

Mid-term Computer Science Project:

영어 단어장 프로그램

Jennifer, Macy, and Aiden

프로젝트 개요

1. 프로젝트 설명
2. 사용한 4가지 *functions*
3. 역할 (*Roles*)
4. 영어 단어장 프로그램 목표
5. 결과 (코드) 발표 및 설명
6. *ChatGPT*의 활용
7. 프로젝트를 통해 배운 점

1. 프로젝트 설명

→ 지금까지 Computer Science 수업들에서 배운 정보들을 토대로 다양한 function들을 활용하여 영어 단어장 프로그램을 만들었습니다. 이 프로그램은 사용자가 직접 자신의 영어 단어 목록과 그 뜻을 관리할 수 있는 간단한 단어장 어플리케이션입니다. 기존에 잘 외워지지 않았던 영어 단어들을 저장하여 나만의 영어 단어 사전을 만들 수 있는 프로그램입니다.

2. 사용한 4가지 functions

- a. 단어 추가
- b. 단어 제거
- c. 단어 랜덤 생성
- d. 단어 전체 출력 (조회)

3. 역할 (Roles)

- 마음: 단어 제거를 할 수 있도록 코드를 작성하기
- 지윤: 단어 추가와 단어 전체출력이 가능하게 만들기
- 민성: 단어 랜덤생성할 때 영어 단어와 한국어 뜻까지 함께 프린트되도록 만들기.

4. 영어 단어장 프로그램 목표

→ dictionary를 사용함으로써 나만의 영어 단어장 프로그램을 만들 수 있다. dictionary 안에 내가 원하는 단어를 추가할 수 있고, 외워진 단어는 다시 삭제할 수 있는 함수를 만들 수 있다. 모든 단어를 한번에 볼 수 있도록 출력하는 함수를 코딩할 수 있다. 마지막으로 랜덤으로 단어를 출력해서 암기를 도와주는 간단한 테스트 기능을 추가할 수 있는 함수 코드를 구현할 수 있다.

또한, 이 프로그램을 처음 만들 때 목표는 사용자가 쉽게 이해하고 사용할 수 있도록 메뉴 구조를 명확하고 간단하게 만드는 것, 그리고 랜덤 학습 기능 function을 추가함으로써 학습의 재미를 더하는 것이었다. 따라서, 결과적으로 기본적인 기능을 충분히 구현하면서 또한 사용자가 쉽게 이해할 수 있는 영어 단어장을 만드는 것이 목표이다.

5. 결과 (코드) 발표 및 설명

-(코드 완성본)

```
import random

# 단어장 덱서너리 초기화
word_dict = {}

# 단어 추가 함수
def add_word():
    word = input("영어 단어: ")
    meaning = input("뜻 (한국어): ")
    word_dict[word] = meaning
    print("단어를 추가하였습니다.")

# 단어 제거 함수
def sub_word():
    word = input("삭제할 단어 : ")
    if word in word_dict:
        del word_dict[word]
        print("단어를 삭제하였습니다.")
    else:
        print(f"{word}은 (는) 목록에 없습니다.\n")

# 단어 랜덤 생성 함수
def random_word():
    if word_dict:
        word, meaning = random.choice(list(word_dict.items()))
        print(f"랜덤 단어: {word}, 뜻: {meaning}")
    else:
        print("단어장이 비어있습니다. 단어를 추가해주세요.")

# 단어 전체 출력 함수
def display_all():
    if word_dict:
        print("=== 전체 단어 ===")
        for word, meaning in word_dict.items():
            print(f"영어 단어: {word}, 한글 뜻: {meaning}")

# 메뉴 인터페이스
def menu():
    while True:
        print("=== 나만의 단어장 프로그램 ===")
        print("1. 단어 추가")
```

```

print("2. 단어 제거")
print("3. 단어 랜덤 생성")
print("4. 전체 단어 목록 보기")
choice = input("원하는 작업을 선택하세요 (1-5): ")

if choice == '1':
    add_word()
elif choice == '2':
    sub_word()
elif choice == '3':
    random_word()
elif choice == '4':
    display_all()
else:
    print("잘못된 입력입니다. 다시 선택하세요.\n")

# 프로그램 실행
menu()

```

6. ChatGPT의 활용

→ 랜덤 기능을 사용할 수 있는 random 모듈의 사용법을 알고자 chat gpt를 활용했습니다. dictionary를 쓸 때 사용할 랜덤생성 기능에 대한 코드를 물어보았으며, chat gpt는 random.choice()를 사용해 랜덤으로 단어와 뜻을 선택하는 것이라고 응답하였습니다. 또한 파이썬의 random 모듈은 기본적으로 리스트나 튜플 같은 데이터에서 요소를 무작위로 선택할 수 있습니다. 하지만 딕셔너리는 키-값 쌍으로 이루어져 있어, 먼저 딕셔너리의 키나 값들을 리스트로 변환한 다음 무작위 선택을 해야 한다고 했습니다. 그 설명을 이용해 코드를 작성하였고 단어를 랜덤으로 출력하는데에 유용하게 쓸 수 있었습니다.

7. 프로젝트를 통해 배운 점

→ random.choice()를 사용해 딕셔너리의 (키, 값) 쌍 중 무작위로 선택하는 방법을 익혔습니다. 이를 통해 데이터(딕셔너리)에서 랜덤 요소를 선택하는 방법에 대한 이해가 깊어졌습니다.

여러 기능(단어 추가, 삭제, 조회 등)을 함수로 구현하여 코드의 재사용성을 높임으로써 함수의 효율성을 깊이 느끼고 함수에 익숙해졌습니다. 또한 사용자와 상호작용하는 메뉴 시스템을 만들어 인터페이스를 설계하는 것에 대해 배웠습니다.