

```
# 1-1, 5*8 배열 형성은 reshape로 쉽게 가능하다.
vector = np.random.randint(1, 101, 40)
print(vector)
array = vector.reshape((5, 8))
print(array)
# 1-2, 행 출력 방법
array_2, array_3 = array[1, :], array[2, :]
print(array_2, array_3)
# 1-3, 열 출력 방법 - 꺼내오는 방법
array_part = array[:, (0, 3, 6, 7)]
print(array_part)
# 1-4, 7번째 열의 평균, 분산
mean, variance = np.mean(array[:, 6]), np.var(array[:, 6])
print(array[:, 6], mean, variance)
'''
[98 77 33 91 26  7 20  3 61  8 20  3  5 85 50 34 67 83 68 98 36 82 35 56
 68 33 65 11 30 44 65 41 53 28 38 34 36 66 59 61]
[[98 77 33 91 26  7 20  3]
 [61  8 20  3  5 85 50 34]
 [67 83 68 98 36 82 35 56]
 [68 33 65 11 30 44 65 41]
 [53 28 38 34 36 66 59 61]]
[61  8 20  3  5 85 50 34] [67 83 68 98 36 82 35 56]
[[98 91 20  3]
 [61  3 50 34]
 [67 98 35 56]
 [68 11 65 41]
 [53 34 59 61]]
[20 50 35 65 59] 45.8 268.56

'''
```

```
# 2
# randint 함수 이용
number = np.random.randint(1, 21, 1)[0]
print(number)
if number >= 10:
    print("P")
else:
    print("NP")
'''
8
NP
'''
```

```

# 3
# randint 이용, 같은 방법
numbers = np.random.randint(1, 21, 8)
print(numbers)
for number in numbers:
    if number >= 10:
        print("P")
    else:
        print("NP")
'''
[ 9  5 11  9 18  1 16  8]
NP
NP
P
NP
P
NP
P
NP
'''

```

```
# 4
# sorted 내장 함수, slicing으로 배열 형성
numbers = [167, 171, 180, 173, 177, 164, 170, 183, 178, 172]
numbers = sorted(numbers)
print(numbers)
print([number for number in numbers if number >= 173])
'''
[173, 177, 178, 180, 183]

'''
```

```
# 5
# random.choice의 p= 값으로 이용
# replace=True
print(np.random.choice(8, 4, p=[0.1, 0.1, 0.1, 0.2, 0.1, 0.2, 0.1, 0.1], replace=True))
'''
[0 1 1 3]

'''
```

```
# 6
# 쉬운 조건문
for i in range(1, 101):
    if i % 3 == 0:
        print("Multiple of 3")
    else:
        print("Not multiple of 3")
'''
Not multiple of 3
Not multiple of 3
Multiple of 3
...
'''
```

```
# 7
# while i<1000 이용.
i = 1
while i < 1000:
    print(i)
    i *= 2
...
1
2
4
8
16
32
64
128
256
512
...
```