세요: 12  
지각 횟수를 입력하세요: 0  
당신의 출석 점수는 17.33점입니다.
```

✓ 두 번째 실행

```
주당 수업 시간을 입력하세요: 6  
결석한 총 시간을 입력하세요: 12  
지각 횟수를 입력하세요: 4  
당신의 출석 점수는 17.11점입니다.
```

✓ 세 번째 실행

```
주당 수업 시간을 입력하세요: 2  
결석한 총 시간을 입력하세요: 8  
지각 횟수를 입력하세요: 0  
당신의 출석 점수는 F (학점 미부여)점입니다.
```

# Q/A

# 감사합니다



주문식교육의 산실  
**영진전문대학교**