# 16차 연습문항\_답안

## **파이썬 기본문항: 람다표현식**

|  |
| --- |
| 수강생 공지 사항   * 제출은 프로그래머스를 통해 해주시기 바랍니다([링크](https://campus.programmers.co.kr/app/courses/24550/curriculum)). * 파일명은 아래와 같은 형식으로 제출해주세요   + 교육생번호\_이름\_교과목\_문항\_N차시.pdf   ex) DR-11111\_홍길동\_파이썬\_연습문항\_1차시.pdf  ex) DR-11111\_홍길동\_파이썬\_추가문항\_1차시.pdf   * 답은 "write your answer"에 적어주세요. 다만 코딩 문제의 경우 output까지 답에 포함시켜 주세요.   ex)     * 답은 캡쳐를 하셔도 되고 텍스트로 넣으셔도 됩니다. * 마감 기한은 문제가 나간 주 **일요일 23:59까지**입니다. |

### 다음 중 파이썬의 람다 함수에 대한 설명으로 틀린 것은 무엇인가?

a. 람다 함수는 한 줄로 작성할 수 있는 익명 함수이다.

b. 람다 함수는 lambda 키워드를 사용하여 정의된다.

c. 람다 함수는 여러 줄로 작성할 수 있다.

d. 람다 함수는 함수의 인수로 전달할 때 유용하다.

e. 람다 함수는 표현식의 결과가 자동으로 반환된다.

**답**

|  |
| --- |
| C. 람다 함수는 한 줄로 작성할 수 있는 간단한 함수로, 여러 줄로 작성할 수 없습니다. |

### 아래의 람다 함수의 코드를 실행했을 때 나타난 결과에 대해 설명해 보시오.

|  |
| --- |
| Example Code |
| increment = lambda x: x + 1  print(increment) |

**답**

|  |
| --- |
| <function <lambda> at 0x7f8b4c9e1f70>  increment 변수는 람다 함수 객체 자체를 참조합니다. 따라서 increment를 출력하면 람다 함수 객체의 메모리 주소가 출력됩니다. |

### 입력받는 정수가 짝수일 때는 제곱하고 홀수일 때는 그대로 두는 조건문을 가진 람다함수를 만들고, 임의의 정수를 테스트하고 결과를 출력하시오.

**답**

|  |
| --- |
| process\_numbers = lambda x: (x\*\*2 if x % 2 == 0 else x)    num = process\_numbers(4)  print(num) # 16 |

### map 함수는 파이썬의 내장 함수로, 하나 이상의 시퀀스(ex: 리스트, 튜플 등)의 각 요소에 대해 지정된 함수를 적용하고, 그 결과를 담은 맵 객체(map object)를 반환합니다. (ex: map(function, iterable,,,) 문제3에서 만들어진 람다 함수와 제공된 리스트를 map 함수의 인자로 적용하는 코드를 작성해 보시오.

제공된 리스트: [1, 2, 3, 4, 5]

**답**

|  |
| --- |
| # 람다 함수 정의  process\_numbers = lambda x: (x\*\*2 if x % 2 == 0 else x)  # 주어진 리스트  numbers = [1, 2, 3, 4, 5]    # map 함수를 사용하여 람다 함수를 numbers 리스트에 적용  result = map(process\_numbers, numbers)    # map 객체를 리스트로 변환하여 출력  print(list(result)) # 출력: [1, 4, 3, 16, 5] |

### 아래의 코드를 람다함수의 형식으로 변환하시오

<주의사항>

- 람다함수에서 사용하는 조건부 표현식에는 :(콜론)을 사용하지 않음

- 람다함수에서 if를 사용하면 반드시 else를 사용해야 함

- 람다함수에서 elif는 사용하지 못함

**답**

|  |
| --- |
| numbers = [2, 3, 4, 5, 6, 7]  result = map(lambda x: x if x <3 else (x \* 2 if x <6 else x \* 3), numbers)  print(list(result)) |

### 아래 두 리스트의 각 요소를 조건에 따라 계산하는 일반 함수를 작성하시오

- 조건: list1의 요소가 짝수이면 list2의 요소를 더하고, 그렇지 않으면 곱함

- 결과를 출력하시오.

list1 = [1, 2, 3, 4, 5]

list2 = [10, 20, 30, 40, 50]

**답**

|  |
| --- |
| def calculate\_elements(list1, list2):        result = []        for x, y in zip(list1, list2):              if x % 2 == 0:                    result.append(y + x)              else:                    result.append(y \* x)        return result  # 두 개의 리스트  list1 = [1, 2, 3, 4, 5]  list2 = [10, 20, 30, 40, 50]  # 함수 호출 및 결과 출력  result = calculate\_elements(list1, list2)  print(result) |

### 문제6의 함수를 람다 함수로 변환해서 프로그램을 실행해 보시오

**답**

|  |
| --- |
| # 두 개의 리스트  list1 = [1, 2, 3, 4, 5]  list2 = [10, 20, 30, 40, 50]  # 두 리스트의 각 요소를 조건에 따라 계산하는 람다 함수  # 조건: list1의 요소가 짝수이면 list2의 요소를 더하고, 그렇지 않으면 곱함  result = map(lambda x, y: y + x if x % 2 == 0 else y \* x, list1, list2)    # 결과 출력  print(list(result)) |

### 아래 리스트에 있는 문자열 중 ‘a'를 포함하는 단어를 걸러내는 프로그램을 filter 함수를 사용하여 작성하시오

words = ["apple", "banana", "cherry", "date", "elderberry", "fig", "grape"]

**답**

|  |
| --- |
| words = ["apple", "banana", "cherry", "date", "elderberry", "fig", "grape"]    def contains\_a(word):        return 'a' in word    # contains\_a 함수를 사용한 filter 예제  words\_with\_a = list(filter(contains\_a, words))    print(words\_with\_a) # 출력: ['apple', 'banana', 'date', 'grape'] |

### 문제8에서 작성된 프로그램을 람다 함수를 이용하는 프로그램으로 변화해서 실행해 보시오.

**답**

|  |
| --- |
| words = ["apple", "banana", "cherry", "date", "elderberry", "fig", "grape"]    # 'a'를 포함하는 단어를 걸러내는 람다 함수를 사용한 filter 예제  words\_with\_a = list(filter(lambda word: 'a' in word, words))    print(words\_with\_a) # 출력: ['apple', 'banana', 'grape'] |

### 정수 하나를 입력 받아서 소수인지 판별하는 프로그램을 작성하시오

**답**

|  |
| --- |
| def is\_prime(num):      if num <= 1:          return False      if num <= 3:          return True      if num % 2 == 0 or num % 3 == 0:          return False      i = 5      while i \* i <= num:          if num % i == 0 or num % (i + 2) == 0:              return False          i += 6      return True  # 테스트  num = int(input('소수인지 확인할 정수를 입력하세요 >> '))  print(is\_prime(num)) |

### 문제10의 함수를 람다 함수로 변환하여 간단히 만드는게 적합한지 서술하시오

**답**

|  |
| --- |
| 소수를 판별하는 함수를 람다 함수로 바꾸는 것은 적합하지 않습니다. 람다 함수는 보통 간단하고 한 줄로 표현할 수 있는 함수에 사용됩니다. 하지만 소수를 판별하는 로직은 일반적으로 한 줄로 간단히 표현하기 어렵고, 여러 조건과 반복문이 필요하기 때문에 람다 함수로 충분히 표현하기 어렵습니다. |