

# ReRe

Rememberance  
Repository

01



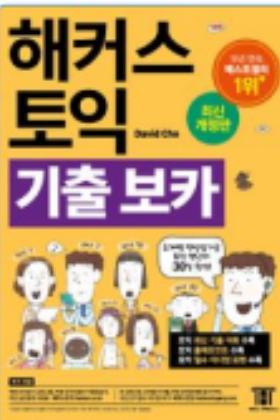
망각 곡선이란? 뤼루 사용법 내 정보



로그인

회원가입

## • 기본 카드북



해커스 토익 기출 보카

뤼루

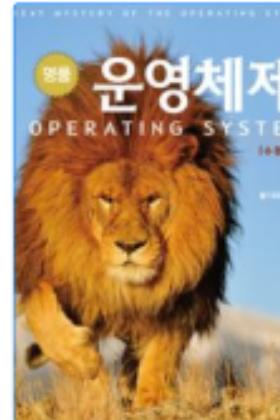
2023년 11월 22일



컴퓨터 활용능력 2급 필기  
시나공

뤼루

2023년 11월 22일



운영체제

뤼루

2023년 11월 22일

## • 나의 카드북

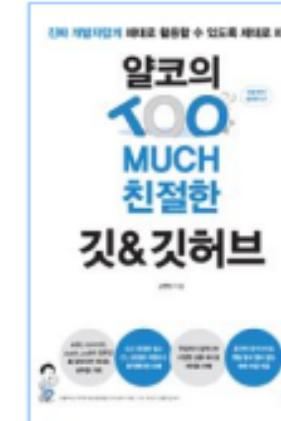


자료구조

윤성우

2023년 11월 22일

수정 | 삭제



운영체제

윤성우

2023년 11월 22일

수정 | 삭제



카드북 이름

example

신규 카드북 등록



망각 곡선이란? 사용법 내 정보



로그인

회원가입

## 회원가입

암기에 목마를 때, 뤄뫄

아이디

example

비밀번호

\*\*\*\*\*



비밀번호 확인

\*\*\*\*\*



닉네임

example

가입하기



망각 곡선이란?

뤼뤼 사용법

내 정보



로그인

회원가입

## · 기본 카드북

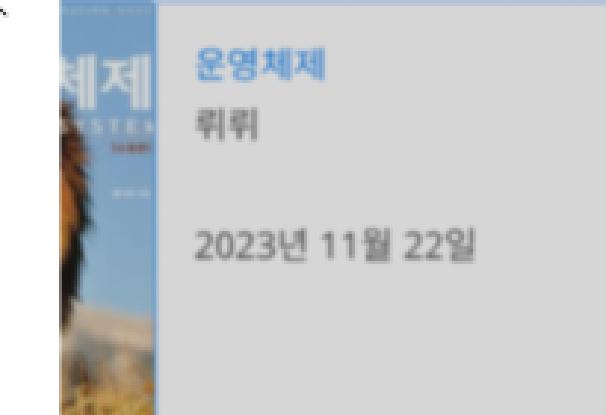


해커스 토익 기출  
뤼뤼  
2023년 11월 22일



아이디

example



운영체제  
뤼뤼  
2023년 11월 22일

## · 나의 카드북



자료구조  
윤성우  
2023년 11월 22일

수정 | 삭제

로그인

깃&깃허브



2023년 11월 22일

수정 | 삭제

이미지 등록

신규 카드북 등록



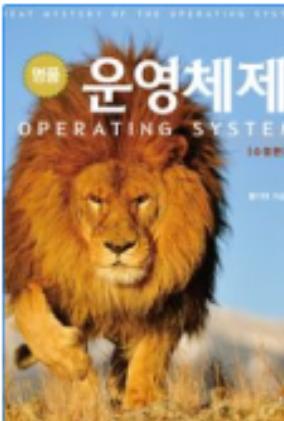
망각 곡선이란? 뤄루 사용법 내 정보



로그인

회원가입

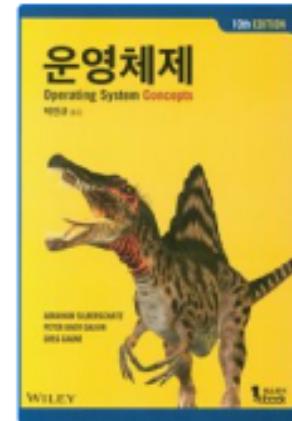
## < 운영체제 검색 결과



명품 운영체제

뤼루

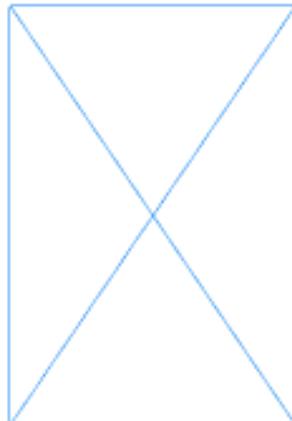
2023년 11월 22일



운영체제

사과

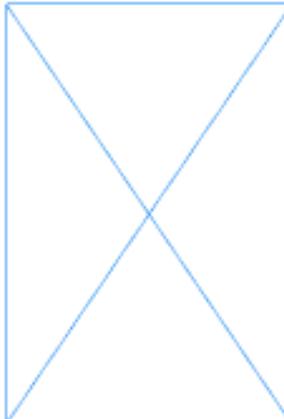
2023년 11월 22일



운영체제 정리

모모

2023년 11월 22일



운영체제

티모

2023년 11월 22일



운영체제 내부구조

다리우스

2023년 11월 22일



망각 곡선이란? 뤼큐 사용법 내 정보



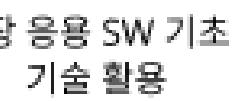
로그인

회원가입



## 정보 처리 산업 기사 실기

[수정](#) | [삭제](#)



1장 응용 SW 기초  
기술 활용



1시간



하루



3일



일주일

[수정](#) | [삭제](#)



2장 UI 테스트



1시간



하루



3일



일주일

[수정](#) | [삭제](#)



3장 화면 구현



1시간



하루



3일



일주일



신규

[망각 곡선이란?](#)[뒤뒤 사용법](#)[내 정보](#)[로그인](#)[회원가입](#)

## 1장 응용 SW 기초 기술 활용

컴퓨터와 시스템의 자원들을  
효율적으로 관리하며, 사용자가  
컴퓨터를 편리하고 효과적으로  
사용할 수 있도록 환경을 제공  
하는 여러 프로그램의 모임으로,  
CPU, 메모리 공간, 기억 장치,  
입출력장치, 등의 자원을 관리  
하는 것은?



망각 곡선이란?

뤼뤼 사용법

내 정보



로그인

회원가입



## 1장 응용 SW 기초 기술 활용

운영체제

1/6

[망각 곡선이란?](#)[뤼루 사용법](#)[내 정보](#)[로그인](#)[회원가입](#)

## 목차명 입력

문제 입력

정답 입력

카드 추가

2개 카드 선택됨 / 6개 카드



[수정](#) | [삭제](#)

컴퓨터와 시스템의 자원들을 효율적으로 관리하며, 사용자가 컴퓨터를 편리하고 효과적으로 사용할 수 있도록 환경을 제공 하는 여러 프로그램의 모임으로, CPU, 메모리 공간, 기억 장치, 입출력장치, 등의 자원을 관리하는 것은?

운영체제



[수정](#) | [삭제](#)

<input checked="" type="checkbox"/>		수정   삭제

<input checked="" type="checkbox"/>		수정   삭제

<input checked="" type="checkbox"/>		수정   삭제

<input checked="" type="checkbox"/>		수정   삭제

# 진행 사항

1. 가능 명세서
2. 와이어 프레임
3. ERD, API
4. 깃 허브 플로우 채택
5. 커밋 컨벤션

## Commits

main

- Commits on Sep 17, 2023
  - CSS 추가**  
mlnwns committed on Sep 17
  - README.md 파일 작성**  
mlnwns committed on Sep 17
  - 배포**  
mlnwns committed on Sep 17
  - 경로 수정**  
mlnwns committed on Sep 17
- Commits on Sep 16, 2023
  - 라우터**  
mlnwns committed on Sep 16

```
git add .
git commit -m "message"
```

## routes edit

 **epitoneproject** committed on 28 Aug

## todoDetail pug edit

 **epitoneproject** committed on 28 Aug

## todoDetail controller setting

 **epitoneproject** committed on 28 Aug

## todos id routes setting

 **epitoneproject** committed on 28 Aug

## db 수정

 **epitoneproject** committed on 28 Aug

**feat: Routing** 진행 중 

Route: 기본 라우트 설정

 **epitoneproject** committed 20 hours ago

**feat: husky & lint-staged** 설정진행 중 

husky: git staging area에 올라온 코드만 자동으로 바꿔주도록 설정

lint-staged: husky가 staging area에 올라온 코드만 바꿔주도록 설정

 **epitoneproject** committed yesterday

가독성 ↑

# Udacity Git Commit Message Style Guide?

**Message Type: Subject** // 최대 50글자가 넘지 않도록 하고 마침표는 찍지 않습니다. 영문으로 표기하는 경우 동사(원형)를 가장 앞에 두고 첫 글자는 대문자로 표기합니다.

(한 줄 띠어 분리)

**Body** // 긴 설명이 필요한 경우에 작성합니다. 어떻게 했는지가 아니라 무엇을 왜 했는지를 작성합니다. 한 줄 당 72자를 넘기지 않도록 합니다.

(한 줄 띠어 분리)

**Footer** // issue tracker ID를 명시하고 싶은 경우에 작성합니다.

# Message Type

feat : 새로운 기능을 추가하였을 때

fix : 버그를 수정하였을 때

docs : README 등 문서 내용을 변경하였을 때

style : 들여쓰기, 세미콜론 등을 변경하였을 때 (코드 수정은 없는 경우)

refactor : 코드 리팩토링을 했을 때(기능의 변경은 없어야 한다.)

design: CSS 등 사용자 UI 디자인 변경

rename: 파일 혹은 폴더명을 수정하는 경우

remove: 사용하지 않는 파일 혹은 폴더를 삭제하는 경우

test : test코드의 작성 및 수정이 이루어졌을 때

chore : 외부 라이브러리 임포트 등의 작업을 완료했을 때

# Subject

1. 제목의 처음은 동사 원형으로 시작합니다.
2. 총 글자 수는 50자 이내로 작성합니다.
3. 마지막에 특수문자는 삽입하지 않습니다. 마침표(.), 느낌표(!), 물음표(?)
4. 제목은 개조식 구문으로 작성합니다.

만약 영어로 작성하는 경우 다음의 규칙을 따릅니다.

- 첫 글자는 대문자로 작성합니다.
- "Fix", "Add", "Change"의 명령어로 시작합니다.

# Footer

1. 꼬리말은 optional이고 이슈 트래커 ID를 작성합니다.
  2. 꼬리말은 "유형: #이슈 번호" 형식으로 사용합니다.
  3. 여러 개의 이슈 번호를 적을 때는 쉼표로 구분합니다.
- 
4. 이슈 트래커 유형은 다음 중 하나를 사용합니다.
    - Fixes: 이슈 수정중 (아직 해결되지 않은 경우)
    - Resolves: 이슈를 해결했을 때 사용
    - Ref: 참고할 이슈가 있을 때 사용
    - Related to: 해당 커밋에 관련된 이슈번호 (아직 해결되지 않은 경우)
- ex) Fixes: #45

# EX)

feat: Summarize changes in around 50 characters or less

More detailed explanatory text, if necessary. Wrap it to about 72 characters or so.

If you use an issue tracker, put references to them at the bottom, like this:

Resolves: #123

# Q & A

Rememberance  
Repository