1문제)

텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

풀이 및 답)

**print(aTuple[::-1])**

슬라이싱할 때 **[start : stop : step]**인 것을 6장에서 배웠었다.

이는 리스트뿐만 아니라 튜플에도 적용된다.

따라서 변수를 하나 다시 선언해서 슬라이싱을 하거나 이렇게 바로 출력할 수 있다.

한 줄로 간결하게 출력하고 싶어서 이렇게 작성했다.

텍스트, 폰트, 스크린샷, 영수증이(가) 표시된 사진

자동 생성된 설명

풀이 및 답)

**(b), (d)**

딕셔너리에서 키를 통해 value를 얻는 방법은 3가지가 있다.

1. 대괄호[]를 이용하는 것이다.
2. .get() 메서드를 이용하는 것이다.
3. .values() 메서드를 통해 모든 value를 가져온다.

텍스트, 폰트, 스크린샷, 화이트이(가) 표시된 사진

자동 생성된 설명

풀이 및 답)

**(b)**

위 문제는 중첩 딕셔너리에 대해 다루는 문제이다.

**딕셔너리 = {키1: {키A: 값A}, 키2: {키B: 값B}}** 일 때

**딕셔너리[키][키]**로 접근하기 때문에

**student[1][“age”]**가 정답이 된다.

텍스트, 폰트, 화이트이(가) 표시된 사진

자동 생성된 설명

풀이 및 답)

**student.pop("score")**

**del student[“score”]**

pop()과 del은 똑같이 딕셔너리를 지우는 기능을 하지만 약간의 차이가 있다.

pop()의 경우 지워진 인덱스의 값을 반환하지만 del은 반환하지 않는다.

**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명**

그러므로 print(removed)의 결과는 89가 나오게 된다.

이렇기 때문에 del이 pop()보다 미세하게 더 빠르다.

**텍스트, 폰트, 스크린샷, 화이트이(가) 표시된 사진

자동 생성된 설명**

풀이 및 답)

**True**

**False**

set2는 set1의 value(값)을 모두 가지고 있으므로 set1은 set2의 부분집합이다.

set1은 set2의 value(값)을 모두 가지고 있지 않으므로 set2는 set1의 부분집합이 아니다.

2문제)

텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

자동 생성된 설명

텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

소스코드)

텍스트, 폰트, 스크린샷, 시계이(가) 표시된 사진

자동 생성된 설명

실행화면)

텍스트, 폰트, 멀티미디어 소프트웨어, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

풀이)

텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

**a = {x : x\*x for x in range(1, 11)}**

a는 딕셔너리 x: x\*x는 출력 수식이다.

x\*x와 x\*\*2는 같은 식이기 때문에 x\*x 대신 x\*\*2를 써도 무방하다.

물론 for문을 따로 써서 풀어서 써도 되겠지만 수업시간에 배운 함축 코드를 사용해보고 싶어 다음과 같이 코딩했다.

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

소스코드)

텍스트, 스크린샷, 폰트, 시계이(가) 표시된 사진

자동 생성된 설명

실행화면)

텍스트, 폰트, 멀티미디어 소프트웨어, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

풀이)

문제1에서 썼던 것과 같이 .values() 매서드를 통해 딕셔너리의 모든 value(값)들을 가져올 수 있다. 한 번에 쓰지 않고 변수에 myDict.values()를 집어넣어준 후에 sum() 메서드로 총 합을 구해도 무방하다. 만약 hap = myDict.values()를 하고 hap을 출력했다면 dict\_values([100, 2000, 320])이 출력되었을 것이다.

텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

소스코드)

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

실행화면)

텍스트, 스크린샷, 멀티미디어 소프트웨어, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

풀이)

먼저 이 문제의 실행결과가 잘못된 것 같다.

모두 포함된 글자가 ‘!’도 있고 띄어쓰기도 있고 ‘l’도 있는데 H w o e만 있는 게 좀 이상하다. 그래서 처음에는 ‘글자’라서 띄어쓰기랑 문장부호는 빼는 건가 싶었는데 알파벳 l도 공통 되어있는데 이건 왜 출력되지 않은거지.. 싶어서 문제 오류를 생각해봤다. 게다가 출력을 받을 때 두개다 ‘첫 번째 문자열’인 것을 보고 확신했다.

먼저 입력 받은 것을 세트로 형 변환시켜주고 &를 이용해 교집합을 comm 변수에 넣어주고 comm변수를 출력해준다.

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

소스코드)

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

실행화면)

텍스트, 스크린샷, 멀티미디어 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명

풀이)

콤마 띄어쓰기로 구분해서 입력 받은 것을 잘라서 세트로 형 변환하고 교집합과 전체집합을 구한 뒤 전체 집합해서 교집합을 빼서 한쪽에만 있는 요소를 정렬해서 출력하였다. 조금이라도 다른 방식으로 입력하면 결과값이 모두 망가지기 때문에 콤마 띄어쓰기를 해서 입력해 달라고 안내하는 문장을 출력하면 더 좋겠지만 거기까지 처리하지는 않았다.

텍스트, 폰트, 라인, 스크린샷이(가) 표시된 사진

자동 생성된 설명

소스코드)

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

실행화면)

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

풀이)

한번에 함축 표현들을 쓰는데 아직 그렇게 친숙하지 않은 것 같아 먼저 다 풀어서 짜본 뒤에 조건에 맞게 함축 표현을 써보았다. 아직 바로바로 함축표현이 나오지 않아 더 연습을 해야 될 것 같다. 주석으로 자세하게 풀이했다. 먼저 교수님이 강의 해주신대로 나중에 많이 쓸 만한 기능들은 함수로 빼는 연습을 해보았다. 그래서 문자열을 분석해주는 analyze 함수를 정의했고 .isalpha(), .isnumeric() 메서드를 이용해 알파벳인지 숫자인지 구분을 하고 카운트한 뒤에 리턴 값으로 딕셔너리를 이용해 넘겼다. 그리고 마지막으로 결과를 출력하였다.