**REPORT**

**[ 과제(1) : LC정리(1) 문제풀이 ]**



|  |  |
| --- | --- |
| 과 목 명 | 파이썬과학프로그래밍기초 |
| 교 수 명 | 김 병 정 |
| 학 번 | 20237107 |
| 작 성 자 | 하 태 영 |
| 제 출 일 | 2025.04.05 |

**한림대학교**

초급 문제:

**문제 1:**

1부터 10까지의 숫자를 제곱한 리스트를 만드세요.

* 조건 :
  + 두가지 풀이방법 모두 작성하시오. (1줄 코드를 작성하시오.)
  + 방법1) LC 사용
* lt1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
  results = [kk\*kk for kk in lt1]  
  # lt1 요소를 제곱하여 results 리스트에 담는다.  
  print(results) # [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
  + 방법2) LC x
* lt1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
  results = []  
  for kk in lt1:  
   results.append(kk\*kk)  
  # lt1 리스트 요소를 제곱하여 results 리스트에 담는다.  
  print(results) # [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]

**문제 2:**

주어진 리스트에서 짝수만을 골라 새로운 리스트로 만드세요.

* numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
* 조건 :
  + 두가지 풀이방법 모두 작성하시오. (1줄 코드를 작성하시오.)
  + 방법1) LC 사용
* numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
  results = [kk for kk in numbers if kk % 2 == 0]  
  # numbers 리스트 요소 중에서 짝수만 results 리스트에 담는다.  
  print(results) # [2, 4, 6, 8, 10]
  + 방법2) LC x
* numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
  results = []  
  for kk in numbers:  
   if kk % 2 == 0:  
   results.append(kk)  
  # numbers 리스트 요소 중에서 짝수만 results 리스트에 담는다.  
  print(results) # [2, 4, 6, 8, 10]

**문제 3:**

'apple', 'banana', 'cherry' 리스트의 각 단어의 길이를 저장한 리스트를 만드세요.

* fruits = ['apple', 'banana', 'cherry']
* 조건 :
  + 두가지 풀이방법 모두 작성하시오. (1줄 코드를 작성하시오.)
  + 방법1) LC 사용
* fruits = ['apple', 'banana', 'cherry']  
  results = [len(kk) for kk in fruits]  
  # fruits 리스트 요소의 길이를 results 리스트에 담는다.  
  print(results) # [5, 6, 6]
  + 방법2) LC x
* fruits = ['apple', 'banana', 'cherry']  
  results = []  
  for kk in fruits:  
   results.append(len(kk))  
  # fruits 리스트 요소의 길이를 results 리스트에 담는다.  
  print(results) # [5, 6, 6]

**문제 4:**

0부터 20까지 숫자 중에서 5의 배수인 숫자만 리스트로 만드세요.

* 조건 :
  + 두가지 풀이방법 모두 작성하시오. (1줄 코드를 작성하시오.)
  + 방법1) LC 사용
* results = [kk for kk in range(21) if kk % 5 == 0]  
  # 0~20까지 5의 배수를 results 리스트에 담는다.  
  print(results) # [0, 5, 10, 15, 20]
  + 방법2) LC x
* results = []  
  for kk in range(21):  
   if kk % 5 == 0:  
   results.append(kk)  
  # 0~20까지 5의 배수를 results 리스트에 담는다.  
  print(results) # [0, 5, 10, 15, 20]

**문제 5:**

리스트 numbers의 모든 숫자에 2를 더한 값을 새로운 리스트에 저장하세요.

* numbers = [1, 2, 3, 4, 5]
* 조건 :
  + 두가지 풀이방법 모두 작성하시오. (1줄 코드를 작성하시오.)
  + 방법1) LC 사용
* numbers = [1, 2, 3, 4, 5]  
  results = [kk+2 for kk in numbers]  
  # numbers 리스트 요소에 2를 더하여 results 리스트 요소에 담는다.  
  print(results) # [3, 4, 5, 6, 7]
  + 방법2) LC x
* numbers = [1, 2, 3, 4, 5]  
  results = []  
  for kk in numbers:  
   results.append(kk+2)  
  # numbers 리스트 요소에 2를 더하여 results 리스트 요소에 담는다.  
  print(results) # [3, 4, 5, 6, 7]

**문제 6:**

주어진 리스트에서 각 숫자를 두 배로 만든 리스트를 만드세요.

* numbers = [3, 6, 9, 12]
* 조건 :
  + 두가지 풀이방법 모두 작성하시오. (1줄 코드를 작성하시오.)
  + 방법1) LC 사용
* numbers = [3, 6, 9, 12]  
  results = [kk\*2 for kk in numbers]  
  # numbers 리스트 요소에 2를 곱하고 results 리스트에 담는다.  
  print(results) # [6, 12, 18, 24]
  + 방법2) LC x
* numbers = [3, 6, 9, 12]  
  results = []  
  for kk in numbers:  
   results.append(kk\*2)  
  # numbers 리스트 요소에 2를 곱하고 results 리스트에 담는다.  
  print(results) # [6, 12, 18, 24]

**중급 문제:**

**문제 1:**

1부터 20까지의 숫자 중에서 소수만 리스트로 만드세요.

* 조건 :
  + 두가지 풀이방법 모두 작성하시오.
  + 방법1) LC 사용
* results = []  
  for kk in range(2, 21):  
   if (kk % i != 0 for i in range(2, int(kk \*\* 0.5) + 1)):  
   results.append(kk)  
  # 2~20까지의 요소를 2~kk의 제곱근까지 반복하고, 소수까지 확인 후 results 리스트에 담는다.  
  print(results) # [2, 3, 5, 7, 11, 13, 17, 19]
  + 방법2) LC x (사용자정의함수 사용)
* numbers = [kk for kk in range(2, 21)  
   if all(kk % i != 0 for i in range(2, int(kk \*\* 0.5) + 1))]  
  # 2~20까지의 요소를 2~kk의 제곱근까지 반복하고, 소수까지 확인 후 results 리스트에 담는다.  
  print(numbers) # [2, 3, 5, 7, 11, 13, 17, 19]

**문제 2:**

리스트에 있는 단어들 중에서 첫 글자가 'a'로 시작하는 단어만을 고르세요.

* words = ['apple', 'banana', 'avocado', 'grape', 'apricot']
* 조건 :
* 두가지 풀이방법 모두 작성하시오. (1줄 코드를 작성하시오.)
* 방법1) LC 사용
* words = ['apple', 'banana', 'avocado', 'grape', 'apricot']  
  results = [kk for kk in words if(kk[0] == 'a')]  
  # words 리스트 요소의 첫글자가 a인 단어를 results 리스트 요소에 담는다.  
  print(results) # ['apple', 'avocado', 'apricot']
* 방법2) LC x
* words = ['apple', 'banana', 'avocado', 'grape', 'apricot']  
  results = []  
  for kk in words:  
   if(kk[0] == 'a'):  
   results.append(kk)  
  # words 리스트 요소의 첫글자가 a인 단어를 results 리스트 요소에 담는다.  
  print(results) # ['apple', 'avocado', 'apricot']

**문제 3:**

두 개의 리스트에서 대응되는 값을 더한 값을 새로운 리스트에 저장하세요.

* list1 = [1, 2, 3] list2 = [4, 5, 6]
* 조건 :
* 두가지 풀이방법 모두 작성하시오. (1줄 코드를 작성하시오.)
* 방법1) LC 사용
* list1 = [1, 2, 3]  
  list2 = [4, 5, 6]  
  results = []  
  for a, b in zip(list1, list2):  
   results.append(a+b)  
  # zip 함수로 두 리스트를 묶어서 각 요소를 짝지어 a, b에 할당하고, a+b를 results 리스트에 추가  
  print(results) # [5, 7, 9]
* 방법2) LC x
* list1 = [1, 2, 3]  
  list2 = [4, 5, 6]  
  results = [a+b for a, b in zip(list1, list2)]  
  # zip 함수로 두 리스트를 묶어서 각 요소를 짝지어 a, b에 할당하고, a+b를 results 리스트에 추가  
  print(results) # [5, 7, 9]

**문제 4:**

주어진 리스트에서 길이가 5 이상인 단어들만 골라 리스트로 만드세요.

* words = ['table', 'apple', 'pen', 'bottle', 'cup']
* 조건 :
* 두가지 풀이방법 모두 작성하시오. (1줄 코드를 작성하시오.)
* 방법1) LC 사용
* words = ['table', 'apple', 'pen', 'bottle', 'cup']  
  results = [kk for kk in words if(len(kk) >= 5)]  
  # words 리스트의 요소의 길이가 5이상 이면 results 리스트에 담는다.  
  print(results) # ['table', 'apple', 'bottle']
* 방법2) LC x
* words = ['table', 'apple', 'pen', 'bottle', 'cup']  
  results = []  
  for kk in words:  
   if(len(kk) >= 5):  
   results.append(kk)  
  # words 리스트의 요소의 길이가 5이상 이면 results 리스트에 담는다.  
  print(results) # ['table', 'apple', 'bottle']

**문제 5:**

두 개의 리스트에서 교집합을 찾아 리스트로 만드세요.

* list1 = [1, 2, 3, 4, 5] list2 = [3, 4, 5, 6, 7]
* 조건 :
* 두가지 풀이방법 모두 작성하시오. (1줄 코드를 작성하시오.)
* 방법1) LC 사용
* list1 = [1, 2, 3, 4, 5]  
  list2 = [3, 4, 5, 6, 7]  
  results = [kk for kk in list1 if kk in list2]  
  # list1에서 list2에 존재하는 원소가 같으면 results 리스트에 담는다.  
  print(results) # [3, 4, 5]
* 방법2) LC x
* list1 = [1, 2, 3, 4, 5]  
  list2 = [3, 4, 5, 6, 7]  
  results = []  
  for kk in list1:  
   if kk in list2:  
   results.append(kk)  
  # list1에서 list2에 존재하는 원소가 같으면 results 리스트에 담는다.  
  print(results) # [3, 4, 5]

**문제 6:**

주어진 2차원 리스트의 각 행에서 최대값을 찾아 리스트로 만드세요.

* matrix = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
* 조건 :
* 두가지 풀이방법 모두 작성하시오. (1줄 코드를 작성하시오.)
* 방법1) LC 사용
* matrix = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]  
  results = [max(kk) for kk in matrix]  
  # matrix 리스트 요소에서 가장 큰 요소를 results 리스트에 담는다.  
  print(results) # [3, 6, 9]
* 방법2) LC x
* matrix = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]  
  results = []  
  for kk in matrix:  
   results.append(max(kk))  
  # matrix 리스트 요소에서 가장 큰 요소를 results 리스트에 담는다.  
  print(results) # [3, 6, 9]

**문제 7:**

주어진 문자열에서 각 단어를 거꾸로 뒤집고, 단어의 순서는 그대로 유지한 리스트를 만드세요.

* sentence = "List comprehension is powerful"
* 조건 :
* 방법1) LC 사용
* sentence = "List comprehension is powerful"  
  words = []  
  for word in sentence.split():  
   words.append(word[::-1])  
  results = ' '.join(words)  
  # sentence 문자열을 공백 기준으로 나누고 각 단어를 뒤집어서 words 리스트에 추가  
  print(results) # tsiL noisneherpmoc si lufrewop
* 방법2) LC x
* sentence = "List comprehension is powerful"  
  words = [word[::-1] for word in sentence.split()]  
  results = ' '.join(words)  
  # sentence 문자열을 공백 기준으로 나누고 각 단어를 뒤집어서 words 리스트에 추가  
  print(results) # tsiL noisneherpmoc si lufrewop

**문제 8:**

두 개의 리스트에서 각 쌍의 곱셈 결과 중에서 짝수만을 선택한 리스트를 만드세요.

* list1 = [1, 2, 3, 4, 5] list2 = [6, 7, 8, 9, 10]
* 조건 :
* 방법1) LC 사용
* list1 = [1, 2, 3, 4, 5]  
  list2 = [6, 7, 8, 9, 10]  
  results = []  
  for a, b in zip(list1, list2):  
   if(a \* b) % 2 == 0:  
   results.append(a \* b)  
  # zip 함수로 두 리스트를 묶어서 각 요소를 짝지어 a, b에 할당하고, a\*b가 짝수인 경우만 results 리스트에 추가  
  print(results) # [6, 14, 24, 36, 50]
* 방법2) LC x
* list1 = [1, 2, 3, 4, 5]  
  list2 = [6, 7, 8, 9, 10]  
  results = [a \* b for a, b in zip(list1, list2) if(a \* b) % 2 == 0]  
  # zip 함수로 두 리스트를 묶어서 각 요소를 짝지어 a, b에 할당하고, a\*b가 짝수인 경우만 results 리스트에 추가  
  print(results) # [6, 14, 24, 36, 50]

**고급 문제:**

**문제 1:**

1부터 100까지의 숫자 중에서 각 숫자의 자릿수 합이 짝수인 숫자들만 고르세요.

* 조건 :
* 두가지 풀이방법 모두 작성하시오.
* 방법1) LC 사용
* results = [kk for kk in range(1, 101)  
   if sum(int(digit) for digit in str(kk)) % 2 == 0]  
  # 1~100에서 자릿수 합이 짝수인지 확인하고, results 리스트에 담는다.  
  # digit 숫자의 각 자릿수를 의미하는 변수명  
  print(results) # [2, 4, 6, 8, 11, 13, 15, 17, 19, 20, 22, 24, 26, 28, 31, 33, 35, 37, 39, 40, 42, 44, 46, 48, 51, 53, 55, 57, 59, 60, 62, 64, 66, 68, 71, 73, 75, 77, 79, 80, 82, 84, 86, 88, 91, 93, 95, 97, 99]
* 방법2) LC x (사용자정의함수 사용)
* results = []  
  for kk in range(1, 101):  
   if sum(int(digit) for digit in str(kk)) % 2 == 0:  
   results.append(kk)  
  # 1~100에서 자릿수 합이 짝수인지 확인하고, results 리스트에 담는다.  
  # digit 숫자의 각 자릿수를 의미하는 변수명  
  print(results) # [2, 4, 6, 8, 11, 13, 15, 17, 19, 20, 22, 24, 26, 28, 31, 33, 35, 37, 39, 40, 42, 44, 46, 48, 51, 53, 55, 57, 59, 60, 62, 64, 66, 68, 71, 73, 75, 77, 79, 80, 82, 84, 86, 88, 91, 93, 95, 97, 99]

**문제 2:**

문자열을 받아서, 각 문자의 아스키 값을 리스트로 만드세요.

* text = "Hello"
* 조건 :
* 두가지 풀이방법 모두 작성하시오. (1줄 코드를 작성하시오.)
* 방법1) LC 사용
* text = "Hello"  
  results = [ord(kk) for kk in text]  
  # text 문자열을 유니코드 코드 포인트를 정수로 반환하여 results 리스트에 담는다.  
  # ord() 함수는 파이썬에서 문자의 유니코드 코드 포인트를 반환하는 내장 함수  
  print(results) # [72, 101, 108, 108, 111]
* 방법2) LC x
* text = "Hello"  
  results = []  
  for kk in text:  
   results.append(ord(kk))  
  # text 문자열을 유니코드 코드 포인트를 정수로 반환하여 results 리스트에 담는다.  
  # ord() 함수는 파이썬에서 문자의 유니코드 코드 포인트를 반환하는 내장 함수  
  print(results) # [72, 101, 108, 108, 111]

**문제 3:**

주어진 리스트에서 중복된 숫자를 제거하고, 중복이 없는 리스트를 만드세요.

* numbers = [1, 2, 2, 3, 4, 4, 5]
* 조건 :
* 두가지 풀이방법 모두 작성하시오.
  + 방법1) LC 사용
* numbers = [1, 2, 2, 3, 4, 4, 5]  
  results = []  
  [results.append(num) for num in numbers if num not in results]  
  # number 리스트 요소 중에 results 리스트에 없으면 results 리스트에 담는다.  
  print(results) # [1, 2, 3, 4, 5]
  + 방법2) LC x (사용자정의함수 사용)
* def remove\_duplicates(nums):  
   return list(set(nums))  
  # 집합 함수로 중복을 제거해주고, 리스트 형변환해준다.  
  numbers = [1, 2, 2, 3, 4, 4, 5]  
  results = remove\_duplicates(numbers)  
  print(results) # [1, 2, 3, 4, 5]
  + 방법3) LC 사용 + 슬라이싱 이용
* numbers = [1, 2, 2, 3, 4, 4, 5]  
  results = []  
  [results.append(num) for num in numbers if num not in results[::]]  
  # numbers 리스트에서 각 숫자를 하나씩 확인하고, 중복되지 않은 숫자만 results 리스트에 담는다.  
  print(results) # [1, 2, 3, 4, 5]

**문제 4:**

다음 프로그램의 출력결과를 작성하고, 설명하시오.

* numbers = [1, 2, 2, 3, 4, 4, 5] unique\_numbers = [] [unique\_numbers.append(x) for x in numbers if x not in unique\_numbers]
* numbers = [1, 2, 2, 3, 4, 4, 5]  
  unique\_numbers = [] # 중복되지 않는 값을 저장할 리스트  
  [unique\_numbers.append(x) for x in numbers if x not in unique\_numbers]  
  # 1. for문 numbers 리스트의 각 요소 x에 대해 반복한다.  
  # 2. if문 x가 unique\_numbers 리스트에 없으면 그 값을 unique\_numbers에 추가한다.  
  # 3. append 조건을 만족하면 해당 값을 unique\_numbers 리스트에 추가합니다.  
  print(unique\_numbers) # [1, 2, 3, 4, 5]

**문제 5:**

주어진 리스트에서 중복된 숫자를 제거하고, 중복이 없는 리스트를 만드세요.

* numbers = [5, 4, 3, 3, 4, 4, 5]
* 조건 :
* 순서를 유지하시오.
* [5, 4, 3] #출력
* numbers = [5, 4, 3, 3, 4, 4, 5]  
  results = []  
  [results.append(kk) for kk in numbers if kk not in results]  
  # number 리스트 요소가 results 리스트 요소에 존재하지 않는 요소들을 results 리스트에 추가한다.  
  print(results) # [5, 4, 3]

**문제 6:**

피보나치 수열의 첫 20개 숫자를 구하고, 그 중에서 짝수만 리스트로 만드세요.

* [0, 2, 8, 34, 144, 610, 2584] #출력
* 조건 :
* 방법1) LC 사용
* fib = [0, 1]  
  for \_ in range(2, 20):  
   fib.append(fib[-1] + fib[-2])  
  # 2~20까지 피보나치 수열의 규칙에 따라 현재 수열의 마지막 두 숫자를 더하여 새로운 수를 fib 리스트에 추가합니다.  
  results = [kk for kk in fib if kk % 2 == 0]  
  # fib 리스트에서 짝수인 경우만 results 리스트에 담는다.  
  print(results) # [0, 2, 8, 34, 144, 610, 2584]
* 방법2) LC x
* fib = [0, 1]  
  for \_ in range(2, 20):  
   fib.append(fib[-1] + fib[-2])  
  # 2~20까지 피보나치 수열의 규칙에 따라 현재 수열의 마지막 두 숫자를 더하여 새로운 수를 fib 리스트에 추가합니다.  
  results = []  
  for kk in fib:  
   if kk % 2 == 0:  
   results.append(kk)  
  # fib 리스트에서 짝수인 경우만 results 리스트에 담는다.  
  print(results) # [0, 2, 8, 34, 144, 610, 2584]

**문제 7:**

주어진 2차원 리스트에서 대각선 요소만 추출하여 리스트로 만드세요.

* matrix = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
* [1, 5, 9] #출력
* 조건 :
* 두가지 풀이방법 모두 작성하시오. (1줄 코드를 작성하시오.)
* 방법1) LC 사용
* matrix = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]  
  results = [matrix[kk][kk] for kk in range(len(matrix))]  
  # matrix 리스트 길이만큼 같은 행과 열을 results 리스트에 담는다.  
  print(results) # [1, 5, 9]
* 방법2) LC x
* matrix = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]  
  results = []  
  for kk in range(len(matrix)):  
   results.append(matrix[kk][kk])  
  # matrix 리스트 길이만큼 같은 행과 열을 results 리스트에 담는다.  
  print(results) # [1, 5, 9]

**문제 8:**

주어진 리스트에서 각각의 요소가 소수인지 여부를 판단하는 불리언 리스트를 만드세요.

* numbers = [10, 15, 7, 19, 22]
* 조건 :
* 방법1) LC 사용
* def is\_prime(n):  
   if n <= 1:  
   return False  
   for kk in range(2, int(n \*\* 0.5) + 1):  
   if n % kk == 0:  
   return False  
   return True  
  # 1 이하면 False, 2 ~ n의 제곱근까지 n은 kk로 나눠지면 false 아닌 경우 True  
  numbers = [10, 15, 7, 19, 22]  
  results = [is\_prime(kk) for kk in numbers]  
  # number 리스트에 있는 요소들을 is\_prime() 처리된 소수를 results 리스트에 담는다.  
  print(results) # [False, False, True, True, False]
* 방법2) LC x
* # LC X  
  def is\_prime(n):  
   if n <= 1:  
   return False  
   for kk in range(2, int(n \*\* 0.5) + 1):  
   if n % kk == 0:  
   return False  
   return True  
  # 1 이하면 False, 2 ~ n의 제곱근까지 n은 kk로 나눠지면 false 아닌 경우 True  
  numbers = [10, 15, 7, 19, 22]  
  results = []  
  for kk in numbers:  
   results.append(is\_prime(kk))  
  # number 리스트에 있는 요소들을 is\_prime() 처리된 소수를 results 리스트에 담는다.  
  print(results) # [False, False, True, True, False]