

Stage 2 - 야구 게임 만들기

숫자 4개를 가지고 자리와 수의 크기를 비교하여 스트라이크와 볼을 판단하는 게임을 만들어봅시다. 이전까지 배운 변수, 할당연산자, 입출력, 조건문, 반복문, 랜덤 라이브러리 등을 활용합니다.

야구 게임 코딩하기

본격적으로 야구 게임을 만들어보겠습니다. 이번 Stage를 통하여 이전 시간에 그렸던 순서도에서 난수를 발생하여 문제를 출제하는 부분과 사용자로부터 정답을 입력받는 부분까지 코딩합니다. 잘 따라오세요!

```
baseball_game (2) × /Users/duhui/venv/python-basic/bin/python /Users/duhui/
야구 게임 프로그램입니다!
[5, 9, 7, 6]
답을 맞추어 보세요 : 1234
index : 0, value : 1
index : 1, value : 2
index : 2, value : 3
index : 3, value : 4

Process finished with exit code 0
|
```

야구 게임

난수 발생하기

랜덤 라이브러리 활용

로또 발생기를 기억하시죠?! 로또 발생기와 아주 똑같습니다. 다만 여기서 조금 달라진 부분이 있다면, 로또 발생기는 7자리지만 야구 게임은 4자리라는 것과 랜덤 라이브러리에서 `randrange`문이 아닌 `randint`문을 사용해서 좀 더 직관적으로 범위를 설정합니다. 아래와 같이 코딩하여 난수를 생성해 봅시다.

```
games/baseball_game
```

```
1 import random
2
3 print("야구 게임 프로그램입니다!")
4
5 new_number = random.randint(0, 9)
6 print(new_number)
```

중복 제거하기

이번에는 'answer'라는 정답 리스트 변수를 하나 선언하고 정답 리스트 안에 새로운 숫자가 있는지 중복여부를 파악하여 4자리의 숫자를 만들어주는 과정까지 코딩해보겠습니다.

```
games/baseball_game
```

```
1 import random
2
3 print("야구 게임 프로그램입니다!")
4
5 # 랜덤하게 4자리의 숫자 만들기 (각 자리의 수는 중복 불가)
6 answer = list()
7
8 while len(answer) != 4:
9     new_number = random.randint(0, 9)
10    if new_number not in answer:
11        answer.append(new_number)
12
13 print(answer)
```

정답 입력받기

사용자로부터 정답 입력받기

위의 과정을 통해 4자리의 정답 리스트를 생성했습니다. 이번에는 사용자로부터 답을 맞추는 과정을 입력받아 그 안의 값을 하나하나 조회해보는 과정입니다.

먼저는 input문을 통해 4자리의 값을 입력받습니다. 입력받은 4자리의 숫자는 문자열 자료형입니다. 문자열 자료형도 enumerate문에 넣어서 반복문을 실행하면 값과 인덱스를 하나하나 조회할 수 있습니다. 아래와 같이 코딩해보세요.

games/baseball_game

```
1  # 유저에게 입력 받음
2  your_answer = input("답을 맞추어 보세요 : ")
3
4  # 스트라이크, 볼 검사
5  for index, value in enumerate(your_answer):
6      print("index : {0}, value : {1}".format(index, value))
```

→ **Stage 4 - 주민번호 인적사항 판별하기**

/week-2/stage-4