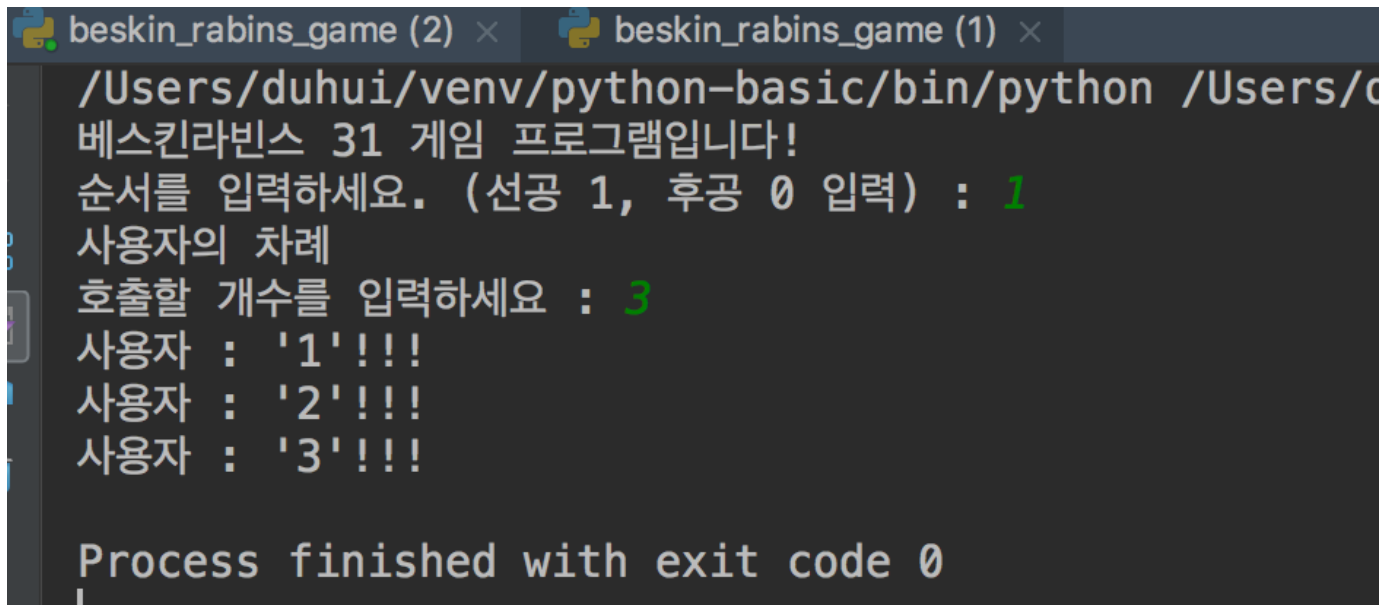


Stage 2 - 누가 먼저할지 순서정하기

사용자로부터 선공, 후공에 대한 정보를 입력받은 뒤 사용자의 차례와 컴퓨터의 차례를 구분하고, 사용자 차례 때의 시행을 제작합니다.

코딩 시작하기!

사용자로부터 선후에 대한 정보를 0과 1을 통해 받도록 합니다. 약간의 수학적 연산을 적용하여 순서를 구분합니다. 이어서 사용자의 차례 때 입력받은 숫자만큼 숫자를 출력해주는 시행을 구현합니다.



```

/Users/duhui/venv/python-basic/bin/python /Users/duhui/venv/python-basic/bin/python
베스킨라빈스 31 게임 프로그램입니다!
순서를 입력하세요. (선공 1, 후공 0 입력) : 1
사용자의 차례
호출할 개수를 입력하세요 : 3
사용자 : '1'!!!
사용자 : '2'!!!
사용자 : '3'!!!

Process finished with exit code 0
  
```













선공, 후공을 입력받고 사용자의 호출까지

순서 구분하기

순서란 어떻게 될까?

순서에 대해서 코딩하기 위해서 이 과정을 수학적으로 접근해야 합니다. 수학과 코딩이 우리의 사고력을 길러준다는 말이 이런 뜻이 아닐까 싶네요! 서로가 호출을 시행한다는 것은 시행을 하면 할수록 숫자가 커짐을 의미합니다. 그리고 그 수는 1씩 증가하죠. 그리고 증가하는 과정에서 특징이

있습니다. 홀수번과 짝수번으로 구분하여 시행하는 사람이 같다는 점이죠. 이를 그림으로 다시 이해해봅시다.

시행	1	2	3	4	5	6	7	8	9	10	11	12
시행자												

순서 이해하기

우리는 홀수와 짝수만 구분하면 선공과 후공을 정할 수 있다는 이야기가 됩니다. 그래서 사용자로 부터 0 또는 1을 입력받아 홀수를 대표하는 1을 입력할 시에는 선공을, 짝수를 대표하는 0을 입력할 시에는 후공이 됩니다.

수학적으로 다시보면, 시행을 2로 나눈 나머지는 언제나 0 또는 1이 되고요. 선공인 차례에는 그 나머지 값이 언제나 1을, 후공인 차례에는 0이 되어서 구분할 수 있습니다.

코딩하기

games/beskin_rabins_game.py

```

1  print("베스킨라빈스 31 게임 프로그램입니다!")
2
3  order = input('순서를 입력하세요. (선공 1, 후공 0 입력) : ')
4  order = int(order)
5
6  call = 0
7  count = 1
8
9  if count % 2 == order:
10     # 사용자의 차례
11     print('사용자의 차례')
12 else:
13     # 컴퓨터의 차례
14     print('컴퓨터의 차례')
```

- 사용자에게 0 또는 1을 입력받아서, 숫자로 형변환합니다.
- 2로 나눈 나머지와 입력받은 값을 비교하여 순서를 구분합니다.

번호 호출하기

range문 이해하기

우리는 이전까지 for문을 사용하면서, 리스트와 함께 enumerate문을 사용하였습니다. 이번에는 정수 자료형과 함께 사용할 수 있는 range문에 대해 알아보고 이것을 가지고 번호를 호출하겠습니다.

없으면 '0'

없으면 '1'

range(시작, 끝, 공차)

range의 구성

- 시작으로 설정된 숫자부터 값을 셉니다.
- 끝으로 설정된 숫자를 제외하고 그 전까지 값을 셉니다.
- 공차에 따라 시작에서부터 끝으로 값이 변합니다.

아래의 예문을 통해 이해해보겠습니다.

example

```
1  for value in range(10):
2      print(value)
3  # 0부터 9까지 출력
4  # 0 1 2 3 4 5 6 7 8 9
5
6  my_list = ['ab', 123, 43.123, 'I love you']
7  for index in range(len(my_list)):
8      print(index)
9  # 리스트의 길이가 4이므로 0부터 3까지 출력
10 # 0 1 2 3
11
12 for value in range(3, 9):
13     print(value)
14 # 3부터 8까지 출력
15 # 3 4 5 6 7 8
16
17 for value in range(2, 20, 2):
18     print(value)
```

```

19 # 2부터 시작하여 20 전까지 2단위로 출력
20 # 2 4 6 8 10 12 14 16 18
21
22 for value in range(10, 2, -1):
23     print(value)
24 # 10부터 시작하여 2로 감소하기 전까지 -1 단위로 출력
25 # 10 9 8 7 6 5 4 3

```

번호 호출하기

사용자의 호출과정에서 range문을 사용하여 호출을 시도합니다. 즉, 사용자로부터 몇 개를 출력할지 입력을 받고, 입력받은 크기만큼 range문과 반복문을 사용해서 출력해주면 됩니다.

games/beskin_rabins_game.py

```

1     size_of_call = input("호출할 개수를 입력하세요 : ")
2     size_of_call = int(size_of_call)
3
4     for _ in range(size_of_call):
5         call += 1
6         print("사용자 : '{0}'!!!".format(call))

```

 for문에서 변수의 자리에 언더바('_')가 들어가면 변수를 받지 않겠다는 뜻입니다.