

예제

1. 어느 식당의 메뉴와 가격은 돈가스 5,000원, 생선가스 5,500원, 우동 2,500원, 초밥 세트 9,000원이다. 이 메뉴들을 빈 dictionary를 생성한 후, 딕셔너리 자료구조를 이용하여 저장해보고 각각의 메뉴와 가격을 순차적으로 출력하는 프로그램을 작성하여 보자.

실행결과

```
===== MENU =====  
[('돈가스', 5000), ('생선가스', 5500), ('우동', 2500), ('초밥세트', 9000)]
```

예제

2. 어느 커피숍에는 메뉴가 4가지 있다. "아메리카노, 카페라떼, 그린티라떼, 모카라떼" 각 메뉴의 가격은 2,000원, 2,500원, 3,000원, 3,500원이다. 이 목록을 dictionary로 작성해보고 아메리카노와 바닐라 라떼가 있는지 없는지 확인해보자.

실행결과

```
===== MENU =====  
['아메리카노', '카페라떼', '그린티라떼', '모카라떼']  
메뉴 주문>> 아메리카노  
주문한 아메리카노의 가격은 2000입니다.
```

```
===== MENU =====  
['아메리카노', '카페라떼', '그린티라떼', '모카라떼']  
메뉴 주문>> 바닐라라떼  
주문한 메뉴는 판매하지 않습니다.
```

예제

3. 동물이름을 한글로 입력받아, 해당되는 영어단어로 알려주는 프로그램을 만들어 보자(동물영어사전 프로그램).

실행결과

```
동물이름>> 강아지
강아지는 영어로 dog입니다.
동물이름>> 코끼리
코끼리는 영어로 elephant입니다.
동물이름>> 개
등록되지 않은 언어입니다.
```

실습 예제

1. 주소록에 친구를 등록하고, 친구를 이름으로 검색한 후 연락처를 출력하는 프로그램을 작성하시오. 실행결과를 참고하여 문제를 해결하시오.

1. 딕셔너리 자료 구조를 사용하여 임의로 친구의 이름과 연락처를 저장한다.
2. 딕셔너리에서 전체 친구의 이름만 검색하여 리스트로 변환한 후 출력한다.
3. 찾는 친구의 이름은 반복문을 이용하여 사용자로부터 입력받아 처리한다.(변수명은 적절하게 사용하시오.)
4. 찾는 친구가 있으면 연락처를 출력하고 없으면 에러 메시지를 출력하고 반복문을 벗어난다.

실행결과

```
['홍길동', '김유신', '강감찬', '이순신', '김좌진']  
serach_name : 김유신  
010-2222-1234  
serach_name : 홍길동  
010-1111-1234  
serach_name : 강감찬  
010-3333-1234  
serach_name : 영희  
Not Found  
프로그램 종료
```

실습 예제

2. 현대인들은 축약어를 많이 사용한다. 예를 들어서 "B4(Before)" "TX(Thanks)" "BBL(Be Back Later)" "BCNU(Be Seeing You)" "HAND(Have A Nice Day)"와 같은 축약어들이 있다. 축약어를 풀어서 일반적인 문장으로 변환하는 프로그램을 작성하여 보자.

1. 딕셔너리 자료 구조를 사용하여 축약어를 저장한다.
2. 번역할 문장은 사용자로부터 입력받아 처리한다.(변수명은 적절하게 사용하시오.)
3. 입력받은 문장을 공백 기준으로 잘라서(split함수 이용) 리스트로 만든 후, 축약어가 포함되어 있으면 축약어를 풀어서 변환한다.

실행결과

번역할 문장을 입력하시오: TX MR.Park
Thanks MR.Park

번역할 문장을 입력하시오: TX Stefan BBL
Thanks Stefan Be Back Later

실습 예제

3. 재고품 관리를 하려고 한다. 비품이 떨어지지 않도록 비품을 3개 이상으로 유지한다고 할 때, 각각 사야할 비품과 가격, 총 비용을 계산해 출력하여라.

품명	가격	현재개수
믹스	6000	2
종이컵	15000	3
커피스틱	3000	1

1. 딕셔너리를 사용하여 품목은 key, 가격과 현재개수는 리스트형태의 value로 저장한다.
2. 사야할 비품의 개수에 따른 가격과, 총 비용을 나타내는 변수를 선언하고 초기화한다.
3. 반복문과 조건문을 이용하여 비품의 잔여개수가 3개보다 작을 때 구입해야 할 비품의 품명과 필요 금액을 계산하여 출력한다.
4. 비품의 재고가 3개 이상일때는 continue문을 이용하여 필요금액 계산없이 반복을 수행한다.
5. 재고가 부족한 비품을 구입하는데 필요한 총 비용을 출력한다.

실행결과

믹스 : 1개 부족
믹스를 구입하는 데 필요한 금액은 6000원 입니다.
커피스틱 : 2개 부족
커피스틱을 구입하는 데 필요한 금액은 6000원 입니다.
부족한 비품을 사는데 필요한 총비용 : 12000