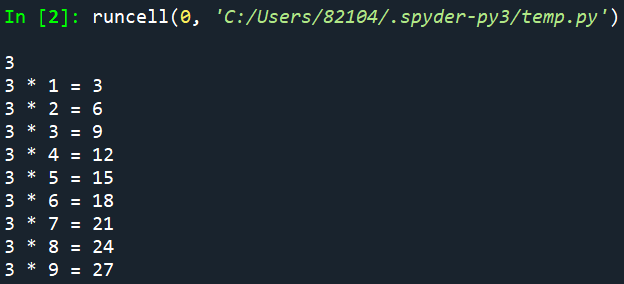
|  |  |  |  |
| --- | --- | --- | --- |
| **2주. Python Basics** | | | |
| 학번 | 32191197 | 이름 | 김채은 |

|  |
| --- |
| Q1. 화면에서 구구단의 단(2~9)를 입력받아 해당 단을 출력하는 프로그램을 작성하시오 |

**Source code :**

|  |
| --- |
| n = int(input())  for i in range(1, 10):  print("{} \* {} = {}".format(n, i, n\*i)) |

**실행화면 캡쳐:**

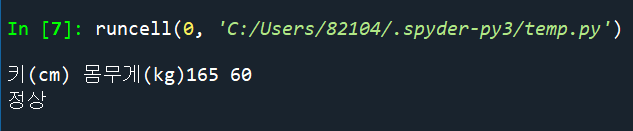


|  |
| --- |
| Q2. 비만도 체질량지수 (BMI)는 체중(kg단위) ÷ (키(m단위))2 로 계산을 한다.  (예: 키170cm, 몸무게 85kg 이면 BMI = 85 ÷ (1.7)2 이다.  BMI 값에 따라 다음과 같이 비만도를 판단한다.  18.5 이하: 저체중, 18.5~23: 정상, 23~25: 과체중, 25~30: 비만, 30이상: 고도비만  화면에서 키(kg)와 몸무게(cm)를 입력받아 BMI 값과 비만도를 출력하는 프로그램을 작성하시오. |

**Source code :**

|  |
| --- |
| height, weight = input("키(cm) 몸무게(kg)").split()  height = int(height)  weight = int(weight)  bmi = weight/((height/100)\*\*2)  if bmi < 18.5:  print("저체중")  elif bmi >= 18.5 and bmi<23:  print("정상")  elif bmi >= 23 and bmi <25:  print("과체중")  elif bmi >= 25 and bmi < 30:  print("비만")  elif bmi >= 30:  printf("고도비만") |

**실행화면 캡쳐:**

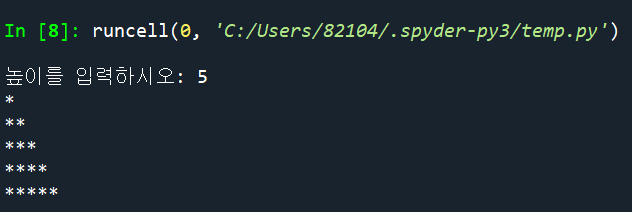


|  |  |
| --- | --- |
| Q3. 다음과 같이 높이를 입력하면 트리를 그리는 프로그램을 작성하시오   |  | | --- | | 높이를 입력하시오: 5  \*  \*\*  \*\*\*  \*\*\*\*  \*\*\*\*\* | |

**Source code :**

|  |
| --- |
| n = int(input("높이를 입력하시오: "))  for i in range(1, n+1):  for j in range(i):  print("\*", end="")  print() |

**실행화면 캡쳐:**

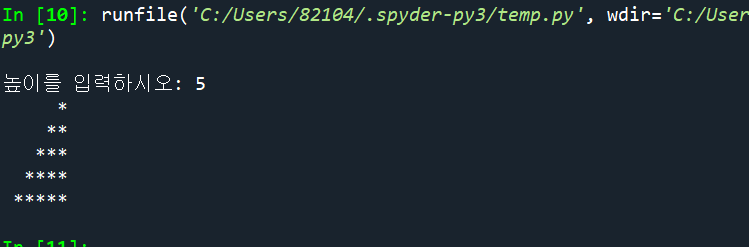


|  |  |
| --- | --- |
| Q4. 다음과 같이 높이를 입력하면 트리를 그리는 프로그램을 작성하시오. 단, 트리를 그리는 부분을 함수로 작성하여 이용하시오.   |  | | --- | | 높이를 입력하시오: 5  \*\*\*\*\*  \*\*\*\*\*  \*\*\*\*\*  \*\*\*\*\*  \*\*\*\*\* | |

**Source code :**

|  |
| --- |
| def draw\_tree(n):  for i in range(1, n+1):  for j in range(n-i, -1, -1):  print(" ", end="")  for j in range(i):  print("\*", end="")  print()      n = int(input("높이를 입력하시오: "))  draw\_tree(n) |

**실행화면 캡쳐:**

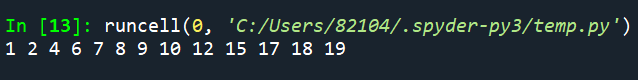


|  |
| --- |
| Q5. 배열 before\_arr 에는 다음과 같은 값이 저장되어 있다.  { 7, 1, 10, 4, 6, 9, 2, 8, 15, 12, 17, 19, 18 }  before\_arr 에 있는 값들을 오름차순으로 정렬하여 배열 after\_arr 에 저장하는 프로그램을 작성하시오. 단, 알고리즘은 다음과 같다.  1) before\_arr에서 가장 작은 값을 찾는다.  2) 그 값을 after\_arr 에 앞쪽에 넣는다.  3) before\_arr에서 찾은 가장 작은 값은 999 로 변경한다.  4) 1)~3) 과정을 배열의 길이만큼 반복한다. |

**Source code :**

|  |
| --- |
| before\_arr = [7, 1, 10, 4, 6, 9, 2, 8, 15, 12, 17, 19, 18]  after\_arr = []  mininum = before\_arr[0]  for i in range(len(before\_arr)):  mininum = before\_arr[0]  min\_idx = 0  for j in range(len(before\_arr)):  if mininum>before\_arr[j]:  mininum = before\_arr[j]  min\_idx = j  after\_arr.append(mininum)  before\_arr[min\_idx] = 999    for i in after\_arr:  print(i, end=" ") |

**실행화면 캡쳐:**

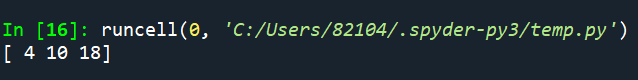


|  |
| --- |
| Q6. 길이가 같은 두 정수 배열에서 동일 인덱스 위치의 값을 곱하여 새로운 배열에 저장하여 반환하는 함수 mul\_arr()을 작성하고 제대로 작동하는지 테스트 하시오.  예) A = { 1,2,3 }, B = { 4,5,6 } 일 때 mul\_arr(A,B) 는 { 4,10,18 } 를 return 한다. |

**Source code :**

|  |
| --- |
| import numpy as np  def mul\_arr(a, b):  return np.multiply(a, b)  a = [1, 2, 3]  b = [4, 5, 6]  c = mul\_arr(a, b)  print(c) |

**실행화면 캡쳐:**

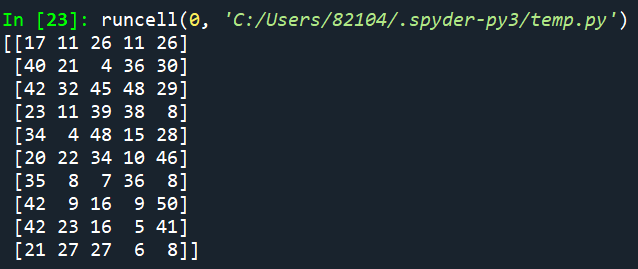


|  |
| --- |
| Q7. 다음과 같은 구조의 numpy array 인 **my\_arr** 를 생성하여 내용을 보이시오.  - 10행 5열의 구조  - array의 내용은 1~50 사이의 정수.  - 1~50의 정수로 1차원 array를 만든 뒤 reshap 으로 2차원 배열로 변형 |

**Source code :**

|  |
| --- |
| import numpy as np  import random  my\_arr = []  for i in range(50):  my\_arr.append(random.randint(1, 50))  my\_arr = np.array(my\_arr)  my\_arr = np.reshape(my\_arr, (10, 5))  print(my\_arr) |

**실행화면 캡쳐:**

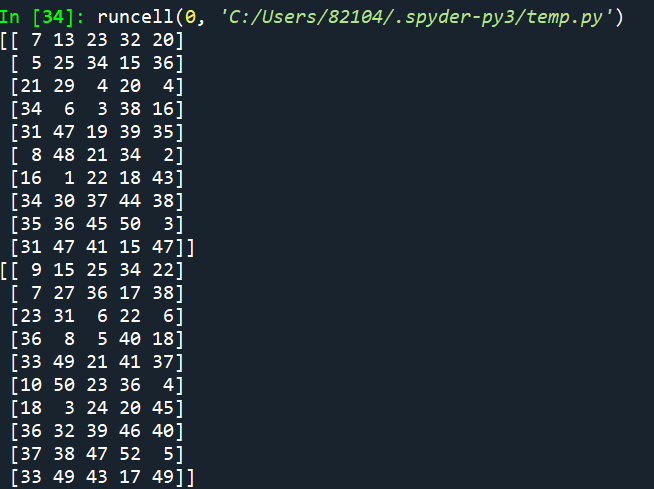


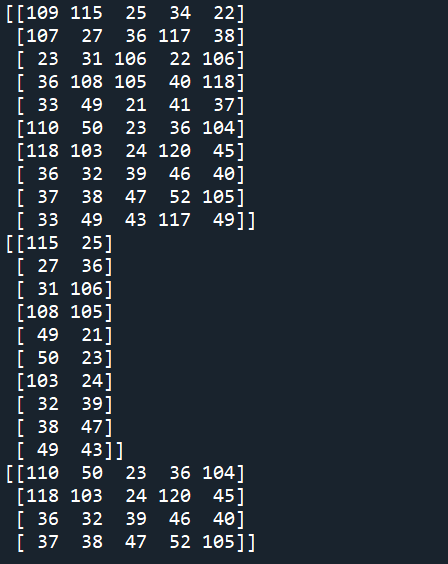
|  |
| --- |
| Q8. Q7에서 생성한 **my\_arr** 에 대해 다음의 문제를 해결하는 코드를 작성하고 실행 결과를 보이시오  (1) **my\_arr** 의 값들에 각각 2를 곱한 결과를 보이시오  (2) **my\_arr** 의 값들중 20 이하의 값들에 대해서만 100을 더한 후에 **my\_arr** 에 저장  하시오. **my\_arr** 의 내용을 보이시오  (3) **my\_arr** 에서 2,3열의 데이터만 잘라서 보이시오.  (4) **my\_arr** 에서 5~8행의 데이터만 잘라서 보이시오. |

**Source code :**

|  |
| --- |
| import numpy as np  import random  my\_arr = []  for i in range(50):  my\_arr.append(random.randint(1, 50))  my\_arr = np.array(my\_arr)  my\_arr = np.reshape(my\_arr, (10, 5))  print(my\_arr)  # 1  my\_arr = my\_arr + 2  print(my\_arr)  # 2  my\_arr = np.where(my\_arr <= 20, my\_arr+100, my\_arr)  print(my\_arr)  # 3  print(my\_arr[:, (1, 2)])  # 4  print(my\_arr[(5, 6, 7, 8), :]) |

**실행화면 캡쳐:**





|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Q9. slide 46의 그래프 작성 코드를 참조하여 다음의 월별 강수량을 그래프로 나타내시오   |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 1월 | 2월 | 3월 | 4월 | 5월 | 6월 | 7월 | 8월 | 9월 | 10월 | 11월 | 12월 | | 20 | 22 | 37 | 79 | 90 | 109 | 288 | 277 | 140 | 50 | 48 | 19 | |

**Source code :**

|  |
| --- |
| import numpy as np  import matplotlib.pyplot as plt  %matplotlib inline  month = np.arange(1, 13)  precipitation = np.array([20, 22, 37, 79, 90, 109, 288, 277, 140, 50, 48, 19])  plt.plot(month, precipitation)  plt.xlabel("month")  plt.ylabel("precipitation")  plt.show() |

**실행화면 캡쳐:**

