※ 단어장 만들기 프로젝트 *영단어장!*

1. 모델 구축 - 표 table → 6/30 화요일
2. 자료구조 설계
   - 2.1. 자료구조도 작도
   - 2.2. 세모리맵 작도 - 스택 → 7/1 수요일
3. 알고리즘 설계
   - 3.1. 일반 설계
     - 3.1.1. 시스템 다이어그램
   - 3.2. 상세 설계
     - 3.2.1. 모델구축
     - 3.2.2. 자료명세서
     - 3.2.3. 처리과정 → 1/2 목요일
     - 3.2.4. NS chart
     - 3.2.5. 검토-메모리맵 ① 테스트 시나리오·구현 ②

---

4. 디스크 파일 설계
   - 4.1. 시스템 다이어그램 - Load Save 추가
   - 4.2. 상세 설계 ③ 테스트 시나리오·구현 ④

---

5. 사용자 인터페이스 설계
   - 5.1. 메뉴설계
   - 5.2. 화면설계
   - 5.3. 제어구조 설계
   - 5.4. 시스템 다이어그램 ⑤ 구현 ⑥

도메인 용어

갱신 Update
- 삽입 : 기재한다 / 뜻 추가하다.
- 삭제 : 지우다
- 변경 : 고치다

질의 Query
- 검색
- 정렬

단어장 기능
1. 적는다.
2. 추가한다. (뜻을)
3. 봤다.
4. 지우다.

1. 포털구조

번호	한글	번호	영어	영어 예문	한글 예문(뜻)
1	포기하다	1	give up	I gave up smoking last year.	나는 작년에 담배끊는것을 포기했다
		2	quit	She quit her job yesterday.	
		3	abandon	They abandoned the project.	
		4	surrender	The soldiers surrendered to the enemy.	
2	시도하다	1	try	I will try my best.	
		2	attempt	He attempted to solve the problem.	
		3	endeavor	She endeavored to improve her english	
		4	seek	They sought to find a better solution	
		5	strive	We strive to provide excellent service	
3	:		:	:	:

X

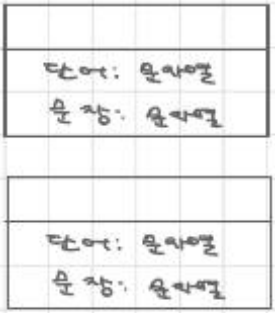
* 도메인 용어

1. 기재하다. Record → 새 단어 입력하기
2. 추가하다. Add → 영어단어 추가하기
3. 찾 다. Find → 단어찾기 → 영어→한글 / 한글→영어
4. 고치다 Correct → 단어 수정하기
5. 지우다. Erase → 단어 지우기 (삭제)
6. 정리하다. Arrange → 단어 정렬하기

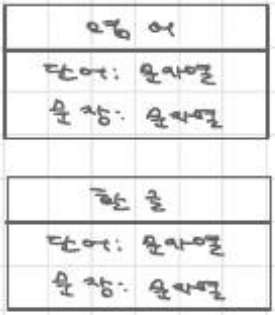
2. 자료구조 설계

2.1. 자료구조도 작도.

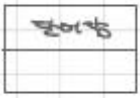
2.1.1. 필드 찾기



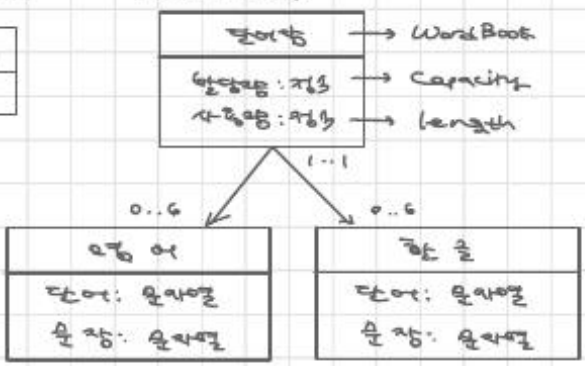
2.1.2. 레코드 찾기



2.1.3. 파일 찾기



2.1.4. 관계 찾기



1. 모델구축

	word	meaning		synonyms	examples
번호	단어	뜻	번호	동의어	예문
1	give up	포기하다	0	give up	I gave up smoking last year.
			1	quit	she quit her job yesterday.
			2	renounce	They renounced a priviledge.
			3	surrender	The soldiers surrendered to the enemy
2	try	시도하다	0	try	I will try my best.
			1	attempt	He attempted to solve the problem.
			2	endeavor	she endeavored to improve her english
			3	seek	They sought to find a better solution
			4	strive	We strive to provide excellent service
3	⋮	⋮			⋮ ⋮



X

* 도메인 용어

1. 기재하다. Record → 새 단어 입력받기
2. 추가하다. Add → 영어단어 추가하기
3. 찾 다. Find → 단어찾기 → 영어 → 반출
   반출 → 영어
4. 고치다 Correct → 단어 수정하기
5. 지우다. Erase → 단어 지우기 (삭제)
6. 정리하다. Arrange → 단어 정렬하기

1. 모델구축

번호	meaning 뜻	번호	word 단어	word class part of speech 품사	gone- 활용형 ~~grammar (pattern)~~ ~~문법 (패턴)~~	examples 예문
1	포기하다	1	give up	phr.v.	give up+N / +V-ing	I gave up smoking last year.
		2	quit	v.	quit+N / quit V-ing	she quit her job yesterday.
		3	renounce	v.	abandon+N	They renounced a priviledge.
		4	surrender	v.	surrender to +V	The soldiers surrendered to the enemy
2	시도하다	1	try	v.	try to V / try V-ing	I will try my best.
		2	attempt	v./n.	attempt to V	He attempted to solve the problem.
		3	endeavor	v./n.	endeavor to V	she endeavored to improve her english
		4	seek	v.	seek to V / seek+N	They sought to find a better solution
		5	strive	v.	strive to N.	We strive to provide excellent service
3	⋮		⋮		⋮	⋮

* 도메인 용어

1. 기재하다. Record → 새 단어 입력받기
2. 찾 다. Find → 단어찾기 → 영어 → 반글
   반글 → 영어
3. 추가하다. Add → 영어단어 추가하기
4. 고치다 Correct → 단어 수정하기
5. 지우다. Erase → 단어 지우기 (삭제)
6. 정리하다. Arrange → 단어 정렬하기

n. 명사 prep. 전치사
v. 동사 conj. 접속사
adj. 형용사 pron. 대명사
adv. 부사 phr. 구
* phr.v. 구동사

X

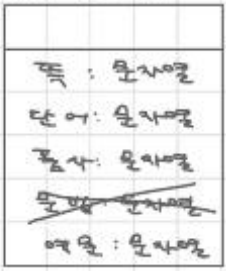
★ 프로젝트 시작하기 전,
하고자 하는 것이 무엇인지 파악하고, 빠르게 기초지식 습득하기 ★

2. 자료구조 설계

2.1. 자료구조도 작도

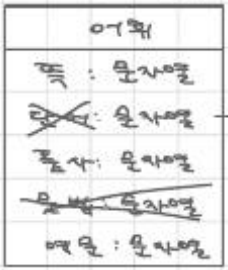
후보
1. 영어 X → S가 만들음
2. 어휘 → 어휘=단어..아닌가
3. 사전 X → =WordBook 느낌.
4. 정보 △
넘 포괄적인가

2.1.1. 필드 찾기

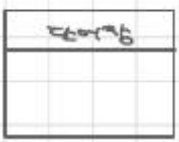


뜻 : 문자열
단어 : 문자열
품사 : 문자열
~~문법 : 문자열~~
예문 : 문자열

2.1.2. 레코드 찾기

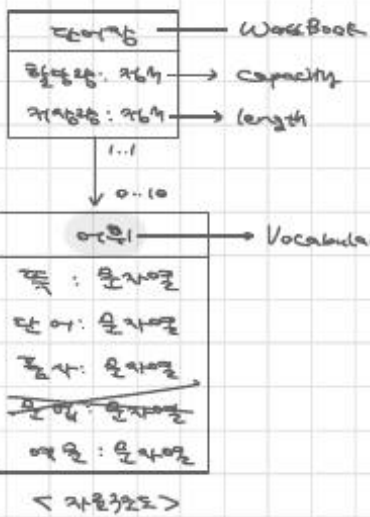


어휘
뜻 : 문자열
~~단어~~ : 문자열 → 철자 ..
품사 : 문자열
~~문법 : 문자열~~
예문 : 문자열

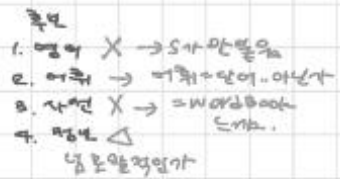


Letter / Spelling

뜻을 가진 가장 작은 단위 : 형태소







Morpheme

유형

2.1.3. 파일찾기

단어장

2.1.4. 관계 맺기

단어장
할당량 : 정수 → capacity
저장량 : 정수 → length

WordBook

1..1

0..10

어휘
뜻 : 문자열
단어 : 문자열
품사 : 문자열
~~문법 : 문자열~~
예문 : 문자열

Vocabulary

X

< 자료구조도 >

1. 모델구축

번호	letter 철자	meaning 뜻	품사	examples 예문
1	attempt	시도하다	동사(v.)	I will attempt to solve the problem.
2	abandon	포기하다, 버리다	동사(v.)	They abandoned the project because of the cost.
3	achieve	이루다, 성취하다	동사(v.)	He achieved his goal through hard work.
4	improve	향상시키다, 개선하다	동사(v.)	Regular practice improves your English.
5	opportunity	기회	명사(n.)	This is a great opportunity to learn something new.
6	environment	환경	명사(n.)	We must protect the environment.
7	reliable	믿을 수 있는	형용사(adj.)	She is a reliable coworker.
8	curious	호기심의 많은	형용사(adj.)	The child was curious about everything.
9	carefully	주의깊게, 조심스럽게	부사(adv.)	Please read the instructions carefully.
10	quickly	빠르게	부사(adv.)	He quickly finished his homework.

* 도메인 용어

1. 기재하다. Record → 새 단어 입력받기
2. 찾 다. Find → 단어찾기 → 영어 → 반글
   반글 → 영어
3. 추가하다. Add → 영어단어 추가하기
4. 고치다 Correct → 단어 수정하기
5. 지우다. Erase → 단어 지우기 (삭제)
6. 정리하다. Arrange → 단어 정렬하기

n. 명사 prep. 전치사
v. 동사 conj. 접속사
adj. 형용사 pron. 대명사
adv. 부사 phr. 구
* phr.v. 구동사

1. 모델구축

뜻·예문 → 고치기
입력

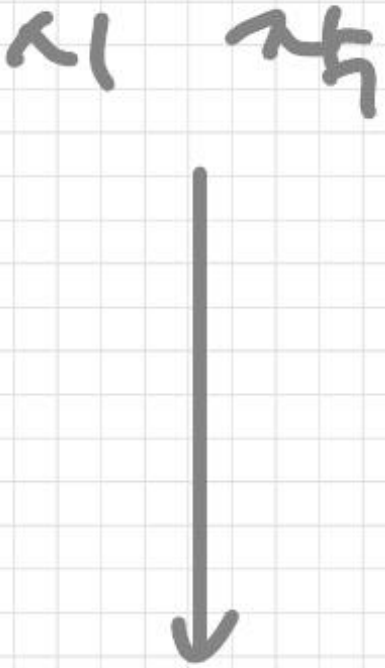
	letter	meaning	partOfSpeech	exampleSentence
번호	철자	뜻	품사	예문
1	attempt	시도하다 ; 시도	v. ; n.	I will attempt to solve the problem.
2	change	바꾸다 ; 변화	v. ; n.	People can change their habits
3	study	공부하다 ; 연구	v. ; n.	I study English every day.
4	record	기록하다 ; 기록	v. ; n.	Please record the meeting.
5	answer	대답하다	v.	She answered my question.
6	improve	향상시키다	v.	Regular practice improves your English.
7	opportunity	기회	n.	This is a great opportunity to learn something new.
8	reliable	믿을 수 있는	adj.	She is a reliable coworker.
9	carefully	주의 깊게	adv.	Please read the instructions carefully.
10	quickly	빠르게	adv.	He quickly finished his homework.

* 도메인 용어

1. 기재하다. Record → 새 단어 입력받기
2. 찾 다. Find → 단어 찾기
3. 고치다 Correct → 단어 수정하기
4. 지우다. Erase → 단어 지우기 (삭제)
5. 정리하다. Arrange → 단어 정렬하기

n. 명사 prep. 전치사
v. 동사 conj. 접속사
adj. 형용사 pron. 대명사
adv. 부사 phr. 구
* phr.v. 구동사

X..


시 작

1. 모델구축

	spelling	partOfSpeech	meaning	exampleSentence
번호	철자	품사	뜻	예문
1	attempt	v.	시도하다	I will attempt to solve the problem.
2	change	v.	바꾸다	People can change their habits
3	study	v.	공부하다	I study English every day.
4	study	n.	연구	Studies found that people are selfish.
5	attempt	n.	시도	Making an attempt is a good thing.
6	improve	v.	향상시키다	Regular practice improves your English.
7	change	n.	변화	An attempt brings change to your life.
8	reliable	adj.	믿을 수 있는	She is a reliable coworker.
9	credible	adj.	믿을 수 있는	The witness gave a credible explanation.
10	quickly	adv.	빠르게	He quickly finished his homework.

* 도메인 용어

1. 기재하다. Record → 단어 입력받기
2. 찾 다. Find → 철자/뜻 찾기
3. 고치다 Correct → 뜻/예문 수정하기
4. 지우다. Erase → 단어 지우기 (삭제)
5. 정리하다. Arrange → 단어 정렬하기

n. 명사 prep. 전치사
v. 동사 conj. 접속사
adj. 형용사 pron. 대명사
adv. 부사 phr. 구
* phr.v. 구동사

2. 자료구조 설계

2.1. 자료구조도 작도

후보
1. 영어 X → S가 만들음
2. 어휘 → 어휘=단어..아닌가
3. 사전 X → =WordBook 느낌.
4. 정보 △
넘 포괄적인가

2.1.1. 필드 찾기

철자 : 문자열
품사 : 문자열
뜻 : 문자열
예문 : 문자열

2.1.2. 레코드 찾기

단어
철자 : 문자열
품사 : 문자열
뜻 : 문자열
예문 : 문자열

→ 철자 Letter / Spelling

2.1.3. 파일찾기

단어장

2.1.4. 관계 맺기

단어장
할당량 : 정수
저장량 : 정수

→ WordBook
→ capacity
→ length

1..1
↓
0..10

단어
철자 : 문자열
품사 : 문자열
뜻 : 문자열
예문 : 문자열

→ Word

< 자료구조도 >

2.2. 데코앱맵 작도

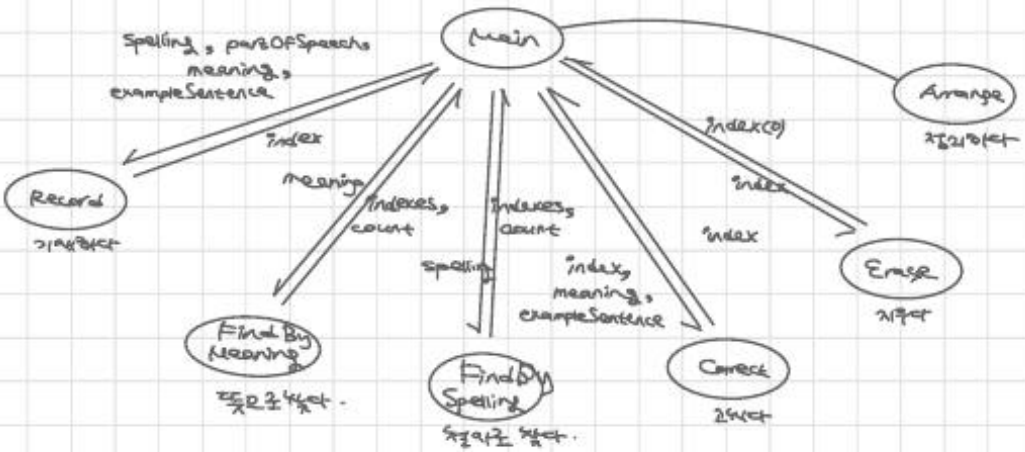
WordBook

word				capacity	length
Spelling	partOf Speech	meaning	exampleSentence	10	0
''	''	''	''		
Spelling	partOf Speech	meaning	exampleSentence		
''	''	''	''		
Spelling	partOf Speech	meaning	exampleSentence		
''	''	''	''		
Spelling	partOf Speech	meaning	exampleSentence		
''	''	''	''		
Spelling	partOf Speech	meaning	exampleSentence		
''	''	''	''		
Spelling	partOf Speech	meaning	exampleSentence		
''	''	''	''		
Spelling	partOf Speech	meaning	exampleSentence		
''	''	''	''		
Spelling	partOf Speech	meaning	exampleSentence		
''	''	''	''		
Spelling	partOf Speech	meaning	exampleSentence		
''	''	''	''		
Spelling	partOf Speech	meaning	exampleSentence		
''	''	''	''		

3. 알고리즘 설계

3.1. 일반설계

3.1.1. 시스템 다이어그램 작도



3.2. 상세설계

3.2.1. 모델구축

3.2.2. 자료명세서 3분

1. 단어장 WordBook WordBook 준비

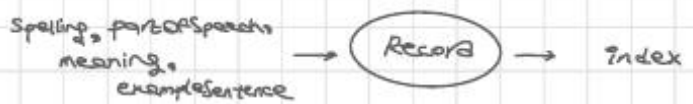
3.2.3. 처리과정

3.2.4. 메모리맵 작도

1. Record 상세설계

1.1. 모델구축

	spelling	partOfSpeech	meaning	exampleSentence
번호	철자	품사	뜻	예문
1	attempt	v.	시도하다	I will attempt to solve the problem.
2	change	v.	바꾸다	People can change their habits
3	study	v.	공부하다	I study English every day.
4	study	n.	연구	Studies found that people are selfish.
5	attempt	n.	시도	Making an attempt is a good thing.
6	improve	v.	향상시키다	Regular practice improves your English.
7	change	n.	변화	An attempt brings change to your life.
8	reliable	adj.	믿을 수 있는	She is a reliable coworker.
9	credible	adj.	믿을 수 있는	The witness gave a credible explanation.
10	quickly	adv.	빠르게	He quickly finished his homework.



spelling, partOfSpeech, meaning, exampleSentence → Record → index

1.2. 자료명세서

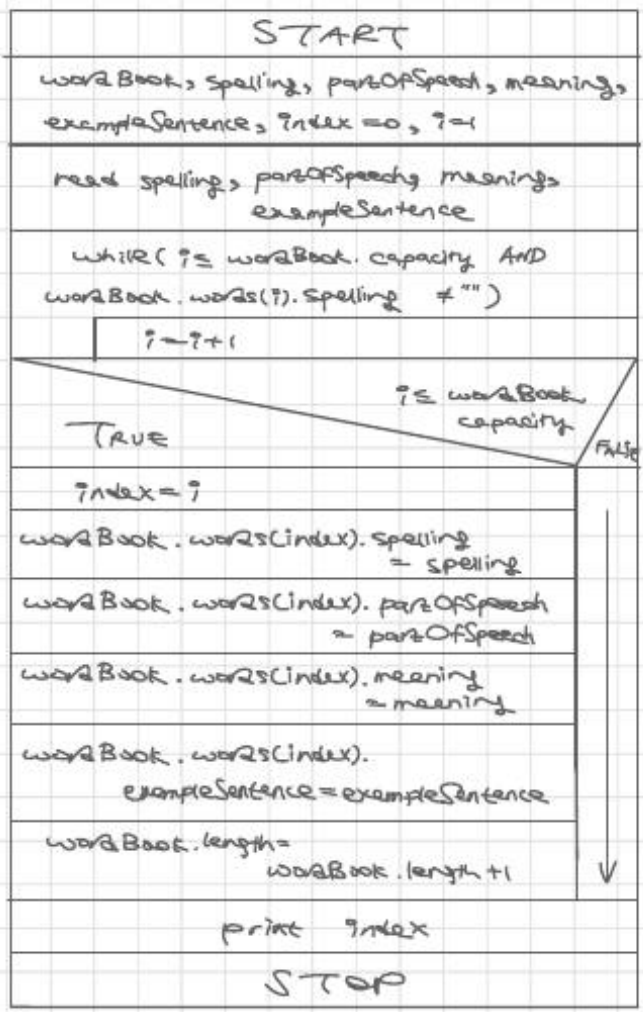
번호	한글	영문	자료유형	3분	비고
	단어장	WordBook	단어장	준비	파일
	단어	Word	단어		레코드
	철자	spelling	문자열		필드
	품사	partOfSpeech	문자열		필드
	뜻	meaning	문자열		필드
	예문	exampleSentence	문자열		필드
	할당량	capacity	정수		
	저장량	length	정수		
1	단어장	WordBook	단어장	준비	파일
2	철자	Spelling	문자열	입력	
3	품사	partOfSpeech	문자열	입력	
4	뜻	meaning	문자열	입력	
5	예문	exampleSentence	문자열	입력	
6	위치	index	정수배열	출력	저장된 위치
7	할당량	capacity	정수	처리	최대 저장개수

g	저장량	length	정수	처리	현재 저장개수
g	반복제어	i	정수	추가	

1.3. 처리과정

1. 철자, 품사, 뜻, 예문을 입력받는다.
2. 빈 줄을 찾는다.
3. 만약 빈 줄이 있으면
   3.1. 각 칸에 적는다.
4. 적은 줄의 위치를 출력한다.
5. 끝내다.

1.4. N-S chart



words				spacing	length
spelling	partOfSpeech	meaning	exampleSentence		
attempt	v.	시도하다	I will attempt to solve the problem.	10	10
spelling	partOfSpeech	meaning	exampleSentence		
change	v.	바꾸다	People can change their habits.		
spelling	partOfSpeech	meaning	exampleSentence		
study	v.	공부하다	I study English everyday.		
spelling	partOfSpeech	meaning	exampleSentence		
study	n.	연구	Studies found that people are selfish.		
spelling	partOfSpeech	meaning	exampleSentence		
attempt	n.	시도	Having an attempt is a good thing.		
spelling	partOfSpeech	meaning	exampleSentence		
improve	v.	향상시키다	Regular practice improve your English.		
spelling	partOfSpeech	meaning	exampleSentence		
change	n.	변화	An attempt brings change to your life.		
spelling	partOfSpeech	meaning	exampleSentence		
reliable	adj.	신뢰할 수 있는	She is a reliable coworker.		
spelling	partOfSpeech	meaning	exampleSentence		
Credible	adj.	신뢰할 수 있는	The witness gave a credible explanation.		

Spelling	partOfSpeech	meaning	exampleSentence
quickly	adv.	快速地	He quickly finished his homework.

Spelling	partOfSpeech	meaning	exampleSentence

index

i

10

4 5 6 7 8 9 10

2. FindMeaning 상세 설계

단어를 찾다.
: 뜻 입력 → 찾기

2.1. 모델구축 : 찾다.

↳ 파일 { 정상 / 질의 < 검색 / 정렬

	spelling	partOfSpeech	meaning	exampleSentence
번호	철자	품사	뜻	예문
1	attempt	v.	시도하다	I will attempt to solve the problem.
2	change	v.	바꾸다	People can change their habits
3	study	v.	공부하다	I study English every day.
4	study	n.	연구	Studies found that people are selfish.
5	attempt	n.	시도	Making an attempt is a good thing.
6	improve	v.	향상시키다	Regular practice improves your English.
7	change	n.	변화	An attempt brings change to your life.
8	reliable	adj.	믿을 수 있는	She is a reliable coworker.
9	credible	adj.	믿을 수 있는	The witness gave a credible explanation.
10	quickly	adv.	빠르게	He quickly finished his homework.



meaning → FindBy Meaning → indexes, count

뜻으로 찾다.

2.2. 자료명세서

번호	한글	영문	자료유형	3분	비고
	단어장	WordBook	단어장	준비	파일
	단어	Word	단어		레코드
	철자	Spelling	문자열		필드
	품사	partOfSpeech	문자열		필드
	뜻	meaning	문자열		필드
	예문	exampleSentence	문자열		필드
	할당량	capacity	정수		
	저장량	length	정수		
1	단어장	WordBook	단어장	준비	파일
2	최대값	MAX	정수	상수	10

3	인덱스(배열)	indexes	정수·배열	입출력
4	개수	count	정수	입출력
5	뜻	meaning	문자열	입출력
6	최댓값	capacity	정수	처리
7	저장값	length	정수	처리
8	반복제어	i	정수	처리
9	반복제어	j	정수	처리

2.3. 처리과정

1. 뜻을 입력받는다.
2. 저장값만큼 반복한다.
   2.1. 뜻이 같으면
      2.1.1. 개수를 센다.
      2.1.2. 인덱스를 기억한다.
3. 개수, 인덱스들을 출력한다.
4. 종료

2.4. N-S chart



words				capacity	length
spelling	partOfSpeech	meaning	exampleSentence		
attempt	v.	시도하다	I will attempt to solve the problem.	10	10
spelling	partOfSpeech	meaning	exampleSentence		
change	v.	바꾸다	People can change their habits.		
spelling	partOfSpeech	meaning	exampleSentence		
study	v.	공부하다	I study English everyday.		
spelling	partOfSpeech	meaning	exampleSentence		
study	n.	연구	Studies found that people are selfish.		
spelling	partOfSpeech	meaning	exampleSentence		
attempt	n.	시도	Making an attempt is a good thing.		
spelling	partOfSpeech	meaning	exampleSentence		
improve	v.	향상시키다	Regular practice improve your English.		
spelling	partOfSpeech	meaning	exampleSentence		
change	n.	변화	An attempt brings change to your life.		
spelling	partOfSpeech	meaning	exampleSentence		
reliable	adj.	신뢰할 수 있는	She is a reliable coworker.		
spelling	partOfSpeech	meaning	exampleSentence		
credible	adj	신뢰할 수 있는	Please read the instructions carefully.		

indexes
1
3

spelling	partOfSpeech	meaning	exampleSentence
quickly	adv.	快速	He quickly finished his homework.

meaning
attempt

count

	2
i	11
j	3

MAX=10

counter2

indexes 1,5

3. FindSpelling 상세 설계

단어를 찾다.
: 뜻 입력 → 찾기

3.1. 오델구동 : 찾다.
↳ 파일 { 정상 / 질의 < 검색 / 정렬

번호	spelling 철자	partOfSpeech 품사	meaning 뜻	exampleSentence 예문
1	attempt	v.	시도하다	I will attempt to solve the problem.
2	change	v.	바꾸다	People can change their habits
3	study	v.	공부하다	I study English every day.
4	study	n.	연구	Studies found that people are selfish.
5	attempt	n.	시도	Making an attempt is a good thing.
6	improve	v.	향상시키다	Regular practice improves your English.
7	change	n.	변화	An attempt brings change to your life.
8	reliable	adj.	믿을 수 있는	She is a reliable coworker.
9	credible	adj.	믿을 수 있는	The witness gave a credible explanation.
10	quickly	adv.	빠르게	He quickly finished his homework.

Spelling → FindBy Spelling → indexes, count

철자로 찾다.

3.2. 자료명세서

번호	한글	영문	자료유형	3분	비고
	단어장	WordBook	단어장	준비	파일
	단어	Word	단어		레코드
	철자	spelling	문자열		필드
	품사	partOfSpeech	문자열		필드
	뜻	meaning	문자열		필드
	예문	exampleSentence	문자열		필드
	할당량	capacity	정수		
	저장량	length	정수		
1	단어장	WordBook	단어장	준비	파일
2	최대값	MAX	정수	상수	10

3	인덱스(배열)	indexes	정수·배열	입출력
4	개수	count	정수	입출력
5	철자	spelling	문자열	입출력
6	최댓값	capacity	정수	처리
7	저장값	length	정수	처리
8	반복제어	i	정수	처리
9	반복제어	j	정수	처리

3.3. 처리과정

1. 철자를 입력받는다.
2. 저장값만큼 반복한다.
   2.1. 철자가 같은지
      2.1.1. 위치를 저장한다.
      2.1.2. 개수를 센다
3. 개수, 위치들을 출력한다.
4. 끝내다.

3.4. N-S chart



words				capacity	length
speling	partOfSpeech	meaning	exampleSentence	10	10
attempt	v.	시도하다	I will attempt to solve the problem.		
speling	partOfSpeech	meaning	exampleSentence		
change	v.	바꾸다	People can change their habits.		
speling	partOfSpeech	meaning	exampleSentence		
study	v.	공부하다	I study English everyday.		
speling	partOfSpeech	meaning	exampleSentence		
study	n.	연구	Studies found that people are selfish.		
speling	partOfSpeech	meaning	exampleSentence		
attempt	n.	시도	Making an attempt is a good thing.		
speling	partOfSpeech	meaning	exampleSentence		
improve	v.	향상시키다	Regular practice improve your English.		
speling	partOfSpeech	meaning	exampleSentence		
change	n.	변화	An attempt brings change to your life.		
speling	partOfSpeech	meaning	exampleSentence		
reliable	adj.	신뢰할 수 있는	She is a reliable coworker.		
speling	partOfSpeech	meaning	exampleSentence		
credible	adj	신뢰할 수 있는	Please read the instructions carefully.		

indexes

8

9

spelling	partOfSpeech	meaning	exampleSentence
quickly	adv.	빠르게	He quickly finished his homework

spelling
멀을 수 있는

count

	2
i	11
j	3

MAX=10

8 | 9

2

3. Correct 상세설계

3.1. 모델구축 : 고치다.

↳ 파일 { 개인 / 질의 < 삽입 / 삭제 / 변경 / 검색 / 정렬

	spelling	partOfSpeech	meaning	exampleSentence
번호	철자	품사	뜻	예문
1	attempt	v.	시도하다	I will attempt to solve the problem.
2	change	v.	바꾸다	People can change their habits
3	study	v.	공부하다	I study English every day.
4	study	n.	연구	Studies found that people are selfish.
5	attempt	n.	시도	Making an attempt is a good thing.
6	improve	v.	더 좋아지다.	The weather is improving
7	change	n.	변화	An attempt brings change to your life.
8	reliable	adj.	믿을 수 있는	She is a reliable coworker.
9	credible	adj.	믿을 수 있는	The witness gave a credible explanation.
10	quickly	adv.	빠르게	He quickly finished his homework.

index, meaning, exampleSentence → Correct → index

3.2. 자료명제서

번호	한글	영문	자료유형	3분	비고
	단어장	WordBook	단어장	준비	파일
	단어	Word	단어		레코드
	철자	Spelling	문자열		필드
	품사	partOfSpeech	문자열		필드
	뜻	meaning	문자열		필드
	예문	exampleSentence	문자열		필드
	할당량	capacity	정수		
	저장량	length	정수		
1	단어장	WordBook	단어장	준비	파일
2	단어	words	단어	처리	
3	위치	index	정수	입·출력	저장 위치

4	뜻	meaning	순서열	입력
5	예문	exampleSentence	순서열	입력

3.5. 처리과정.

1. 수정할 위치, 뜻, 예문을 입력받는다
2. 입력받은 뜻으로 저장한다
3. 입력받은 예문으로 저장한다
4. 수정된 내용을 출력한다.
5. 끝낸다.

3.6. N-S chart

START
WordBook, index, meaning, exampleSentence
read index, meaning, exampleSentence
WordBook.words(index).meaning = meaning
WordBook.words(index).exampleSentence = exampleSentence
print index
STOP

words				spacing	length
Spelling	partOfSpeech	meaning	exampleSentence		
attempt	v.	시도하다	I will attempt to solve the problem.	10	10
Spelling	partOfSpeech	meaning	exampleSentence		
change	v.	바꾸다	People can change their habits.		
Spelling	partOfSpeech	meaning	exampleSentence		
study	v.	공부하다	I study English everyday.		
Spelling	partOfSpeech	meaning	exampleSentence		
study	n.	연구	Studies found that people are selfish.		
Spelling	partOfSpeech	meaning	exampleSentence		
attempt	n.	시도	Having an attempt is a good thing.		
Spelling	partOfSpeech	meaning	exampleSentence		
improve	v.	더 좋아지다	The weather is improving.		
Spelling	partOfSpeech	meaning	exampleSentence		
change	n.	변화	An attempt brings change to your life.		
Spelling	partOfSpeech	meaning	exampleSentence		
reliable	adj.	신뢰할 수 있는	She is a reliable coworker.		
Spelling	partOfSpeech	meaning	exampleSentence		
credible	adj.	신뢰할 수 있는	The witness save a credible		

explanation.

spelling	partOfSpeech	meaning	exampleSentence
quickly	adv.	快速	He quickly finished his homework.

Index

6

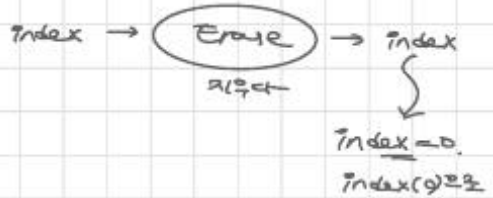
meaning	exampleSentence
天气好转	The weather is improving.

4. Erase 상세설계

4.1. 모델구축 : 지우다

↳ 파일 { 갱신 < 생입, 상세, 변경 / 실의 < 검색, 저장

	spelling	partOfSpeech	meaning	exampleSentence
번호	철자	품사	뜻	예문
1	attempt	v.	시도하다	I will attempt to solve the problem.
2	change	v.	바꾸다	People can change their habits
3	study	v.	공부하다	I study English every day.
4	study	n.	연구	Studies found that people are selfish.
5	attempt	n.	시도	Making an attempt is a good thing.
6	—	v.	더 좋아지다.	The weather is improving
7	change	n.	변화	An attempt brings change to your life.
8	reliable	adj.	믿을 수 있는	She is a reliable coworker.
9	credible	adj.	믿을 수 있는	The witness gave a credible explanation.
10	quickly	adv.	빠르게	He quickly finished his homework.



index → Erase → index

지우다

index = 0.

index(0)으로

4.2. 자료명세서

번호	한글	영문	자료유형	3분	비고
	단어장	WordBook	단어장	준비	파일
	단어	Word	단어		레코드
	철자	spelling	문자열		필드
	품사	partOfSpeech	문자열		필드
	뜻	meaning	문자열		필드
	예문	exampleSentence	문자열		필드
	할당량	capacity	정수		
	저장량	length	정수		
1	단어장	WordBook	단어장	준비	파일
2	단어	words	단어	처리	
3	위치	index	정수	입·출력	저장 위치

4	철자	Spelling	문자열	처리
5	뜻	meaning	문자열	처리
6	품사	partOfSpeech	문자열	처리
7	예문	exampleSentence	문자열	처리

4.3. 처리과정

1. 지울 위치를 입력받는다.
2. 위치의 철자에 하이폴우 적는다
3. 위치를 옮긴다.
4. 위치를 출력한다
5. 끝내다.

지움 표시를 한다

4.4. N-S chart

START
wordBook.index
read index
wordBook.words(index).spelling = "—"
index = 0
print index
STOP

words				spacing	length
spelling	partOfSpeech	meaning	exampleSentence		
attempt	v.	시도하다	I will attempt to solve the problem.	10	10
spelling	partOfSpeech	meaning	exampleSentence		
change	v.	바꾸다	People can change their habits.		
spelling	partOfSpeech	meaning	exampleSentence		
study	v.	공부하다	I study English everyday.		
spelling	partOfSpeech	meaning	exampleSentence		
study	n.	연구	Studies found that people are selfish.		
spelling	partOfSpeech	meaning	exampleSentence		
attempt	n.	시도	Making an attempt is a good thing.		
spelling	partOfSpeech	meaning	exampleSentence		
—	v.	향상시키다	The weather is improving.		
spelling	partOfSpeech	meaning	exampleSentence		
change	n.	변화	An attempt brings change to your life.		
spelling	partOfSpeech	meaning	exampleSentence		
reliable	adj.	신뢰할 수 있는	She is a reliable coworker.		
spelling	partOfSpeech	meaning	exampleSentence		
carefully	adv.	조심스럽게	Please read the instructions carefully.		

spelling	partOfSpeech	meaning	exampleSentence
quickly	adv.	快速地	He quickly finished his homework.

Index
0

5. Arrange 상세설계

5.1. 모델구축: 정리하다

→ 파일 { 개선 ← 생활, 쓰레, 연결 / 질의 ← 검색, 정렬

Arrange

정리하다.

letter 기준 정렬.

ASCII 코드

작은값 → 큰값

모름차순

	spelling	partOfSpeech	meaning	exampleSentence
번호	철자	품사	뜻	예문
1	attempt	v.	시도하다	I will attempt to solve the problem.
2	change	v.	바꾸다	People can change their habits
3	study	v.	공부하다	I study English every day.
4	study	n.	연구	Studies found that people are selfish.
5	attempt	n.	시도	Making an attempt is a good thing.
6	—	v.	더 좋아지다.	The weather is improving
7	change	n.	변화	An attempt brings change to your life.
8	reliable	adj.	믿을 수 있는	She is a reliable coworker.
9	credible	adj.	믿을 수 있는	The witness gave a credible explanation.
10	quickly	adv.	빠르게	He quickly finished his homework.



	spelling	partOfSpeech	meaning	exampleSentence
번호	철자	품사	뜻	예문
1	attempt	v.	시도하다	I will attempt to solve the problem.
2	attempt	n.	시도	Making an attempt is a good thing.
3	change	v.	바꾸다	People can change their habits
4	change	n.	변화	An attempt brings change to your life.
5	credible	adj.	믿을 수 있는	The witness gave a credible explanation.
6	reliable	adj.	믿을 수 있는	She is a reliable coworker.
7	study	v.	공부하다	I study English every day.
8	study	n.	연구	Studies found that people are selfish.
9	quickly	adv.	빠르게	He quickly finished his homework.
10	—	v.	더 좋아지다.	The weather is improving

1. 빈칸 맨 아래도 = 내림차순
2. 저장량 · count
3. 알파벳순 모름차순

5.2. 자료명세서

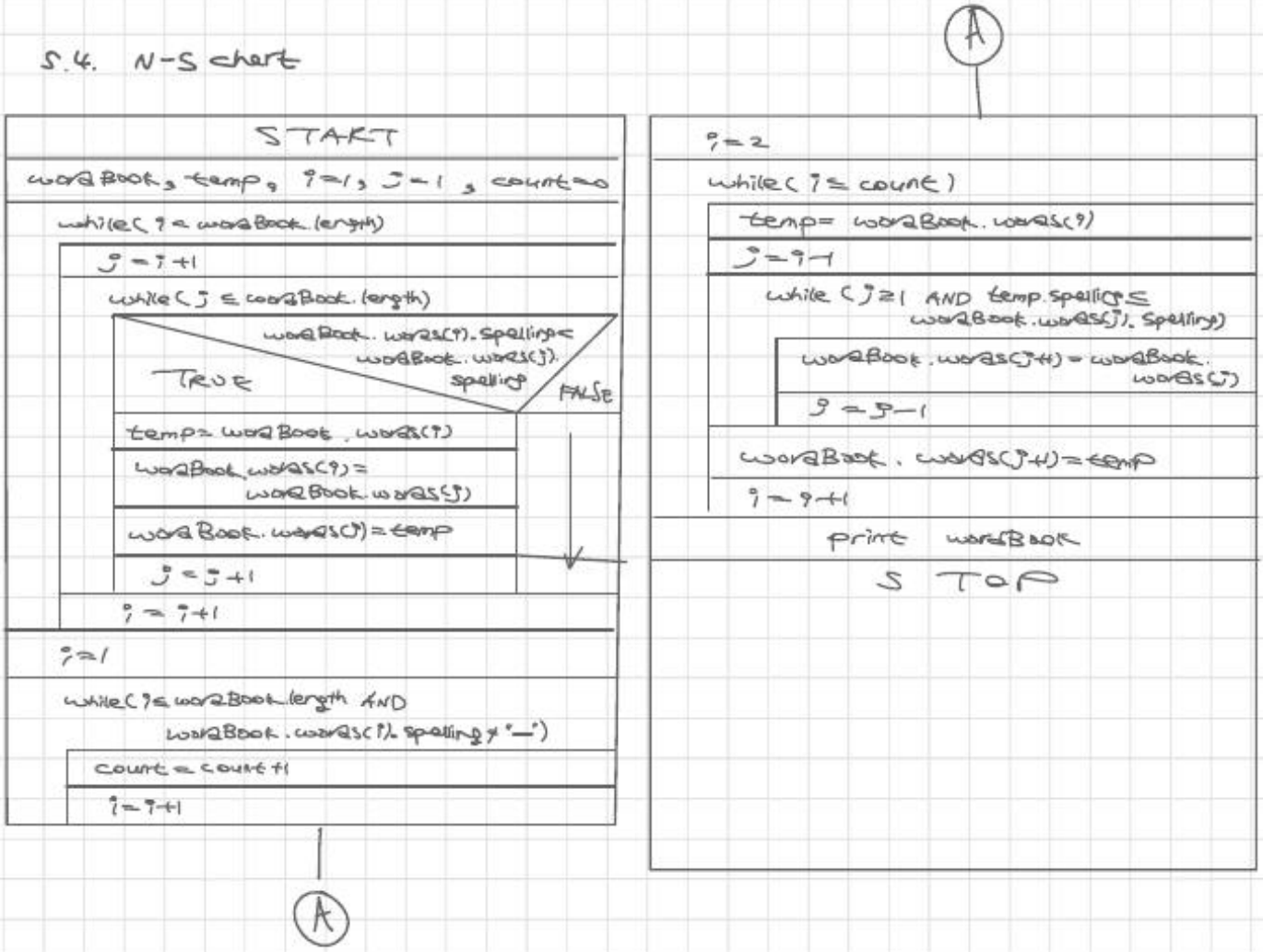
번호	한글	영문	자료유형	3분	비고
	단어장	WordBook	단어장	준비	파일
	단어	Word	단어		레코드
	철자	spelling	문자열		필드
	품사	partOfSpeech	문자열		필드
	뜻	meaning	문자열		필드
	예문	exampleSentence	문자열		필드
	할당량	capacity	정수		
	저장량	length	정수		
1	단어배열	words	단어배열	처리	
2	철자	spelling	문자열		
3	품사	partOfSpeech	문자열		
4	뜻	meaning	문자열		
5	예문	exampleSentence	문자열		
6	할당량	capacity	정수		
7	저장량	length	정수		
8	반복제어	;			
9	반복제어	;			

5.3. 처리과정.

1. 저장량만큼 반복한다.
   1.1. 만약 이름이 더 크면
      1.1.1. 위로 올려 정리한다. → 내림차순 정렬
2. 저장량만큼 반복한다.
   2.1. 빈칸이 아니면
      2.1.1. 개수를 센다.
3. 저장량만큼 반복한다. → 빈칸 제외하여 센 count, wordBook.length = count
   3.1. 만약 이름이 작으면
      3.1.1. 위로 올려 정리한다. → 오름차순 정렬
   3.2. 이름이 같으면
      3.2.1. 품사 기준 오름차순으로 정리한다.
4. wordBook을 출력한다.
5. 끝내다.

1. 철자로 내림차순 한다.
2. 지운 줄을 뺀 적인 줄을 센다.
3. 철자로 오름차순 한다.

5.4. N-S chart


START
wordBook, temp, i=1, j=1, count=0
while(i < wordBook.length)
j = i+1
while(j ≤ wordBook.length)
wordBook.words(i).spelling < wordBook.words(j).spelling
TRUE
FALSE
temp = wordBook.words(i)
wordBook.words(i) = wordBook.words(j)
wordBook.words(j) = temp
j = j+1
i = i+1
i=1
while(i ≤ wordBook.length AND wordBook.words(i).spelling ≠ '—')
count = count+1
i = i+1
A
A
i = 2
while(i ≤ count)
temp = wordBook.words(i)
j = i-1
while(j ≥ 1 AND temp.spelling ≤ wordBook.words(j).spelling)
wordBook.words(j+1) = wordBook.words(j)
j = j-1
wordBook.words(j+1) = temp
i = i+1
print wordBook
STOP

2) 버블 정렬 Bubble Sort.

→ 앞에 있는 애들이랑 비교하여 큰 애를 뒤로 밀어내기

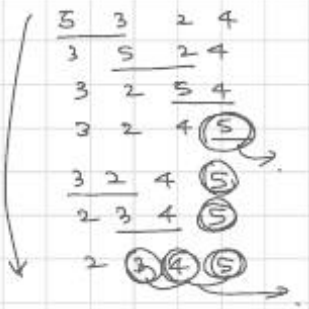
* 거품이 뒤로 올라가는 것처럼
큰 값이 계속 뒤로 밀려남.

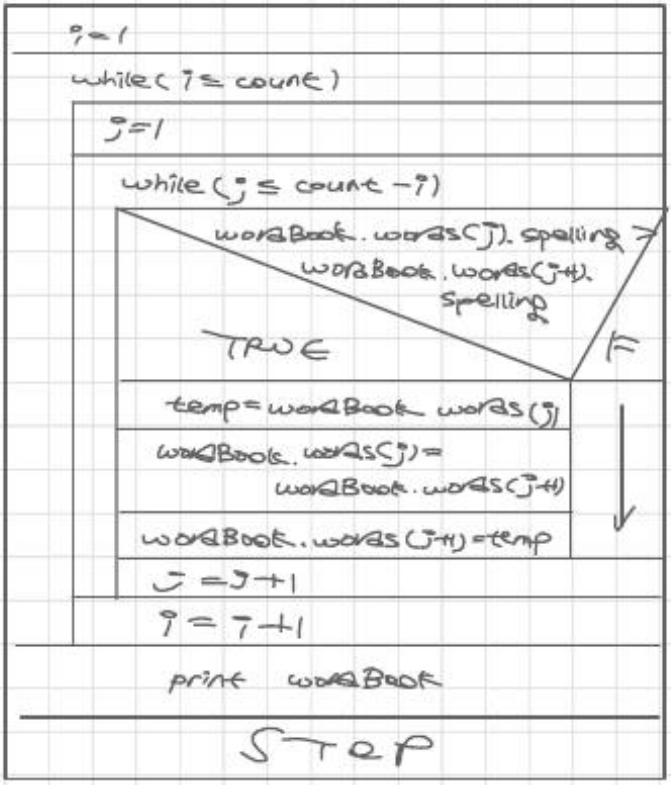
5 3 2 4
3 5 2 4
3 2 5 4
3 2 4 5

3 2 4 5
2 3 4 5
2 3 4 5

등등 .. 이러면
— 이 올라감

```
i=1
while( i ≤ count )
    j=1
    while( j ≤ count - i )
        woraBook.woras(j).spelling >
            woraBook.woras(j+1).
                spelling
        TRUE                          F
        temp = woraBook.woras(j)
        woraBook.woras(j) =
            woraBook.woras(j+1)
        woraBook.woras(j+1) = temp
        j = j+1
    i = i+1
print woraBook
STOP
```





words				spacing	length
spelling	partOfSpeech	meaning	exampleSentence		
attempt	n.	시도	Making an attempt is a good thing	10	10
spelling	partOfSpeech	meaning	exampleSentence		
attempt	v.	시도하다	I will attempt to solve the problem		
spelling	partOfSpeech	meaning	exampleSentence		
Carefully	adv.	조심스럽게	please read the instructions carefully		
spelling	partOfSpeech	meaning	exampleSentence		
change	v.	바꾸다	People can change their habits		
spelling	partOfSpeech	meaning	exampleSentence		
change	n.	변화	An attempt brings change to your life		
spelling	partOfSpeech	meaning	exampleSentence		
study	n.	연구	Studies found that people are selfish		
spelling	partOfSpeech	meaning	exampleSentence		
study	v.	공부하다	I study English everyday		
spelling	partOfSpeech	meaning	exampleSentence		
reliable	adj.	믿을 수 있는	She is a reliable coworker.		
spelling	partOfSpeech	meaning	exampleSentence		

quickly	adv.	빠르게	He quickly finished his homework

spelling	partOfSpeech	meaning	exampleSentence
—	V.	향상되다	The weather is improving.

temp

spelling	partOfSpeech	meaning	exampleSentence

count

	9
i	5
j	1

$j < 8$

words				spacing	length
spelling	partOfSpeech	meaning	exampleSentence	10	9
attempt	n.	시도	Making an attempt is a good thing.		
spelling	partOfSpeech	meaning	exampleSentence		
attempt	v.	시도하다	I will attempt to solve the problem		
spelling	partOfSpeech	meaning	exampleSentence		
Carefully	adv.	조심스럽게	please read the instructions carefully		
spelling	partOfSpeech	meaning	exampleSentence		
change	n.	변화	An attempt brings change to your life		
spelling	partOfSpeech	meaning	exampleSentence		
change	v.	바꾸다	People can change their habits.		
spelling	partOfSpeech	meaning	exampleSentence		
reliable	adj	믿을 수 있는	She is a reliable coworker.		
spelling	partOfSpeech	meaning	exampleSentence		
Study	n.	연구	Studies found that people are selfish		
spelling	partOfSpeech	meaning	exampleSentence		
Study	V	공부하다	I study English everyday.		
spelling	partOfSpeech	meaning	exampleSentence		
quickly	adv.	빠르게	He quickly finished his homework		

1
2
3
4
5
6
7
8
9

spelling	partOfSpeech	meaning	exampleSentence
——	v.	天气变好	The weather is improving

Temp

spelling	partOfSpeech	meaning	exampleSentence

count

9

;

10

;

6

테스트 시나리오

0. 적재하다.

1. attempt, 동사, 시도하다, I will attempt to solve the problem. 을 기재한다.

2. change, 동사, 바꾸다, People can change their habits. 를 기재한다.

3. study, 동사, 공부하다, I study English everyday. 를 기재한다.

4. study, 명사, 연구, Studies found that people are selfish. 을 기재한다.

5. attempt, 명사, 시도, Making an attempt is a good thing. 을 기재한다.

6. improve, 동사, 향상시키다, Regular practice improves your English. 을 기재한다.

7. change, 명사, 변화, An attempt bring change to your life. 를 기재한다.

8. reliable, 형용사, 믿을 수 있는, She is a reliable coworker. 를 기재한다.

9. credible, 형용사, 믿을 수 있는, The witness sare a credible explanation. 을 기재한다.

10. quickly, 부사, 빠르게, He quickly finished his homework. 를 기재한다.

11. attempt 로 찾는다.

12. 믿을 수 있는 으로 찾는다.

13. 6번째 뜻과 예문을 더 좋아지다, the weather is improving 으로 바꾼다.

14. 6번째 improve 를 지운다.

15. 정리하다.

16. 저장한다.

17. 끝내다.

```
#C 76조
// WordBook.c
# include <stdio.h>
# include <string.h>
# define MAX 10

typedef signed long int Long;

typedef struct _ word {
        char spelling [64];
        char partOfSpeech [8];
        char meaning [32];
        char exampleSentence [100];
} Word;

typedef struct _ WordBook {
        Word words [MAX];
        Long capacity;
        Long length;
} WordBook;

int main (int argc, char * argv[]);
Long Record ( WordBook *wordBook, char (*spelling), char (*partOfSpeech),
                                char(*meaning), char (*exampleSentence));
void FindMeaning( WordBook *wordBook, char (*meaning) , Long (* indexes), Long
                                                                    *count);
void FindSpelling ( WordBook *wordBook, char (*spelling) , Long (* indexes), Long
                                                                    *count);
Long Correct ( WordBook *wordBook, Long index, char (*meaning), char (*example
                                                                    Sentence));
Long Erase( WordBook *wordBook, Long index);
```

```
int main( int argc, char *argv[]) {
    WordBook wordBook = {
        { "", "", "", "" },
        { "", "", "", "" },
        { "", "", "", "" },
        { "", "", "", "" },
        { "", "", "", "" },
        { "", "", "", "" },
        { "", "", "", "" },
        { "", "", "", "" },
        { "", "", "", "" },
        { "", "", "", "" },
        MAX, 0
    };

    Long index;
    Long indexes[MAX];
    Long count;
    Long i=0;
```

```
index = Record( &wordBook, "attempt", "동사", "시도하다", "I will attempt to solve the problem.");
printf( "%s\t%s\t%s\t%s\n", wordBook.words[ index].spelling,
                            wordBook.words[ index].partOfSpeech,
                            wordBook.words[ index].meaning,
                            wordBook.words[ index].exampleSentence);

index = Record( &wordBook, "change", "동사", "바꾸다", "People can change their habits.");
printf( "%s\t%s\t%s\t%s\n", wordBook.words[ index].spelling,
                            wordBook.words[ index].partOfSpeech,
                            wordBook.words[ index].meaning,
                            wordBook.words[ index].exampleSentence);
```

```
Index = RecordC(&wordBook, "study", "동사", "영어공부하다", "I study English everyday.");
printfC("%s\t%s\t%s\t%s\n", wordBook.words[Index].spelling,
                            wordBook.words[Index].partOfSpeech,
                            wordBook.words[Index].meaning,
                            wordBook.words[Index].exampleSentence);

Index = RecordC(&wordBook, "study", "명사", "연구", "Studies found that people are selfish.");
printfC("%s\t%s\t%s\t%s\n", wordBook.words[Index].spelling,
                            wordBook.words[Index].partOfSpeech,
                            wordBook.words[Index].meaning,
                            wordBook.words[Index].exampleSentence);

Index = RecordC(&wordBook, "attempt", "명사", "시도", "Making an attempt is a good thing.");
printfC("%s\t%s\t%s\t%s\n", wordBook.words[Index].spelling,
                            wordBook.words[Index].partOfSpeech,
                            wordBook.words[Index].meaning,
                            wordBook.words[Index].exampleSentence);

Index = RecordC(&wordBook, "improve", "동사", "향상하다", "Regular practice improves your
printfC("%s\t%s\t%s\t%s\n", wordBook.words[Index].spelling,                    English.");
                            wordBook.words[Index].partOfSpeech,
                            wordBook.words[Index].meaning,
                            wordBook.words[Index].exampleSentence);

Index = RecordC(&wordBook, "change", "명사", "변화", "An attempt bring change to your life.");
printfC("%s\t%s\t%s\t%s\n", wordBook.words[Index].spelling,
                            wordBook.words[Index].partOfSpeech,
                            wordBook.words[Index].meaning,
                            wordBook.words[Index].exampleSentence);

Index = RecordC(&wordBook, "reliable", "형용사", "믿을 수 있는", "She is a reliable coworker.");
printfC("%s\t%s\t%s\t%s\n", wordBook.words[Index].spelling,
                            wordBook.words[Index].partOfSpeech,
                            wordBook.words[Index].meaning,
                            wordBook.words[Index].exampleSentence);
```

```
index = RecordC & wordBook, "credible", "형용사", "믿을 수 있는", "The witness gave a credible explanation");
printf("%s\t%s\t%s\n", wordBook.words[index].spelling,
                       wordBook.words[index].partOfSpeech,
                       wordBook.words[index].meaning,
                       wordBook.words[index].exampleSentence);

index = RecordC & wordBook, "quickly", "부사", "빠르게", "He quickly finished his homework.");
printf("%s\t%s\t%s\n", wordBook.words[index].spelling,
                       wordBook.words[index].partOfSpeech,
                       wordBook.words[index].meaning,
                       wordBook.words[index].exampleSentence);

FindMeaningC(&wordBook, "믿을 수 있는", indexes, &count);
i = 0;
while(i < count) {
    printf("%s\t%s\t%s\n", wordBook.words[indexes[i]].spelling,
    wordBook.words[indexes[i]].partOfSpeech, wordBook.words[indexes[i]].meaning,
                           wordBook.words[indexes[i]].exampleSentence);
    i++;
}

FindSpelling(&wordBook, "attempt", indexes, &count);
i = 0;
while(i < count) {
    printf("%s\t%s\t%s\n", wordBook.words[indexes[i]].spelling,
    wordBook.words[indexes[i]].partOfSpeech, wordBook.words[indexes[i]].meaning,
                           wordBook.words[indexes[i]].exampleSentence);
    i++;
}

index = Correct(&wordBook, 5, "명사", "The weather is improving.");
printf("%s\t%s\n", wordBook.words[index].meaning,
                   wordBook.words[index].exampleSentence);
```

```
index = Erase(&wordBook, 4);
if (index == -1)
    printf("삭제할 단어가 없습니다.\n");
}

Arrange(&wordBook);
i = 0;
while( i < wordBook.length) {
    printf("%s\t%s\t%s\t%s\n", wordBook.words[i].spelling,
            wordBook.words[i].partOfSpeech, wordBook.words[i].meaning,
            wordBook.words[i].exampleSentence);
    i++;
}
```

* Record

```
Long Record( WordBook *wordBook, char (*spelling), char (*partOfSpeech),
                                 char (*meaning), char (*exampleSentence)){

    Long index=-1;
    Long i=0;

    // 1. 철자, 품사, 뜻, 예문을 입력받는다.
    // 2. 빈 줄을 찾는다.
    while( i < wordBook->capacity &&
                        strcmp(wordBook->words[i].spelling,"")!=0){

        i++;
    }
    // 3. 찾아낸 빈 줄에 입력
    if( i < wordBook->capacity){
        // 3.1. 적절한 곳에 저장한다.
        index=i;
        strcpy(wordBook->words[index].spelling, spelling);
        strcpy(wordBook->words[index].partOfSpeech, partOfSpeech);
        strcpy(wordBook->words[index].meaning, meaning);
        strcpy(wordBook->words[index].exampleSentence, exampleSentence);
        wordBook->length++;
    }
    // 4. 저장한 줄의 위치를 기억한다.
    return index;
    // 5. 끝내다.
}
```

```
// FindMeaning
void FindMeaning(WordBook *wordBook, char (*meaning), Long (*indexes),
                                                    Long *count){
    Long i=0;
    Long j=0;
    *count=0;

    // 1. 뜻을 입력받는다.
    // 2. 저장되어있는 단어를 반복한다.
    while(i<wordBook->length){
        // 2.1. 뜻이 같으면
        if(strcmp(wordBook->wordList[i].meaning, meaning)==0){
            // 2.1.1. 개수를 센다.
            (*count)++;
            // 2.1.2. 인덱스를 기억한다.
            indexes[j]=i;
            j++;
        }
        i++;
    }
    // 3. 개수, 인덱스들을 출력한다.
    return count, indexes;
    // 4. 종료한다.
}
```

```
// findSpelling
void findSpelling( WordBook *wordBook, char (*spelling), long (*indexes),
                                                    long *count){
    long i=0;
    long j=0;
    *count=0;

    // 1. 단어를 입력받는다.
    // 2. 저장된단어만큼 반복한다.
    while( i<wordBook->length){
        // 2.1. 철자가 같으면
        if (strcmp(wordBook->wordList[i].spelling, spelling)==0){
            // 2.1.1. 개수를 센다.
            (*count)++;
            // 2.1.2. 인덱스를 기억한다.
            indexes[j]=i;
            j++;
        }
        i++;
    }
    // 3. 개수, 인덱스들을 돌려준다.
    return count, indexes;
    // 4. 끝내다.
}
```

// Correct

```
Long Correct(WordBook *wordBook, Long index, char (*meaning),
                                        char (*exampleSentence))
{
    // 1. 수정할 위치, 뜻, 예문을 입력받는다.
    // 2. 입력받은 뜻으로 수정한다.
    wordBook->words[index].meaning = meaning;
    // 3. 입력받은 예문으로 수정한다.
    wordBook->words[index].exampleSentence = exampleSentence;
    // 4. 수정된 위치를 출력한다.
    return index;
    // 5. 끝내다.
}
```

// Erase

```
Long Erase(WordBook *wordBook, Long index)
{
    // 1. 지울 위치를 입력받는다.
    // 2. 위치의 첫자에 지움표시를 한다.
    strcpy(wordBook->words[index].spelling, "-");
    // 3. 위치를 줄인다.
    index--;
    // 4. 위치를 출력한다.
    return index;
    // 5. 끝내다.
}
```

```
// Arrange
void Arrange(WordBook *wordBook){
    Word temp;
    Long i=0;
    Long j=0;
    Long count=0;

    // 1. 단어순으로 내림차순정렬한다.
    while(i< wordBook->length){
        j=i+1;
        while(j< wordBook->length){
            if(strcmp(wordBook->words[i].spelling,
                      wordBook.words[j].spelling)<0){
                temp=wordBook->words[i];
                wordBook->words[i]=wordBook->words[j];
                wordBook->words[j]=temp;
            }
            j++;
        }
        i++;
    }
    // 2. 같은 단어를 찾은 개수를 센다.
    i=0;
    while(i< wordBook->length &&
          strcmp(wordBook->words[i].spelling,"_")!=0){
        count++;
        i++;
    }
    // 3. 단어순 오름차순정렬한다. (대소문자구분)
    i=0;
    while(i<count){
        j=0;
        while(j<count-1){
            if(strcmp(wordBook->words[j].spelling,
                      wordBook->words[j+1].spelling)>0){
```

```
                temp= wordBook -> words[j];
                wordBook-> words[j] = wordBook -> words[j+1];
                wordBook-> words[j+1] = temp;
            }
            j++;
        }
        i++;
    }

    // 4. wordBook 을 정렬한다.
    // 5. 출력한다.
}
```

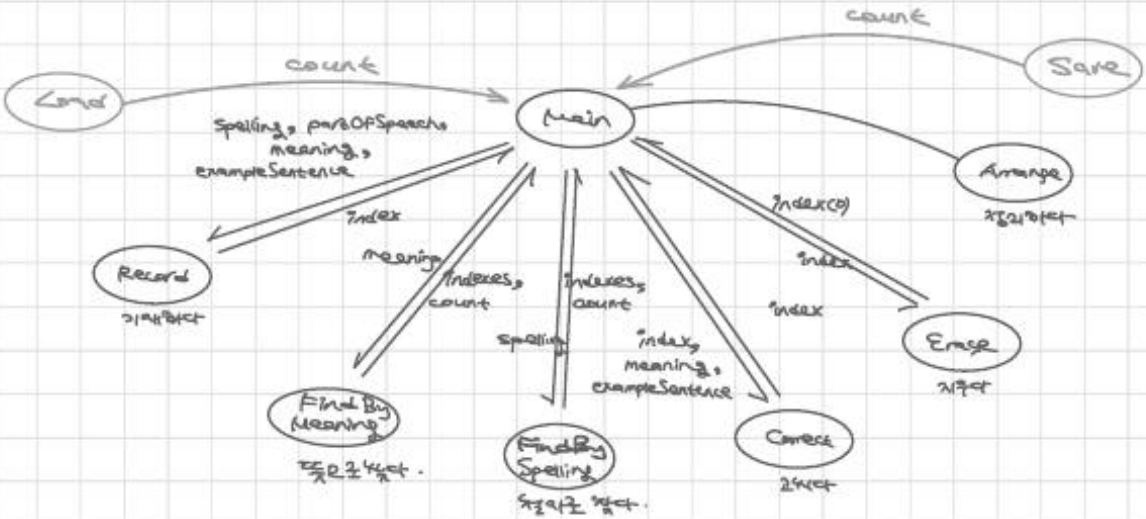
main.c 프로그램 시작 (main 함수), 메뉴, Load/Save

wordBook.c Record, Find, Correct, Erase .. 함수구현

wordBook.h 구조체, 상수, 함수원형 (선언)

4. 디스크 파일 설계

4.1. 시스템 다이어그램 – Load/Save 추가

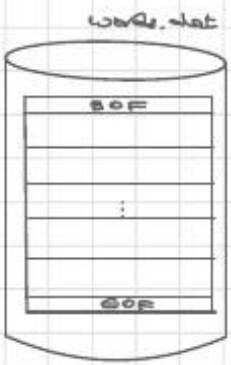


4.2. 상세설계

1. Load 상세설계

1.1. 모델 구축





1. 저장된 단어상 불러오기
2. 파일에서 단어들을 읽는다.
3. WordBook에 저장
4. count 를 센다.
5. count 출력
6. 끝내다.

1.2. 자료명세서

번호	한글	영문	자료유형	3번	비고
	단어장	WordBook	단어장	준비	파일
	단어	Word	단어		레코드
	철자	Spelling	문자열		필드
	품사	partOfSpeech	문자열		필드
	뜻	meaning	문자열		필드
	예문	exampleSentence	문자열		필드
	할당량	Capacity	정수		
	저장량	length	정수		
	최댓값	MAX	정수	상수	10
	단어배열	words	단어배열	입력	
	철자	Spelling	문자열	입력	
	품사	partOfSpeech	문자열	입력	
	뜻	meaning	문자열	입력	
	예문	exampleSentence	문자열	입력	
	할당량	Capacity	정수	처리	
	저장량	length	정수	처리	
	반복제어	i	정수	육가	

1.3. 처리과정.

1. 파일의 끝이 아닌동안 반복한다.
   1.1. 철자, 품사, 뜻, 예문을 입력받는다.
   1.2. 단어들에 적는다.
   1.3. 개수를 센다.
2. 개수를 출력한다.
3. 끝내다.


START
wordBook, word, ? = 1
read word.spelling, word.partOfSpeech
word.meaning, word.exampleSentence
while (NOT EOF)
wordBook.words(?) = word
? = ? + 1
wordBook.length =
wordBook.length + 1
read word.spelling,
word.partOfSpeech word.meaning,
word.exampleSentence
print wordBook.length
STOP

2. Save 상세설계

Save → count



		B O F	
attempt	동사	시도하다	I will attempt to solve the problem.
attempt	명사	시도	Making an attempt is a good thing.
change	동사	바꾸다	People can change their habits.
change	명사	변화	An attempt brings change to your life.
credible	형용사	믿을수있는	The witness gave a credible explanation.
reliable	형용사	믿을수 있는	She is a reliable coworker.
study	동사	공부하다	I study Spanish everyday.
study	명사	연구	Studies found that people are selfish
quickly	부사	빠르게	He quickly finished his homework.
——	동사	더 좋아지다	The weather is improving
		E O F	

2.2. 자료명세서

번호	한글	영문	자료유형	3분	비고
	단어장	WordBook	단어장	준비	파일
	단어	Word	단어		레코드
	철자	Spelling	문자열		필드
	품사	partOfSpeech	문자열		필드
	뜻	meaning	문자열		필드
	예문	exampleSentence	문자열		필드
	할당량	capacity	정수		
	저장량	length	정수		
	저장량	length	정수	보관	
	단어배열	words	단어배열	입력	
	철자	Spelling	문자열	입력	
	품사	partOfSpeech	문자열	입력	
	뜻	meaning	문자열	입력	
	예문	exampleSentence	문자열	입력	
	할당량	capacity	정수	처리	
	반복제어	i	정수	화가	

1. 사용량산출 반복한다.
   1.1. 철자, 품사, 뜻, 예문을 출력한다.
2. 저장량을 출력하다.
3. 끝내다.

```
START
wordBook, i=1
while(i≤ wordBook.length)
    print wordBook.words(i).
    spelling, wordBook.words(i).
    partOfSpeech, wordBook.words(i)
    meaning, wordBook.words(i).
    exampleSentence
    i=i+1
print wordBook.length
STOP
```

```
#Load
Long Load(WordBook *wordBook){
    Word word;
    Long i=0;
    size_t size;

    FILE *file;
    file=fopen("words.dat","rb");

    if(file!=NULL){
        // 1.1. 철자, 품사, 뜻, 예문을 입력받는다.
        size=fread(&word, sizeof(word), 1, file);

        // 1. 파일의 끝이 아닐동안 반복한다.
        while(!feof(file)&&size!=0){
            // 1.2. 단어를 저장한다.
            wordBook->words[i]=word;
            i++;
            // 1.3. 개수증가
            wordBook->length++;
            // 1.1. 철자, 품사, 뜻, 예문을 입력받는다.
            size=fread(&word, sizeof(word), 1, file);
        }
        // 2. 개수감소
        // 3. 닫는다.
        fclose(file);
    }
    return wordBook->length++;
}
```

// Save

```
Long Save(WordBook *wordBook){
    Long i=0;

    FILE *file;
    file=fopen("Words.dat","wb");

    if(file!=NULL){
        while(i<wordBook->length){ // 1. 사용단어수를 반복한다.
            // 1.1. 단어와 뜻과, 뜻, 예문을 출력한다.
            fwrite(wordBook->words+i, sizeof(Word), 1, file);
            i++;
        }
        // 2. 파일을 출력한다.
        // 3. 끝내다.
        fclose(file);
    }
    return wordBook->length;
}
```

1. 메뉴설계

0123456789012345678901234567890123456789012345678901234567890123456789

0
1
2
3
4
5
6
7
8
9
0
1
2
3
4
5
6
7
8
9
0
1
2
3
4

단어장 version 0.05 - 설자조

==========

[1] 찾 기

[2] 고치기

[3] 지우기

메뉴를 선택하십시오.

80-20=60%2=70

X

2

1. 메뉴설계

0123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

0
1
2
3
4
5
6
7
8
9
0
1
2
3
4
5
6
7
8
9
0
1
2
3
4

80-17 = 63%2 = 36

- 달여장 version 0.05
- ==========
- [1] 기재하기
- [2] 관리하기 → 개념이해X
  관리하다는
  기능자체가 X
- [3] 정리 하기
- [4] 전체보기
- [0] 끝내기
- ----------
- 메뉴를 선택하십시오.

24-11 = 13%2 = 6

24-9 = 15%2 = 6

1. 메뉴설계

0123456789012345678901234567890123456789012345678901234567890123456789

0
1
2
3
4
5
6
7
8
9
0
1
2
3
4
5
6
7
8
9
0
1
2
3
4

달여장 version 0.05 - [2] 관리하기

==========

[1] 뜻으로, 찾,,,,기

[2] 뜻으로, 고,치,기

[3] 뜻으로, 지,우,기

[4] 철자로, 찾,,,,기

[5] 철자로, 고,치,기

[6] 철자로, 지,우,기

메뉴를 선택하십시오.

80-20=60%2=70

2

X

5. 사용자 인터페이스

5.1. 메뉴 설계

5.2. 화면 설계

5.3. 제어구조 스케치

5.4. 시스템 다이어그램

* 사용자 인터페이스 (User Interface, UI)

정의: 프로그램이 제공하는 기능들을 출력하고 선택하는 방식 (메뉴, Menu)

사용자가 데이터를 입력받는 방식 (입력방식, Mask)

사용자가 데이터를 출력하는 방식 (출력방식, Format)

* 구별방식

1. 문자기반 UI – Character User Interface (CUI)
2. 그림기반 UI – Graphic User Interface (GUI)

* console (콘솔): 입력과 동시에 출력이 가능한 장치 → terminal 같은 존재

크기: 80(열) × 25(행)

25 − 11 = 13.
13 % 2 = 6

* prompt (프롬프트):

"메뉴를 선택하십시오!"

↳ 9×2=18
3×1=3 → 21.

80 − 21 = 59.
59 % 2 = 29.

* 알고리즘: 시작 → 끝, 유한성

↕

* 프로그램: 무한성.

끝내기 명령 전까지 끝내면 X

↓

∴ 끝내기 있어야 함
명령

1. 메뉴설계 * 메인메뉴

0123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890

0	\n
1	\n
2	\n
3	\n
4	\n
5	\n
6	달여장 version 0.05
7	==========
8	[1] 기재하기
9	[2] 찾ㅣㅣㅣ기
0	[3] 고ㅣ식ㅣ기
1	[4] 지ㅣ우ㅣ기
2	[5] 정리하기
3	[6] 전체보기
4	[0] 끝ㅣ내ㅣ기
5	----------
6	메뉴를 선택하십시오.
7	24-11 = 13%2=6
8	24-9 = 15%2=6
9	
0	
1	
2	
3	
4	

1. 메뉴설계 * 서브메뉴

0123456789012345678901234567890123456789012345678901234567890123456789

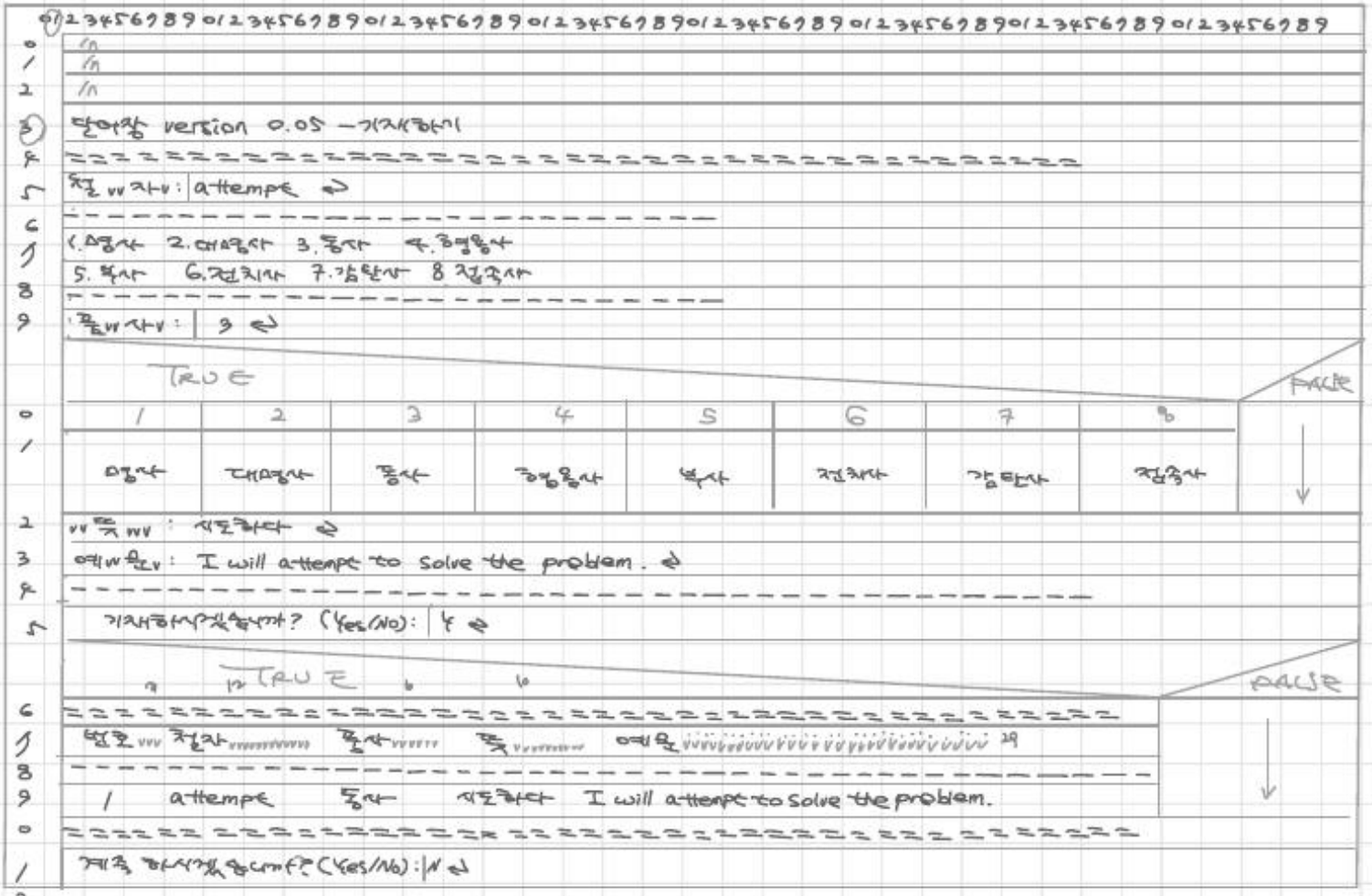
0	/n
1	/n
2	/n
3	/n
4	/n
5	/n
6	/n
7	/n
8	→ 단어장 version 0.05 - 찾vvvv기
9	==========
0	[1] 뜻으로 v찾기 FindByMeaning
1	[2] 철자로 v찾기 FindBySpelling
2	[0] 상위vvv 메뉴
3	----------
4	메뉴를 선택하십시오.
5	
6	
7	
8	
9	
0	
1	
2	
3	
4	

80-20=60%2=70

2

24-7=17%2=8

2. 화면설계 (Screen Design) - 1. 기차예약


달여왕 version 0.05 - 기차예약
첫 vv 자v : attempt
1.명사 2.대명사 3.동사 4.형용사
5.부사 6.전치사 7.감탄사 8.접속사
품vv 사v : 3
TRUE
FALSE
1 2 3 4 5 6 7 8
명사 대명사 동사 형용사 부사 전치사 감탄사 접속사
vv 뜻 vvv : 시도하다
예w 문v : I will attempt to solve the problem.
기차예약하시겠습니까? (Yes/No) : Y
TRUE
FALSE
번호 vvv 철자 vvvvvvvvvv 품사 vvvvvv 뜻 vvvvvvvv 예문 vvvvvvvvv

2. 화면설계 (Screen Design) - 2. 실습 - ① 텍스트로 찾기

```
0123456789012345678901234567890123456789012345678901234567890123456789012345678
0
1
2
3
4
5
6
① 찾아줘 version 0.05 - 텍스트로 찾기
8
9   찾을 수 있는 단어 목록 :
0
1   번호 vvv  단어 vvvvvvvvvv  철자 vvvvvvvvvvv  품사 vvvvv  예문 vvvvvvvvvv
```

2. 화면설계 (Screen Design) — 2. 찾기 — ② 철자로 찾기

나가기 넣음.

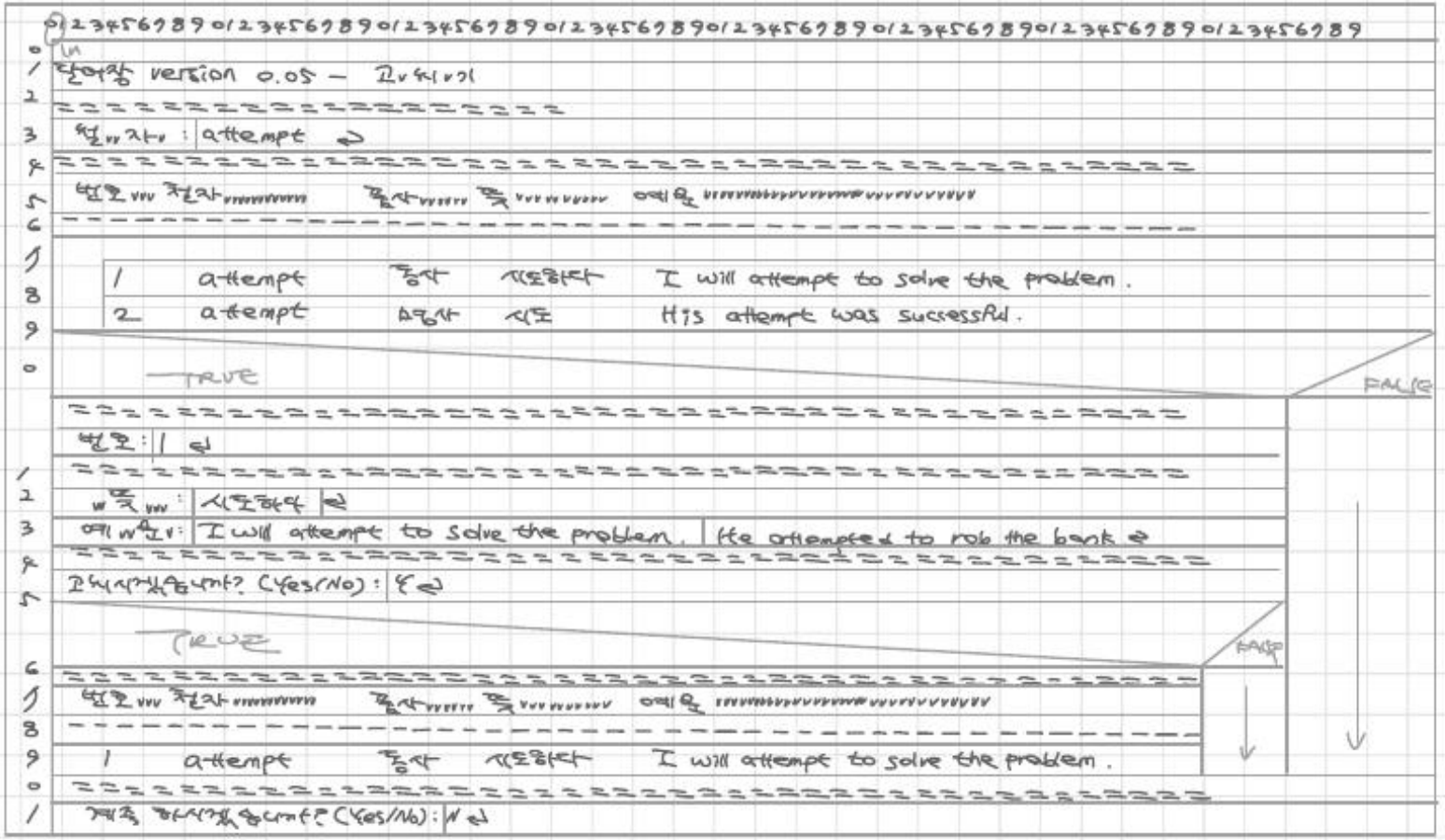
```
0123456789012345678901234567890123456789012345678901234567890123456789
\n
\n
\n
\n
\n
\n
\n
단어장 version 0.05 — 철자로 찾기
==========
철자로 찾기: | attempt ↵
==========
번호 철자 품사 뜻 예문
----------
  1   attempt   동사   시도하다   I will attempt to solve the problem.
  2   attempt   명사   시도       Making an attempt is a good thing.
==========
계속 하시겠습니까? (Yes/No): | N ↵
```

2. 화면설계 (Screen Design) — 3. 고시기

```
0123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789
단어장 version 0.05 — 고시기
==========================================
찾자 : attempt ↵
==========================================
번호 철자          품사      뜻          예문
------------------------------------------
  1   attempt   동사   시도하다   I will attempt to solve the problem.
  2   attempt   명사   시도       His attempt was successful.
```

TRUE / FALSE

```
==========================================
번호 : 1 ↵
==========================================
뜻 : 시도하다 ↵
예문 : I will attempt to solve the problem. | He attempted to rob the bank ↵
==========================================
고시시겠습니까? (Yes/No) : Y ↵
```

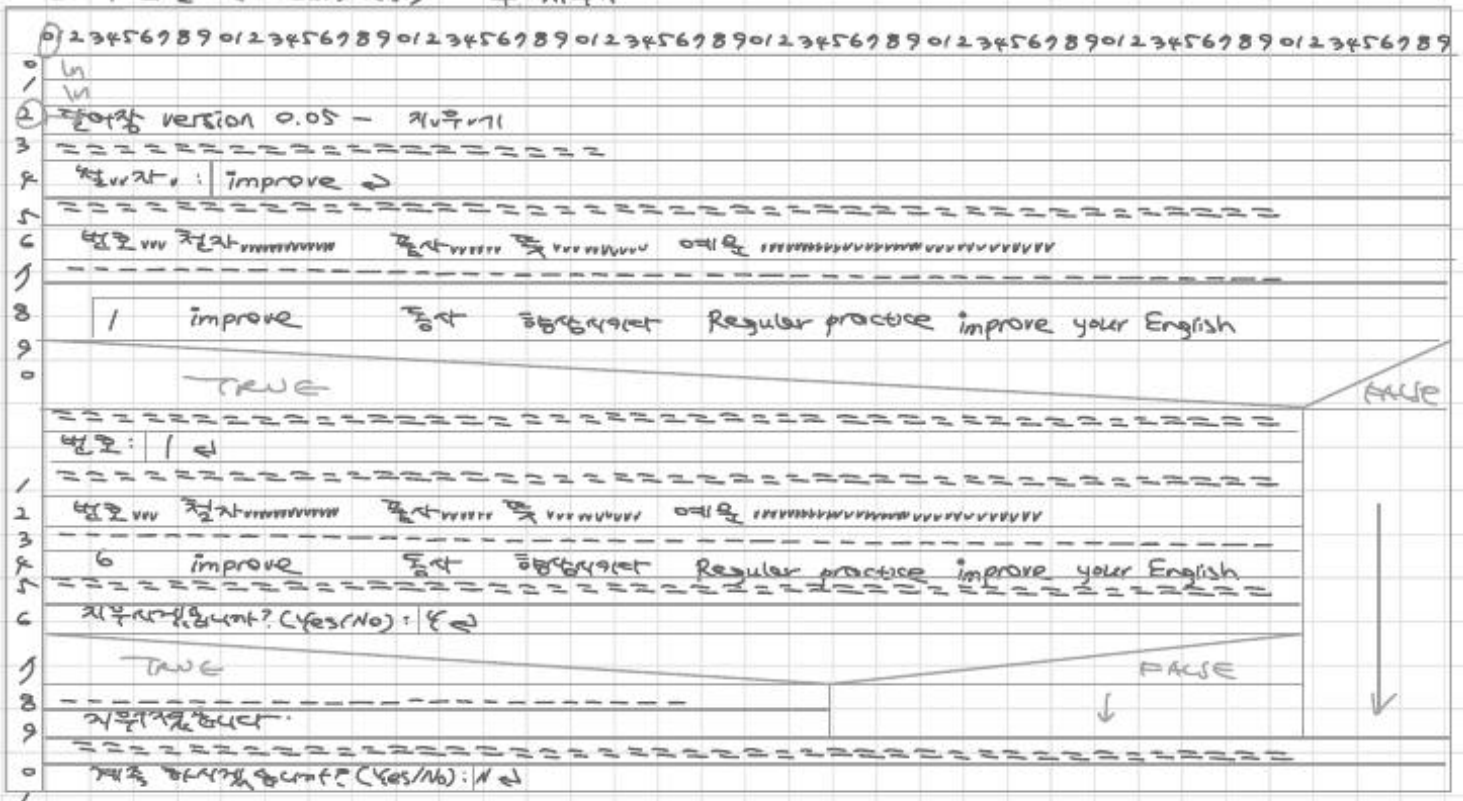


TRUE / FALSE

```
==========================================
번호 철자          품사      뜻          예문
------------------------------------------
  1   attempt   동사   시도하다   I will attempt to solve the problem.
==========================================
계속 하시겠습니까? (Yes/No) : N ↵
```

과~과=7%2=1

2. 화면설계 (Screen Design) — 4 지우기


0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9
4 프로그램 version 0.05 — 지우기
명령어 : improve
명령 vvv 철자vvvvvvvvvv 품사vvvvvv 뜻 vvv vvvvvv 예문 vvvvvvvvv

2. 화면설계 (Screen Design) – 5. 정리하기

0123456789 0123456789 0123456789 0123456789 0123456789 0123456789 0123456789 0123456789

\n
\n
\n
\n

※ → 단어장 version 0.05 – 정리하기

==========

번호 vvv 철자 vvvvvvvvvv 품사 vvvvv 뜻 vvvvvvvv 예문 vvvvvvvvv

2. 화면설계 (Screen Design) — 6. 전체내용

0123456789 0123456789 0123456789 0123456789 0123456789 0123456789 0123456789 0123456789

단어장 version 0.05 — 전체내용

번호 vvv 철자 vvvvvvvvvv 품사 vvvvv 뜻 vvvvvvvvv 예문 vvvvvvvvv

단어장 version 0.05 - 뜻으로 찾기

뜻 : 사용할 수 있는

번호 뜻 철자 품사 예문

1 사용할 수 있는 reliable 형용사 she is ~.

2 사용할 수 있는 credible 형용사 the witness

계속하시겠습니까? (yes/no) : no

단어장 version 0.05 - 검색기

찾을 단어: attempt

번호 철자 품사 뜻 예문

1 attempt ~

2 attempt ~

번호 : 2

계속하시겠습니까? (yes/no) : no

단어장 version 0.05 - 수정기

수정할 단어: improve

번호 철자 품사 뜻 예문

1 improve 동사 향상시키다 Regular ~.

번호 : 1

번호 철자 품사 뜻 예문

1 improve 동사 향상시키다 Regular ~.

수정하시겠습니까? (yes/no) : no

계속하시겠습니까?

단어장 version 0.05 - 지우기

철자: improve ↵

번호 철자 품사 뜻 예문

1 improve 동사 향상시키다 Regular ~.

번호: 1 ↵

번호 철자 품사 뜻 예문

6 improve 동사 향상시키다 Regular ~.

지우시겠습니까? (Yes/No) : Y ↵

지워졌습니다.

계속하시겠습니까?

단어장 version 0.05 - 정리하기

번호 철자 품사 뜻 예문

1 attempt 동사 시도하다

2 attempt 명사 시도

3 change 동사 바꾸다

4 change 명사 변화

⋮

10 — 부사 빠르게

아무 키나 누르세요!

단어장 version 0.05 — 기재하기

==========

첫 자: improve ↵

---

1. 명사 2. 동사 3. 형용사 4. 부사

5. 전치사 6. 대명사 7. 접속사 8. 감탄사

품 사: 2 ↵

뜻: 향상시키다↵

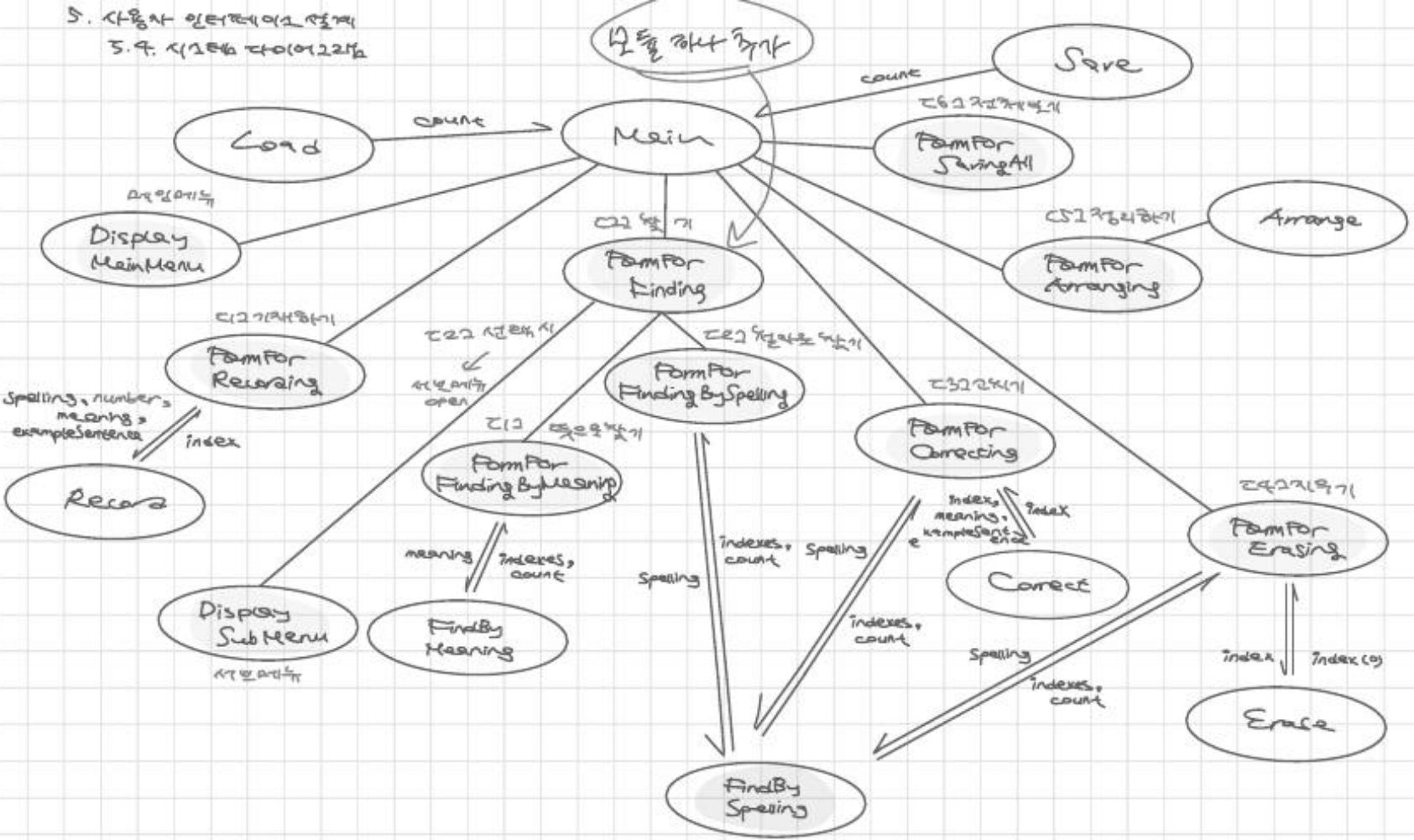
예문: I will improve my English ↵

기재하시겠습니까? (Yes/No): N ↵

==========

계속하시겠^

5. 사용자 인터페이스 설계
5.4. 시스템 다이어그램



* main → form → 실제 기능함수
↓
사용자가 어떤 기능을
사용할지만 열정
- 전체 흐름 담당
- 메인메뉴 선택만 처리

* DisplayMainMenu
- 메인메뉴 출력만 담당

* FormForFinding
- 드르그 찾기 선택시 → * DisplaySubMenu
- 서브메뉴 출력
- FormForFindingByMeaning
- FormForFindingBySpelling
→ 둘으로 찾기 / 철자로 찾기

```
void FormForFinding(WordBook *wordBook) {
    int menu;

    DisplaySubMenu();
    scanf("%d", &menu);

    Switch (menu) {
        case 1: FormForFindingByMeaning(wordBook);
            break;
        case 2: FormForFindingBySpelling(wordBook);
            break;
        case 0:
            break;
    }
}
```

Form.c

```
#define MAX 10

void DisplayMenu();
void FormForRecording(WordBook *wordBook);
void FormForFinding(WordBook *wordBook);
void DisplaySubMenu();
void FormForFindingByMeaning(WordBook *wordBook);
void FormForFindingBySpelling(WordBook *wordBook);
void FormForCorrecting(WordBook *wordBook);
void FormForErasing(WordBook *wordBook);
void FormForArranging(WordBook *wordBook);
void FormForSavingAll(WordBook *wordBook);

int main(int argc, char *argv[]){
    WordBook wordBook = {
        { {"", "", "", ""},
          {"", "", "", ""},
          {"", "", "", ""},
          {"", "", "", ""},
          {"", "", "", ""},
          {"", "", "", ""},
          {"", "", "", ""},
          {"", "", "", ""},
          {"", "", "", ""},
          {"", "", "", ""} },
        MAX, 0
    };

    int menu;
```

```
count = Load( &wordBook);
DisplayMenu();
rewind( stdin);
Scanf("%d", &menu);
while ( menu!=0){
        Switch (menu) {
                case 1: FormForRecording( &wordBook);
                        break;
                Case 2: FormForFinding( &wordBook);
                        break;
                case 3: FormForCorrecting( &wordBook);
                        break;
                case 4: FormForErasing ( &wordBook);
                        break;
                case 5: FormForArranging( &wordBook);
                        break;
                case 6: FormForSavingAll (&wordBook);
                        break;
                default:
                        break;
        }
        DisplayMenu();
        rewind (Stdin);
        Scanf("%d", &menu);
}
```

```
// form.c
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#define MAX 10

typedef signed long int Long;

typedef struct _word{
        char spelling[64];
        char partOfSpeech[8];
        char meaning[64];
        char exampleSentence[128];
}Word;

typedef struct _wordBook{
        Word words[MAX];
        Long capacity;
        Long length;
}WordBook;

int main(int argc, argv[]);
Long Load(WordBook *wordBook);
Long Record(WordBook *wordBook, char(*spelling), char(*partOfSpeech),
                                char(*meaning), char(*exampleSentence));

void FindByMeaning(WordBook *wordBook, char(*meaning), Long (*indeces), Long *count);
void FindBySpelling(WordBook *wordBook, char(*spelling), Long (*indeces), Long *count);
Long Correct(WordBook *wordBook, Long index, char(*meaning), char(*exampleSentence));
Long Erase(WordBook *wordBook, Long index);
void Arrange(WordBook *wordBook);
Long Save(WordBook *wordBook);
```

```
void DisplayMenu();
void FormForRecording(WordBook *wordBook);
void FormForFinding(WordBook *wordBook);
void DisplaySubMenu();
void FormForFindingByMeaning(WordBook *wordBook);
void FormForFindingBySpelling(WordBook *wordBook);
void FormForCorrecting(WordBook *wordBook);
void FormForErasing(WordBook *wordBook);
void FormForArranging(WordBook *wordBook);
void FormForSavingAll(WordBook *wordBook);

int main(int argc, char* argv[2]){
    WordBook wordBook = {
        { {"","","","",
          {"","","","",
          {"","","","",
          {"","","","",
          {"","","","",
          {"","","","",
          {"","","","",
          {"","","","",
          {"","","","",
          {"","","","",
          {"","","","",
          {"","","","",
          MAX, 0
        };

    int menu;

    count = Load(&wordBook);
    DisplayMainMenu();
    rewind(stdin);
    Scanf("%d", &menu);
    while (menu!=0){
        Switch(menu){
            case(: FormForRecording(&wordBook);
                break;
```

```
            case 2: FormForFinding(&wordBook);
                    break;
            case 3: FormForCorrecting(&wordBook);
                    break;
            case 4: FormForErasing(&wordBook);
                    break;
            case 5: FormForArranging(&wordBook);
                    break;
            case 6: FormForSavingAll(&wordBook);
                    break;
            default:
                    break;
        }
        DisplayMainMenu();
        rewind(stdin);
        scanf("%d", &menu);
    }
    Save(&wordBook);
    return 0;
}
```

```
void DisplayMenu(){
    system("clear");
    printf("\n\n\n\n\n\n");
    printf("\t\t\t 단어장 Version 0.05\n");
    printf("\t\t\t==========================\n");
    printf("\t\t\t 1그 기재하기\n");
    printf("\t\t\t 2그 찾,,,,기\n");
    printf("\t\t\t 3그 고,씨,기\n");
    printf("\t\t\t 4그 지,우,기\n");
    printf("\t\t\t 5그 정리하기\n");
    printf("\t\t\t 6그 전체보기\n");
    printf("\t\t\t 7그 끝,내,기\n");
    printf("\t\t\t ------------------------\n");
    printf("\t\t\t 메뉴를 선택하십시오.");
}

void FormForRecording(WordBook *wordBook){
    char spelling[64];
    char partOfSpeech[8];
    char meaning[64];
    char exampleSentence[128];
    char going='Y';
    char recording;
    Long index;
    Long number;

    while( going=='Y' || going=='y'){
        system("clear");
        printf("\n\n\n");
        printf("단어장 Version 0.05- 기재하기\n");
        printf("==========================\n");
        printf("철,자,:"); rewind(stdin); scanf("%s", spelling);
        printf("------------------------\n");
        printf("1.명사는 2.대명사는 3.동사는 4.형용사\n");
        printf("5.부사는 6.전치사는 7.감탄사는 8.접속사\n");
```

```
printf("----------------------------------------\n");
printf("품사 : "); rewind(stdin); scanf("%d", number);
switch (number) {
        case 1: strcpy(partOfSpeech, "명사");
                break;
        case 2: strcpy(partOfSpeech, "대명사");
                break;
        case 3: strcpy(partOfSpeech, "동사");
                break;
        case 4: strcpy(partOfSpeech, "형용사");
                break;
        case 5: strcpy(partOfSpeech, "부사");
                break;
        case 6: strcpy(partOfSpeech, "전치사");
                break;
        case 7: strcpy(partOfSpeech, "감탄사");
                break;
        case 8: strcpy(partOfSpeech, "접속사");
                break;
        default:
                break;


printf("뜻 : "); rewind(stdin); scanf("%s", meaning);
printf("예문 : "); rewind(stdin); gets(exampleSentence);
printf("----------------------------------------\n");
printf("기록하시겠습니까? (Yes/No) : ");
rewind(stdin); recording = getchar();
if( recording == 'Y' || recording == 'y') {
    index = Record(wordBook, spelling, partOfSpeech, meaning,
                                                exampleSentence);
    printf("========================\n");
    printf("번호 철자 품사 뜻 예문
                                                    \n");

    printf("----------------------------------------\n");
    printf("%d\t%s\t%s\t%s\t%s\n", index+1,
```

```
        wordBook->words[index2].spelling,
        wordBook->words[index2].partOfSpeech,
        wordBook->words[index2].meaning,
        wordBook->words[index2].exampleSentence);
    }
    printf("==========\n");
    printf("계속 하시겠습니까? (Yes/No) : ");
    rewind(stdin); going = getchar();
  }
}
```

```
void FormPerFinding(WordBook *wordBook){
    int menu;

    DisplaySubMenu();
    rewind(stdin);
    scanf("%d", &menu);

    while(menu!=0){
        switch(menu){
            case 1: FormPerFindingByMeaning(wordBook);
                break;
            case 2: FormPerFindingBySpelling(wordBook);
                break;
            default:
                break;
        }
        DisplaySubMenu();
        rewind(stdin);
        scanf("%d", &menu);
    }
}
```

```
void FormForFindingByMeaning(WordBook *wordBook){
    char meaning[64];
    Long indexes[MAX];
    Long count;
    char going='Y';
    char finding;
    Long i;

    while(going=='Y'||going=='y'){
        system("clear");
        printf("\n\n\n\n\n\n");
        printf("단어장 Version 0.05 - 뜻으로 찾기\n");
        printf("==========================\n");
        printf("뜻 : "); rewind(stdin); scanf("%s", meaning);
        printf("==========================\n");
        printf("번호 뜻 철자 품사 예문");
        printf("-----------------------\n");
        FindByMeaning(wordBook, meaning, indexes, &count);

        i=0;
        while(i<count){
            printf("%d\t%s\t%s\t%s\t%s\n", i+1,
                wordBook->words.meaning[indexes[i]],
                wordBook->words.spelling[indexes[i]],
                wordBook->words.partOfSpeech[indexes[i]],
                wordBook->words.exampleSentence[indexes[i]]);
            i++;
        }
        printf("==========================\n");
        printf("더 계속 하시겠습니까? (Yes/No) :");
        rewind(stdin); going = getchar();
    }
}
```

```
void FormForFindingBySpelling (WordBook *wordBook){
    char spelling [64];
    Long indexes [MAX];
    Long count;
    char going = 'Y';
    char finding;
    Long i;

    while( going=='Y'|| going=='y'){
        system('clear");
        printf("\n\n\n\n\n\n\n");
        printf("단어장 Version 0.05 - 철자로 찾기\n");
        printf("==========================\n");
        printf("철자:"); rewind( stdin); scanf('%s", spelling);
        printf("==========================\n");
        printf('번호 철자 품사 뜻 예문 );
        printf("-------------------------\n");
        FindBySpelling(wordBook, spelling, indexes, &count);

        i=0;
        while( i< count){
            printf("%d\t %s\t %s\t %s\t %s\n", i+1,
                wordBook->words.spelling [indexes [i]],
                wordBook->words.partOfSpeech [indexes [i]],
                wordBook->words.meaning [indexes [i]],
                wordBook->words.exampleSentence [indexes [i]]);
            i++;
        }
        printf("==========================\n");
        printf("계속 하시겠습니까? (Yes/No) :")    ;
        rewind(stdin); going = getchar();
    }
}
```

```
void FormForCorrecting(WordBook *wordBook){
    char spelling[64];
    char meaning[64];
    char exampleSentence[128];
    Long indexes[MAX];
    Long count;
    Long index;
    Long number;
    char going='Y';
    char correcting;
    Long i;

    while(going=='Y'||going=='Y'){
        system("clear");
        printf("\n");
        printf("단어장 version 0.05-2ᄂᆞᆫᄂᆡᆫ\n");
        printf("==========================\n");
        printf("찾ᄂᆞᆫ:"); rewind(stdin); scanf("%s", spelling);
        printf("==========================\n");
        printf("번호 찾아 풀이 풀이 예문");
        printf("-------------------------\n");

        FindBySpelling(wordBook, spelling, indexes, &count);

        i=0;
        while(i<count){
            printf("%d\t%s\t %s\t %s\t %s\t", i+1,
                wordBook->words.spelling[indexes[i]],
                wordBook->words.partOfSpeech[indexes[i]],
                wordBook->words.meaning[indexes[i]],
                wordBook->words.exampleSentence[indexes[i]]);
            i++;
        }
        if(count>0){
            printf("==========================\n");
            printf("찾ᄂᆞᆫ:"); rewind(stdin); scanf("%d", &number);
```

```
    index= indexes[number-1];
    printf("=============================\n");
    printf("[illegible]: %s", wordBook->words[index].meaning);
    rewind(stdin); scanf("%s", meaning);

    if(strcmp(meaning,"")==0){
        strcpy(meaning, wordBook->words[index].meaning);
    }

    printf("[illegible]: %s", wordBook->words[index].exampleSentence);
    rewind(stdin); gets(exampleSentence);

    if(strcmp(exampleSentence,"")==0){
        strcpy(exampleSentence, wordBook->words[index].exampleSentence);
    }
    printf("=============================\n");
    printf("[illegible]? (Yes/No):");
    rewind(stdin); correcting= getchar();

    if(correcting=='Y'||correcting='y'){
        index= Correct(wordBook, index, meaning, exampleNumber);
        printf("=============================\n");
        printf("[illegible]");
        printf("-----------------------------\n");
        printf("%d\t%s\t %s\t %s\t %s\t", index+1,
            wordBook->words.meaning[index],
            wordBook->words.spelling[index],
            wordBook->words.partOfSpeech[index],
            wordBook->words.exampleSentence[index]);
    }
}
printf("=============================\n");
printf("[illegible]? (Yes/No):");
rewind(stdin); going= getchar();
}

}
```

```
void RemforErasing(WordBook * wordBook){
    char spelling[64];
    Long index;
    Long indexes[MAX];
    Long number;
    Long count;
    Long i;
    char erasing;
    char going='Y';

    while ( going=='Y'|| going=='y'){
        System("clear");
        printf("\u\u");
        printf("단어장 version 0.05-지우기\u");
        printf ("=========================\n");
        printf("철자:"); rewind(stdin); scanf("%s", spelling);
        printf ("=========================\n");
        printf("번호 철자 품사 뜻 예문");
        printf("-------------------------\n");

        FindBySpelling(wordBook, spelling, indexes, &count);

        i=0;
        while( i< count){
            printf ("%d\t%s\t %s\t %s\t %s\t", i++,
                wordBook->words.spelling[indexes[i]],
                wordBook->words.partOfSpeech[indexes[i]],
                wordBook->words.meaning[indexes[i]],
                wordBook->words.exampleSentence[indexes[i]]);
            i++;
        }
```

```
if (count > 0) {
    printf("=============================\n");
    printf("번호: "); rewind(stdin); scanf("%d", &number);
    index = indexes[number-1];
    printf("=============================\n");
    printf("번호 철자 품사 뜻 예문");
    printf("-----------------------------\n");

    printf("%d\t%s\t%s\t%s\t%s\n", index+1,
      wordBook -> words[index].spelling,
      wordBook -> words[index].partOfSpeech,
      wordBook -> words[index].meaning,
      wordBook -> words[index].exampleSentence);
    printf("=============================\n");
    printf("지우시겠습니까?(Yes/No):");
    rewind(stdin); erasing = getcher();
    if( erasing == 'Y' || erasing == 'y') {
        index = Erase(wordBook, index);
        printf("-----------------------------\n");
        printf("지워졌습니다.\n");
    }
}
printf("=============================\n");
printf("계속하시겠습니까?(Yes/No):");
rewind(stdin); going = getcher();
}
}
```

```
void formForArranging(WordBook * wordBook){
    Long i=0;
    Long count;

    Arrange(wordBook);
    while (i< wordBook->length){
        system("clear");
        printf("\n\n\n\n");
        printf("단어장 Version 0.05 - 정리하기\n");
        printf("=========================================\n");
        printf("번호 철자 품사 뜻 예문 ");
        printf("-----------------------------------------\n");

        count=1;
        while (i< addressBook->length && count <= MAX){
            printf("%d\t%s\t%s\t%s\t%s\n", i+1,
            wordBook->words[i].spelling, wordBook->words[i].partOfSpeech,
            wordBook->words[i].meaning,
             wordBook->words[i].exampleSentence);

            i++;
            count++;
        }
        printf("=========================================\n");
        printf("아무 키나 누르세요!");
        rewind(stdin); getcher();
```

```
void FormForSavingAll (WordBook * wordBook){
    Long i=0;
    Long count;

    Arrange(wordBook);
    while (i< wordBook-> length) {
        system("clear");
        printf("\n\n\n\n");
        printf(" 단어장 Version 0.05 - 전체보기\n");
        printf("=========================================\n");
        printf("번호 ..... 철자 ..... 품사 ..... 뜻 ..... 예문 ..... );
        printf("-----------------------------------------\n");

        count=1;
        while (i< addressBook-> length && count <= MAX){
            printf(" %d\t%s\t %s\t %s\t %s\n", i+1,
                wordBook -> words[i]. spelling,
                wordBook -> words[i]. partOfSpeech,
                wordBook -> words[i]. meaning,
                wordBook -> words[i]. exampleSentence);
                i++;
                count++;
        }
        printf("=========================================\n");
        printf(" 아무 키나 누르세요!");
        rewind(stdin); getcher();
```

ㅂ
ㅂ
ㅂ
ㅂ

단어장 Version 0.05- 전체보기

번호	철자	동사	뜻	예문
1	attempt	명사	시도	~
2	attempt	동사	시도하다	~
3	change	명사	변화	~
4	change	동사		~
5	credible	형용사	믿을 수 있는	
6	reliable	〃	〃	~
7	study	동사	공부하다	~
8	study	명사	연구	~
9	quickly	부사	빠르게	~
10	—	형용사		~

9 = 10
count = 11

아무리나 누르세요!