

리눅스(Linux) - 우분투 데스크톱 20.04, 센토스(CentOS) -> ImageMagick 기반의 g++ 프로그램 (사진압축 도구)

2021. 04. 11

도도(Dodo)

GNU/GPL v3 License

목차

-
1. 문서 이력
 2. 라이선스
 3. [안내 사항] 지구가 많이 아파요 (2021-04-11)
 4. [작업] 소개 (2021-04-11)
 5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)
 6. [프로그램] g++ (pthread) 컴파일 (2021-04-11)
 7. [프로그램] 우분투 20.04, g++ 설치(2021-04-11)
 8. [작업] 소개 - 번역 / 오픈소스 - 참여 활동[(lxqt)Tsujan/FeatherNote - 패더노트] (2021-04-11)
 9. [작업] 번역 / 오픈소스 - 참여 활동[(lxqt)Tsujan/FeatherNote - 패더노트] (2021-04-11)
 10. [리눅스 배포판] 프로그램 소개 / [(lxqt) FeatherNote - 패더노트] (2021-04-11)
 11. [부록1] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(쉘 프로그래밍) (2021-04-11)
 12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)
-

1. 문서 이력

2021-04-11 19:50 문서 최초 작성, 도도(Dodo)

2021-04-11 20:30 문서 작성 완료, 도도(Dodo)

2. 라이선스

Copyright 2021. Dodo [리눅스(Linux) / 이미지 / 사진 / 그림 - 압축 (자동화 도구) G++(c++) (2021-04-11)]
→ (Linux / image / photo / picture-compression (automation tool) G++(c++) (2021-04-11))

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program.
If not, see <<https://www.gnu.org/licenses/>>.

3. [안내사항] 지구가 많이 아파요 (2021-04-11)

꼭 참고하길 바란다.

지구가 많이 아파요. “The earth hurts a lot.”



3. [안내사항] 지구가 많이 아파요 (2021-04-11)

꼭 참고하길 바란다.

지구가 많이 아파요. “The earth hurts a lot.”



4. [작업] 소개 (2021-04-11)

우분투 20.04, 센토스(CentOS) 운영체제를 설치한 상태에서 진행하길 바란다.

- * 작업 1차 시작: 2021-04-11(일요일/Sunday) 11:30
- * 작업 1차 종료: 2021-04-11(일요일/Sunday) 12:20
- * 작업 2차 시작: 2021-04-11(일요일/Sunday) 13:20
- * 작업 2차 종료: 2021-04-11(일요일/Sunday) 16:20
- * 작업 3차 시작: 2021-04-11(일요일/Sunday) 17:10
- * 작업 3차 종료: 2021-04-11(일요일/Sunday) 18:10

노트(Note)

IT인력은 대한민국의 서울특별시, 경기도, 제주도, 대전광역시에 전부 거주하지 않음.
IT personnel do not all reside in Seoul, Gyeonggi-do, Jeju-do, and Daejeon in Korea.

개발 언어(Programming Language)

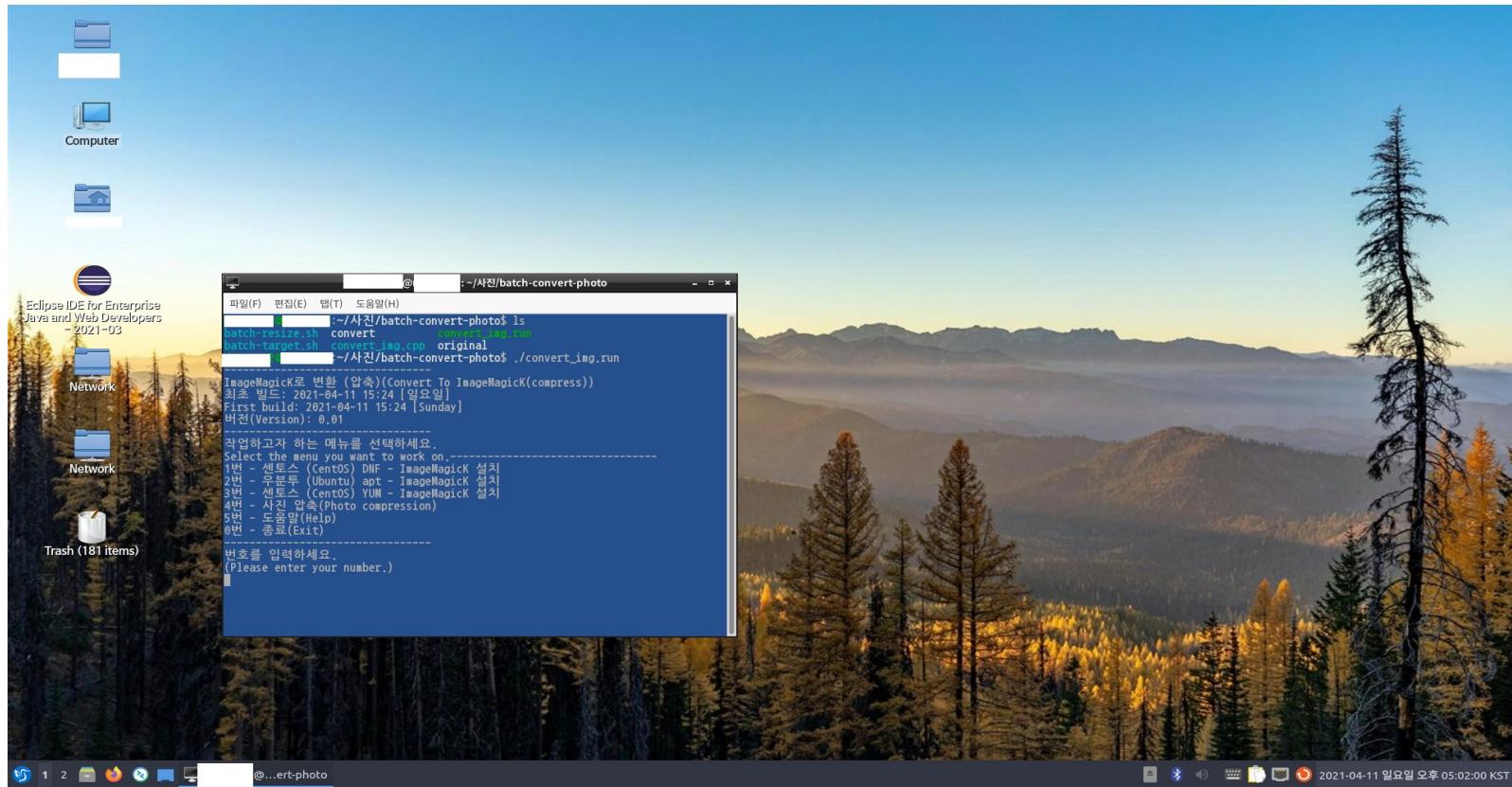
G++, GCC

운영체제(Operation System)

우분투 20.04,
센토스(CentOS)
센토스 스트림(CentOS-Stream)

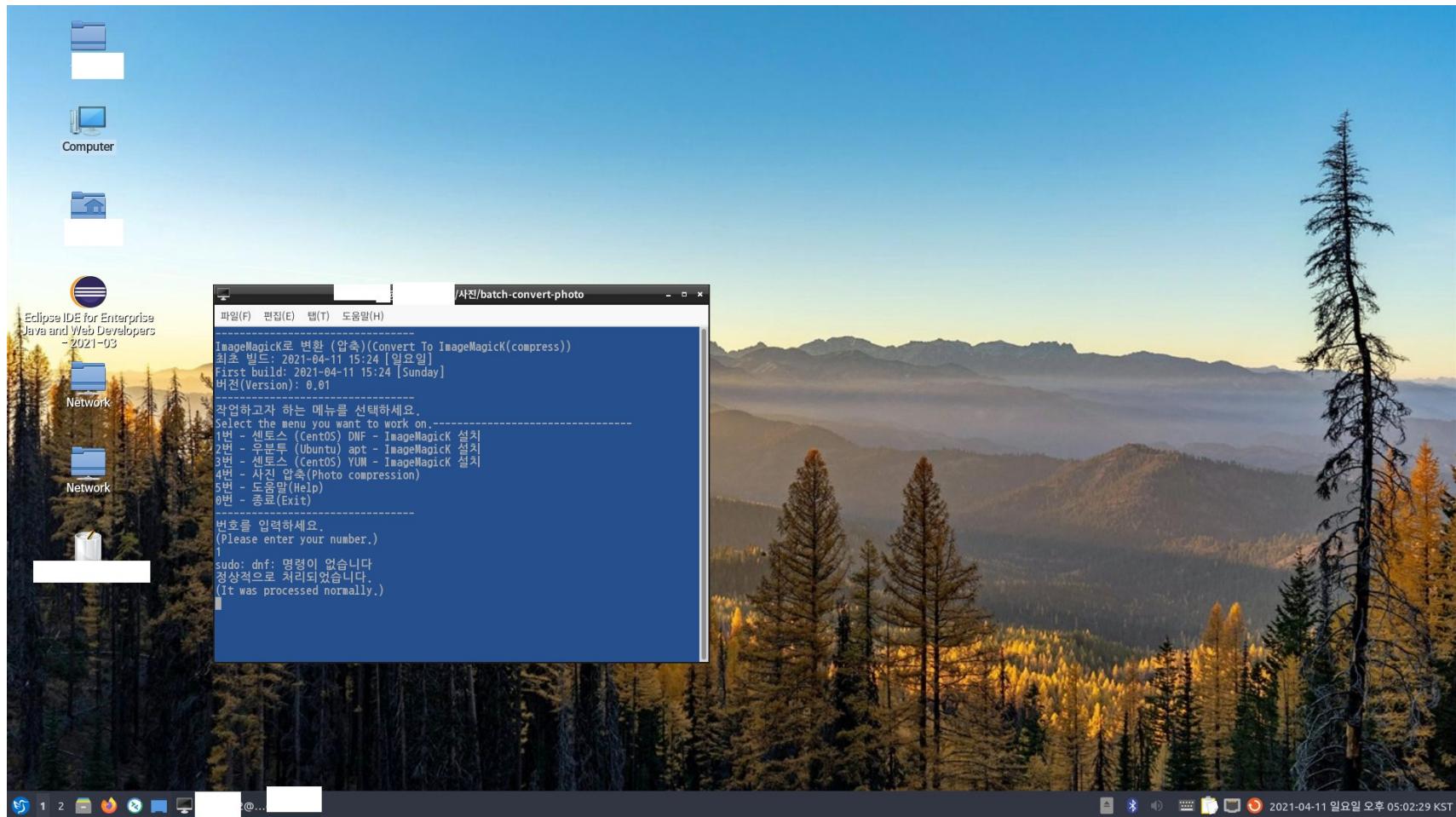
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



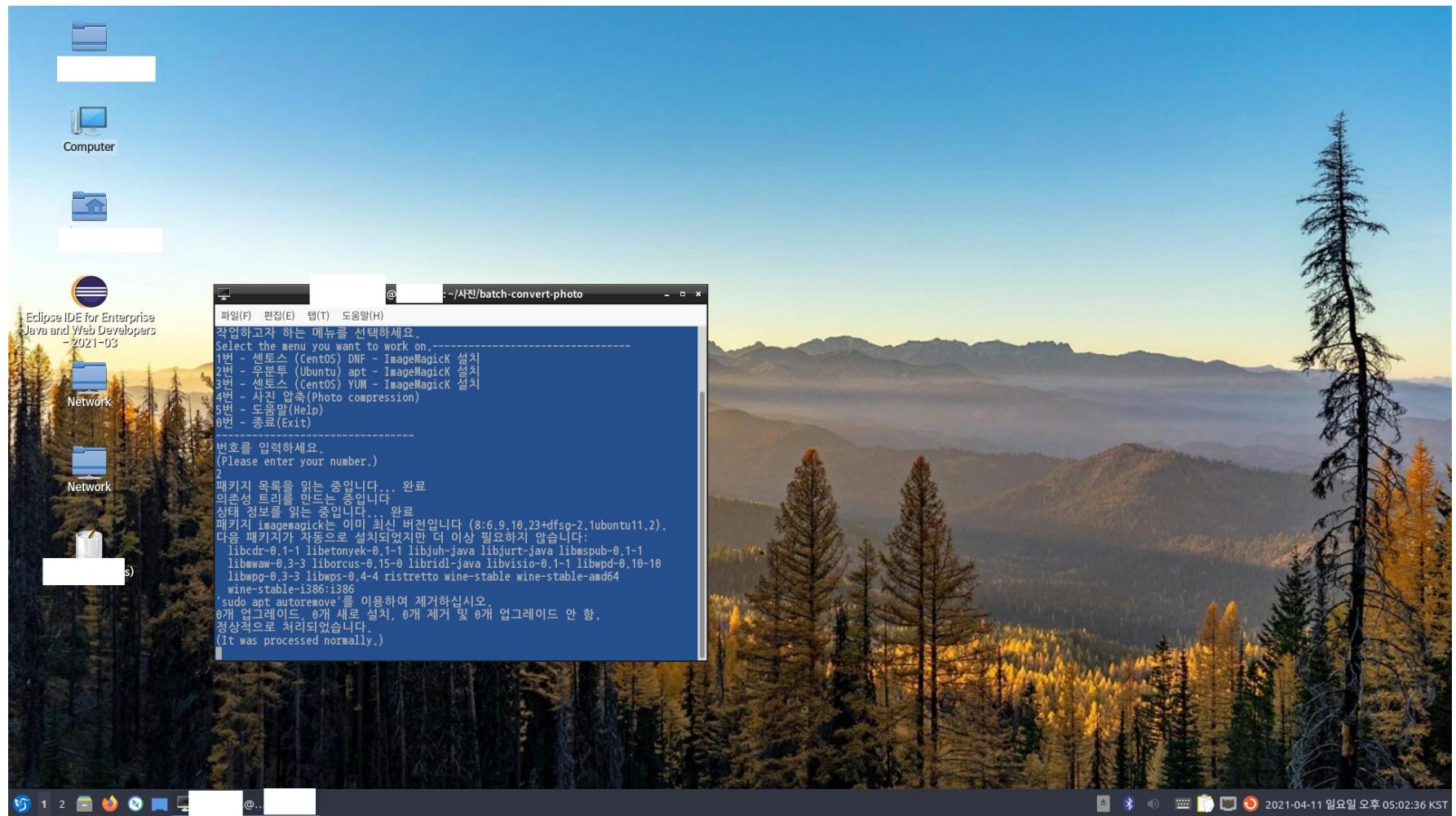
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



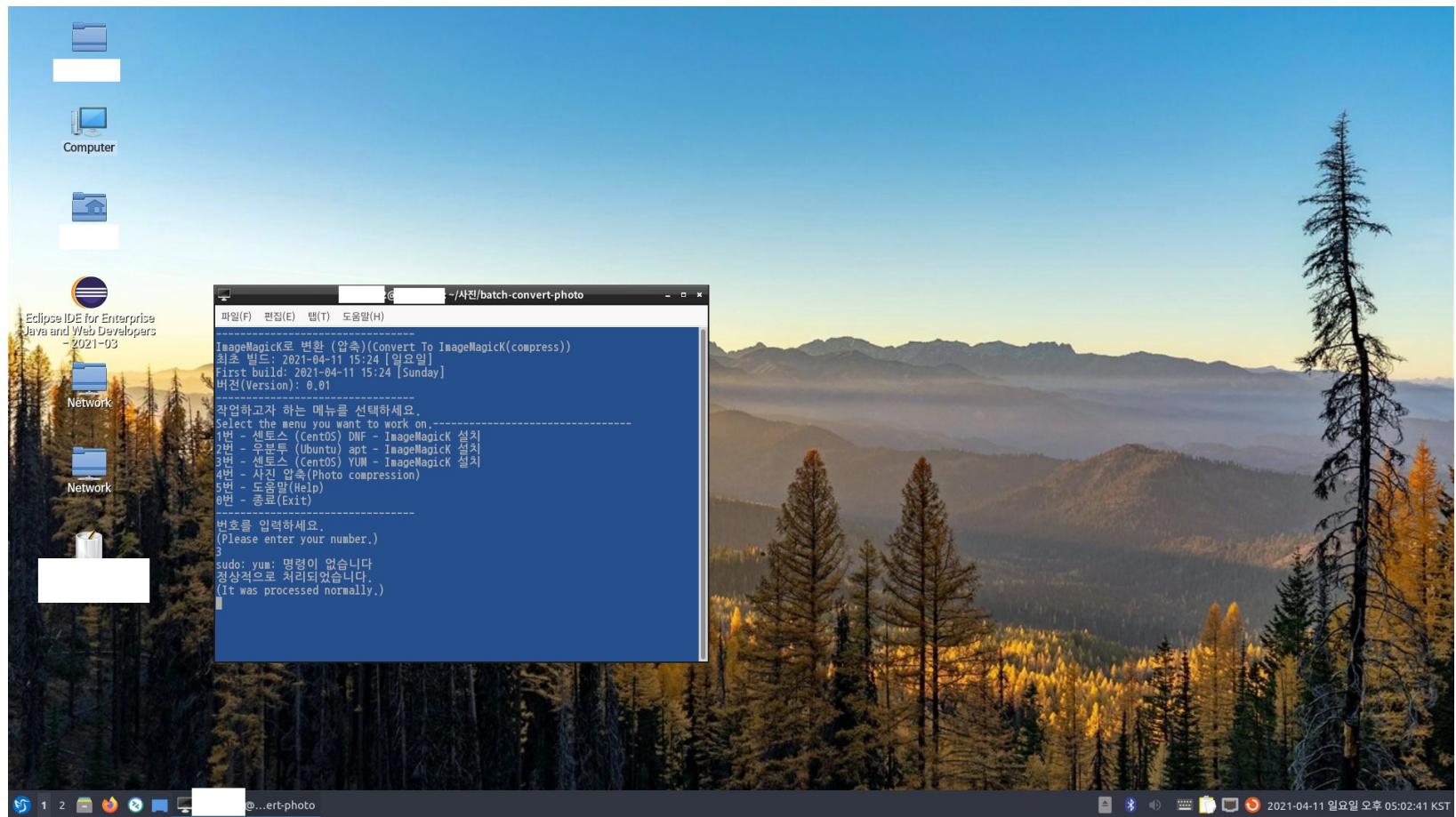
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



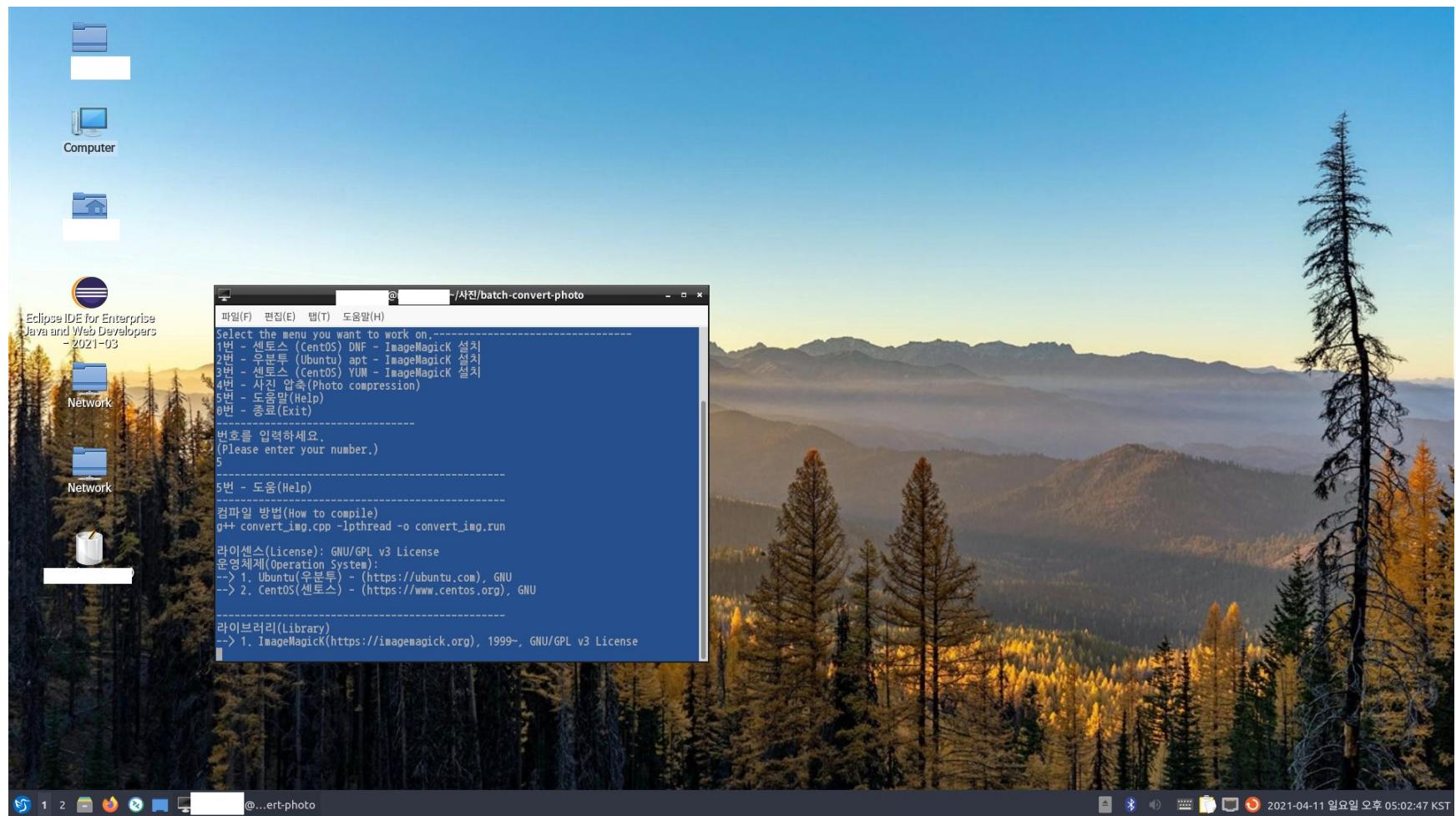
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



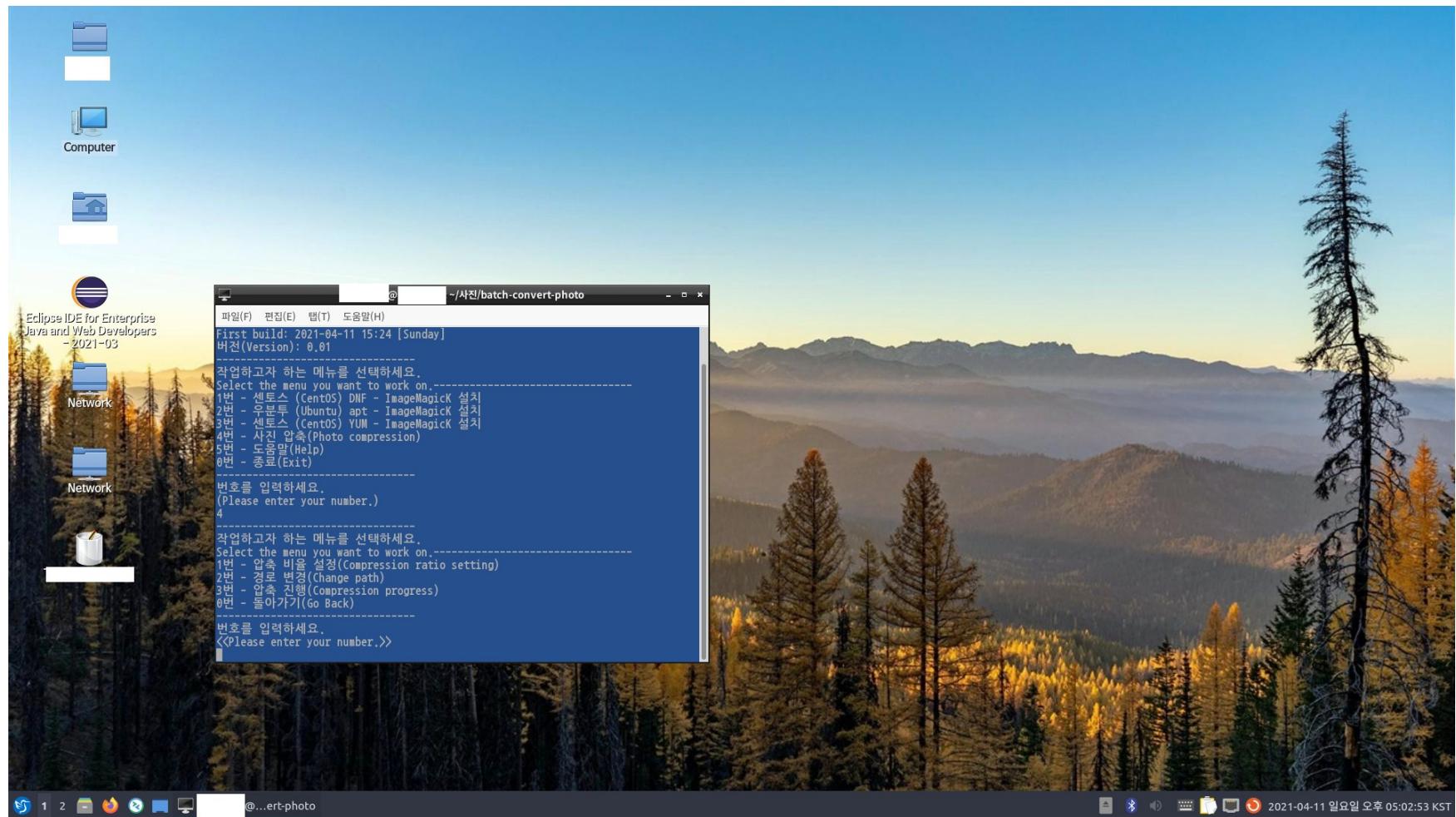
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



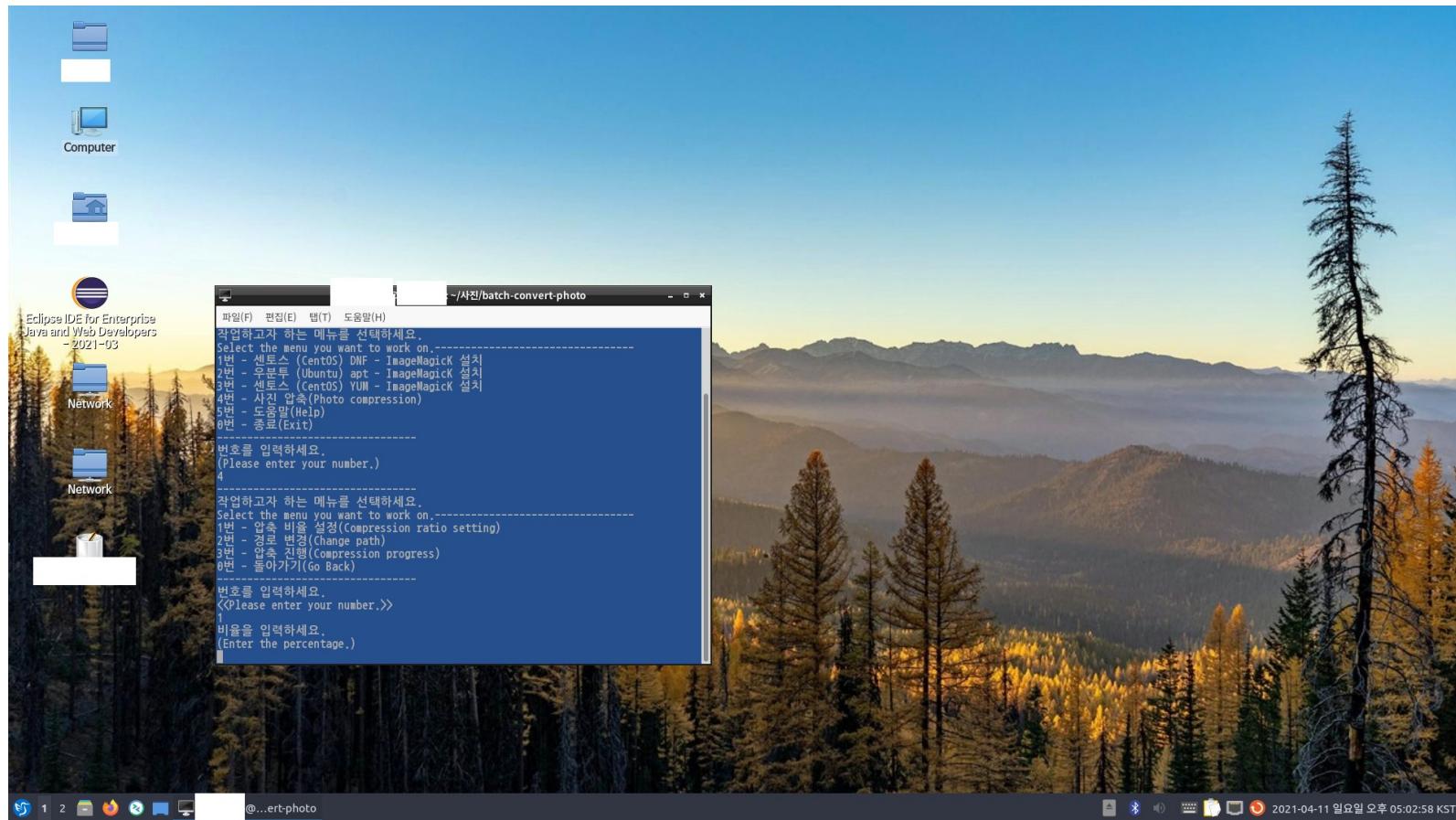
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



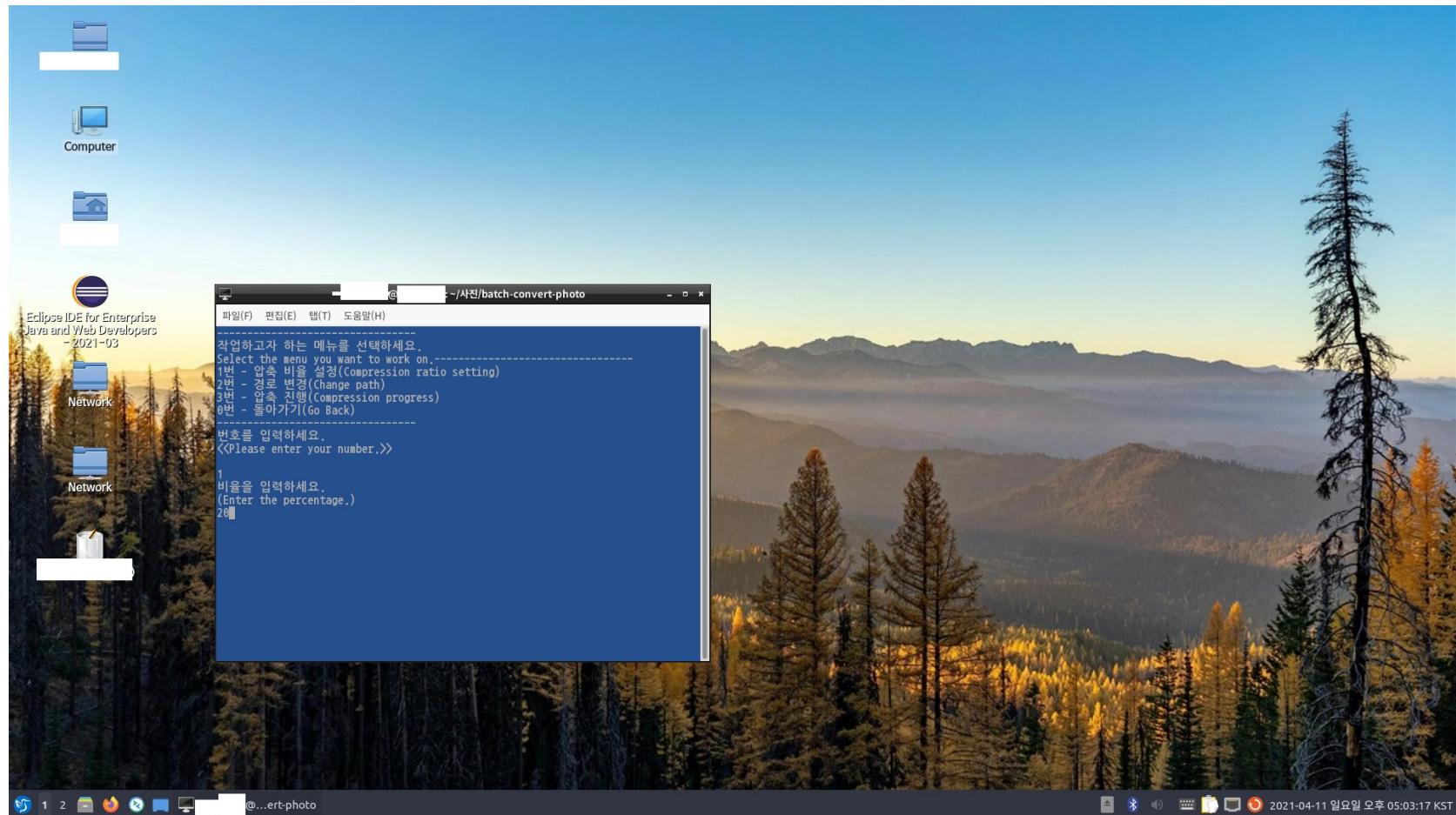
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



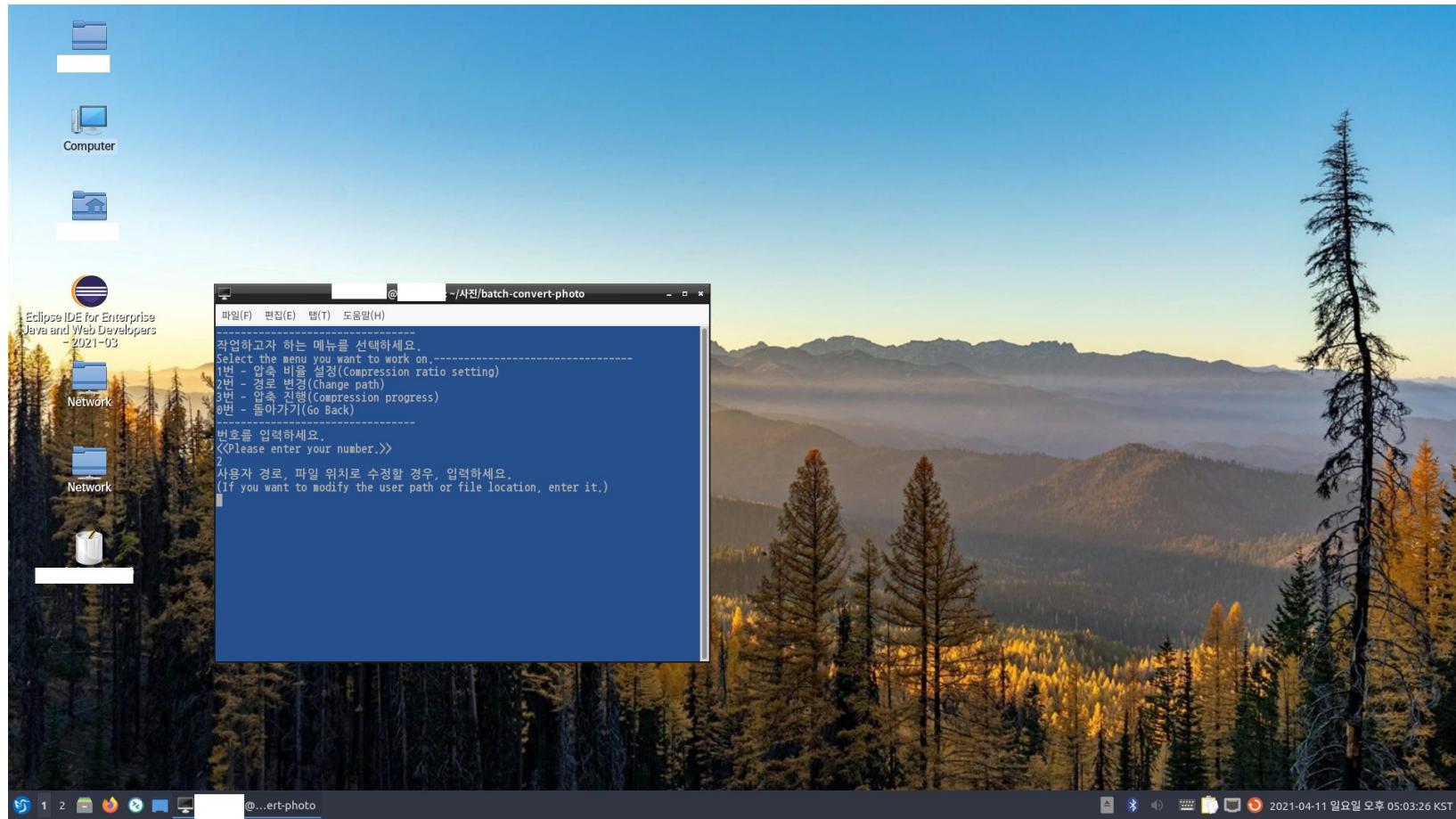
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



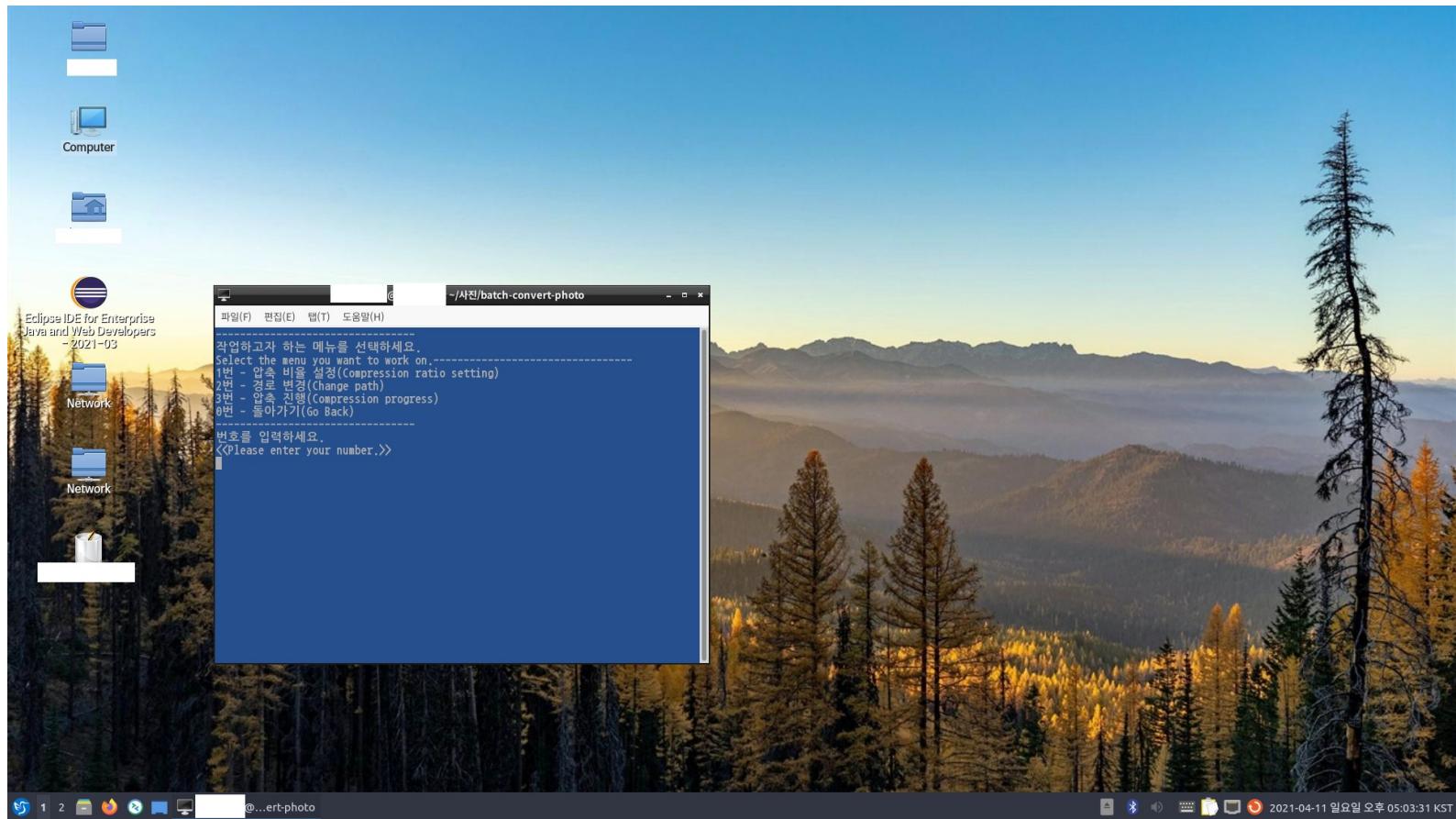
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



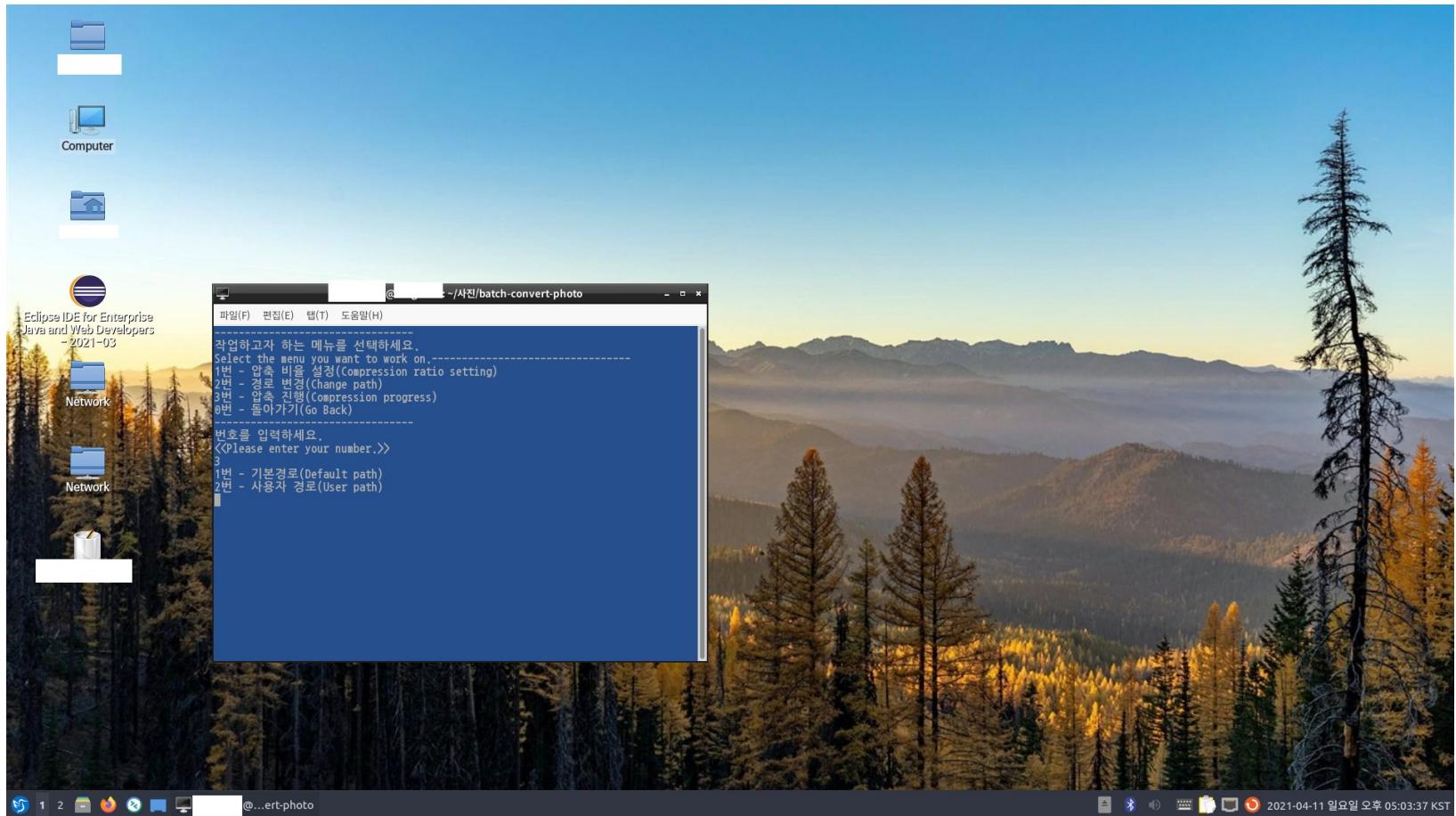
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



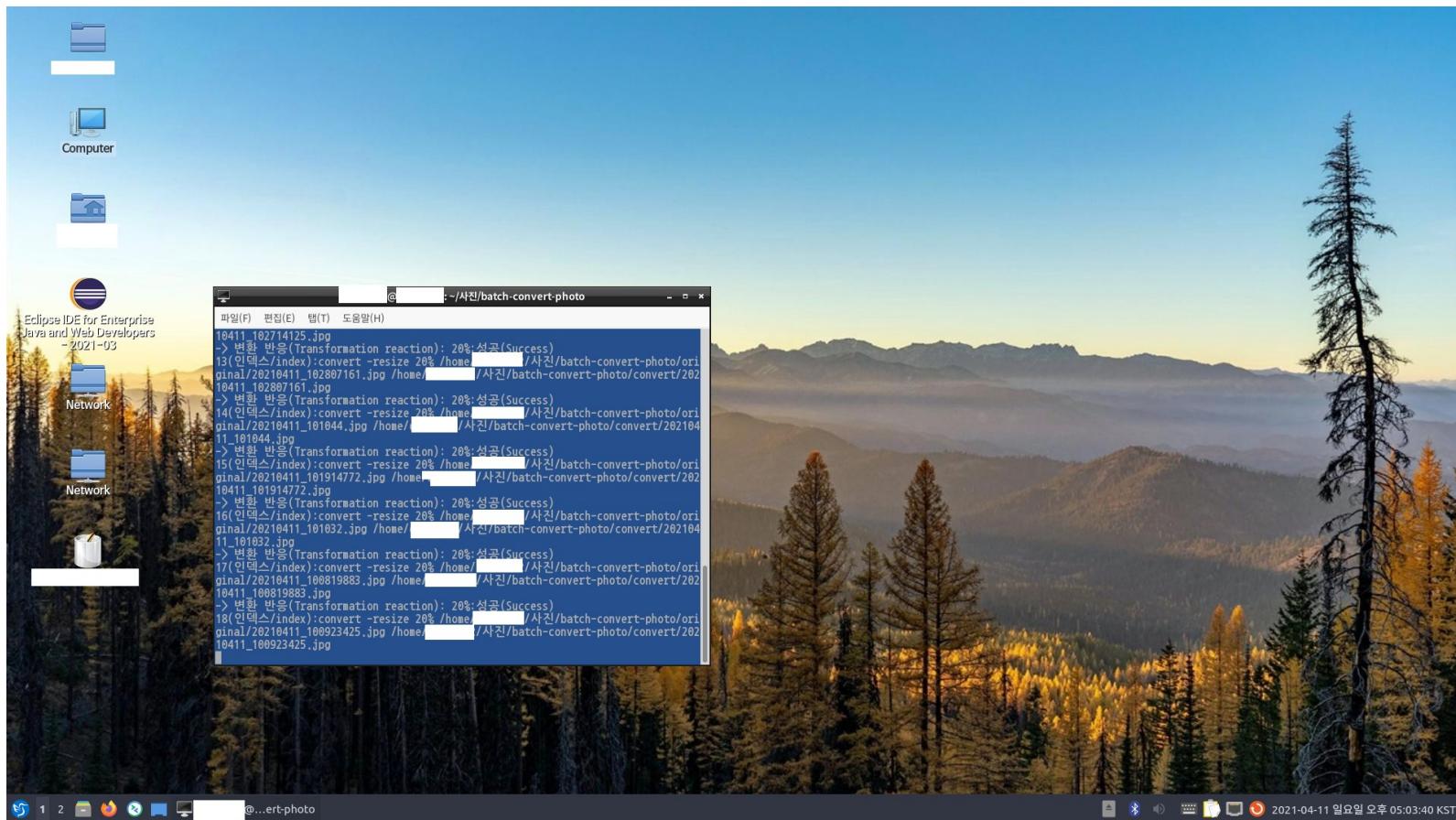
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



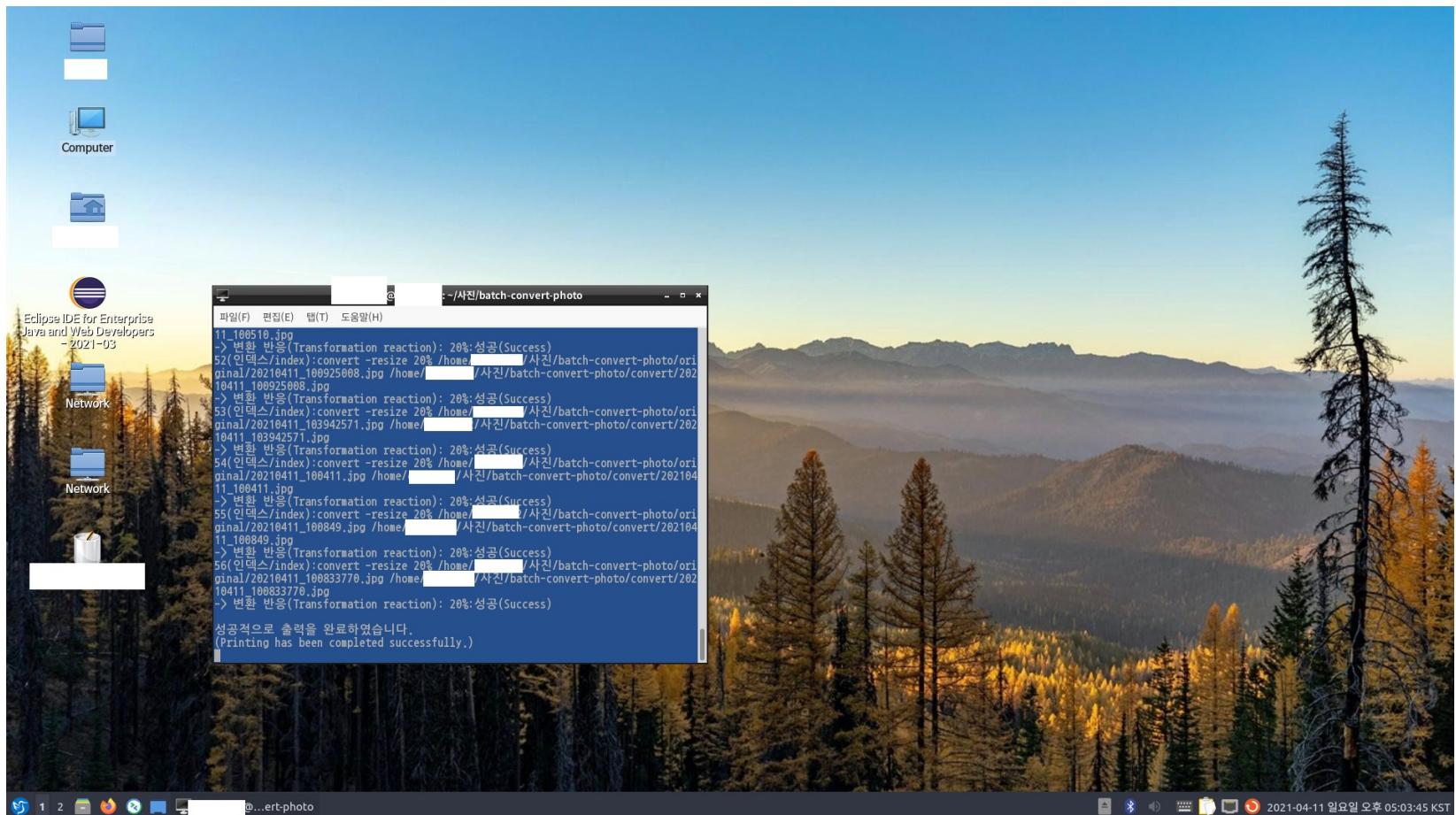
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



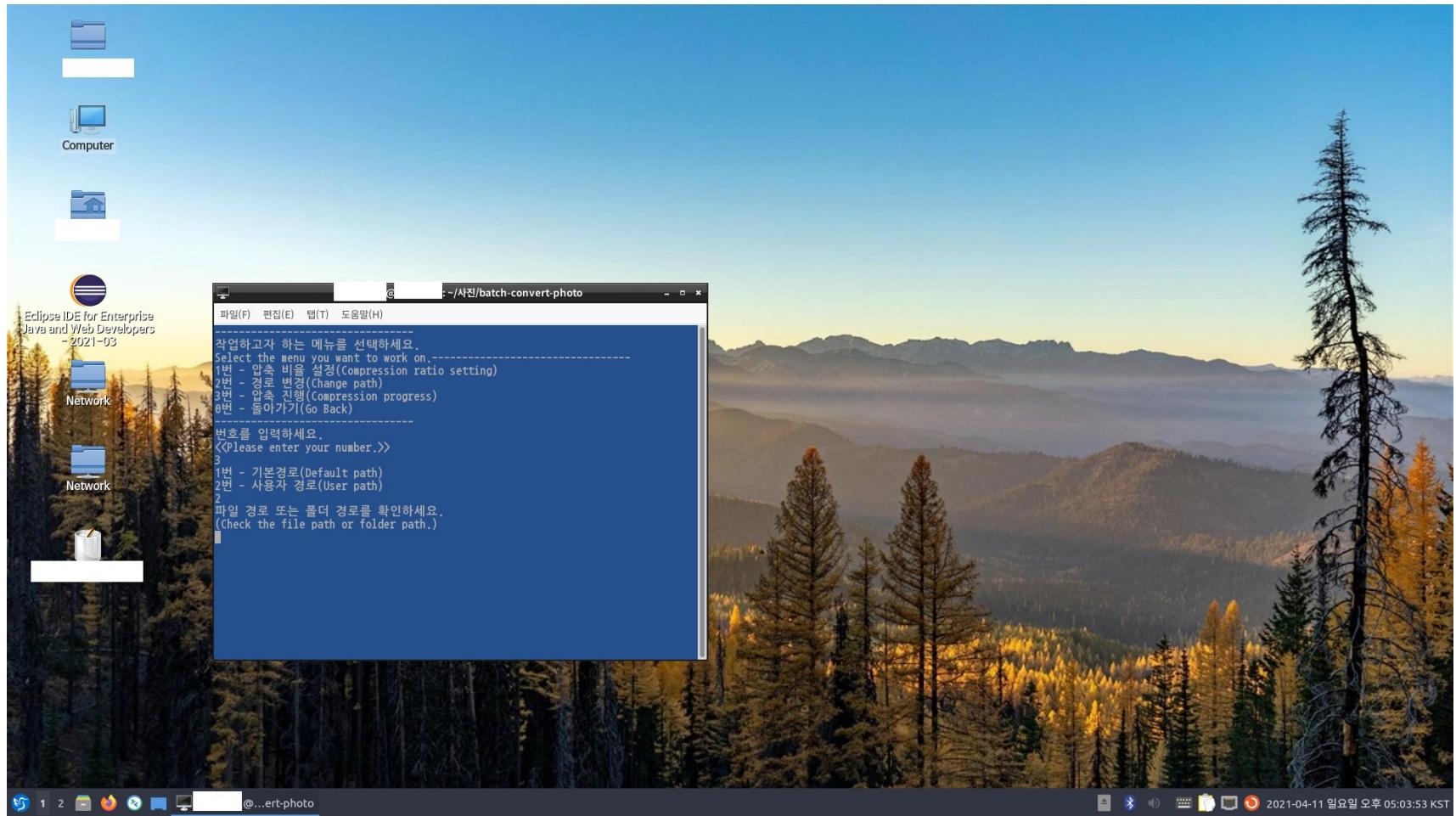
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



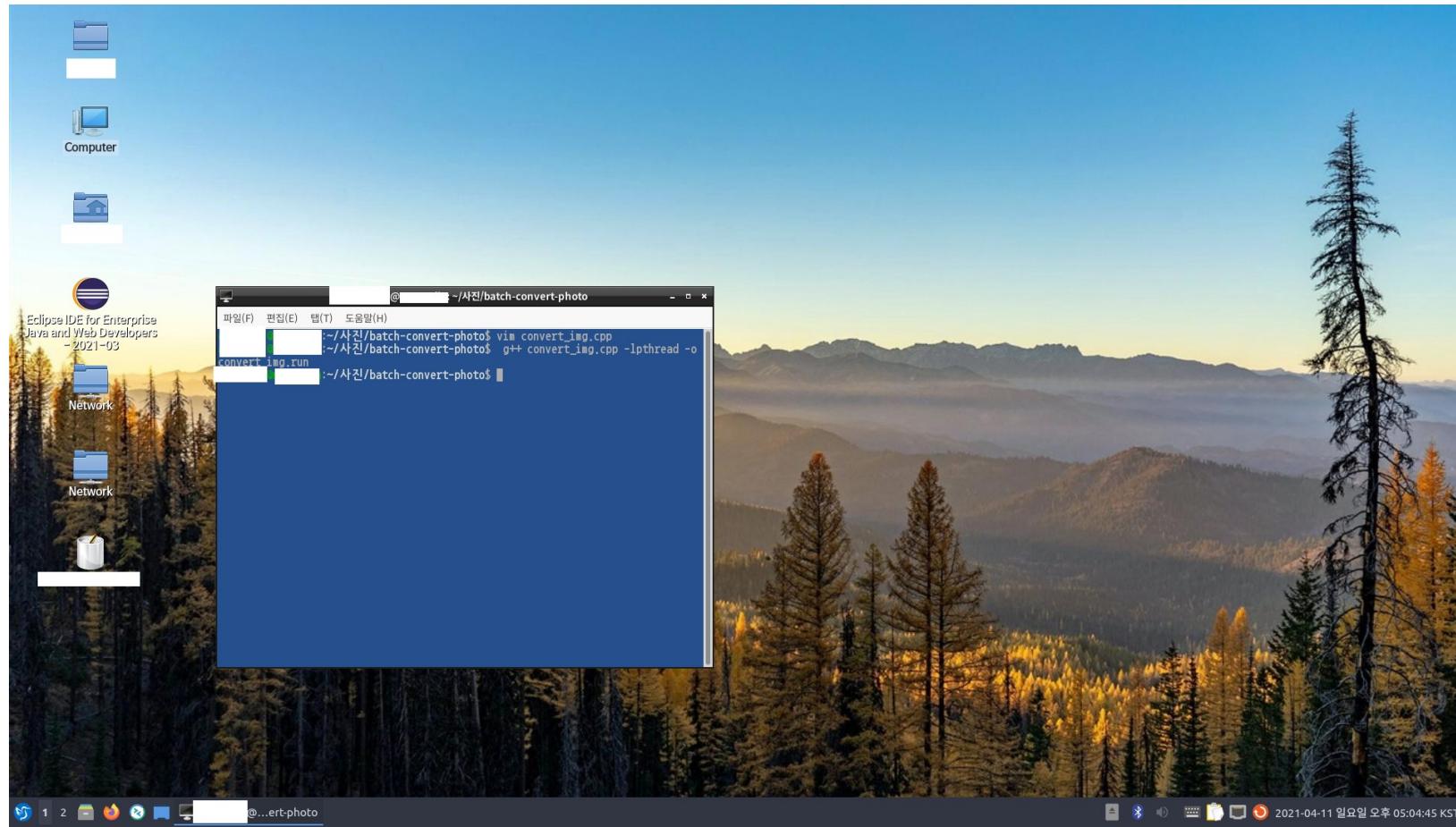
5. [프로그램] 실제 구현 모습 - 시연(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



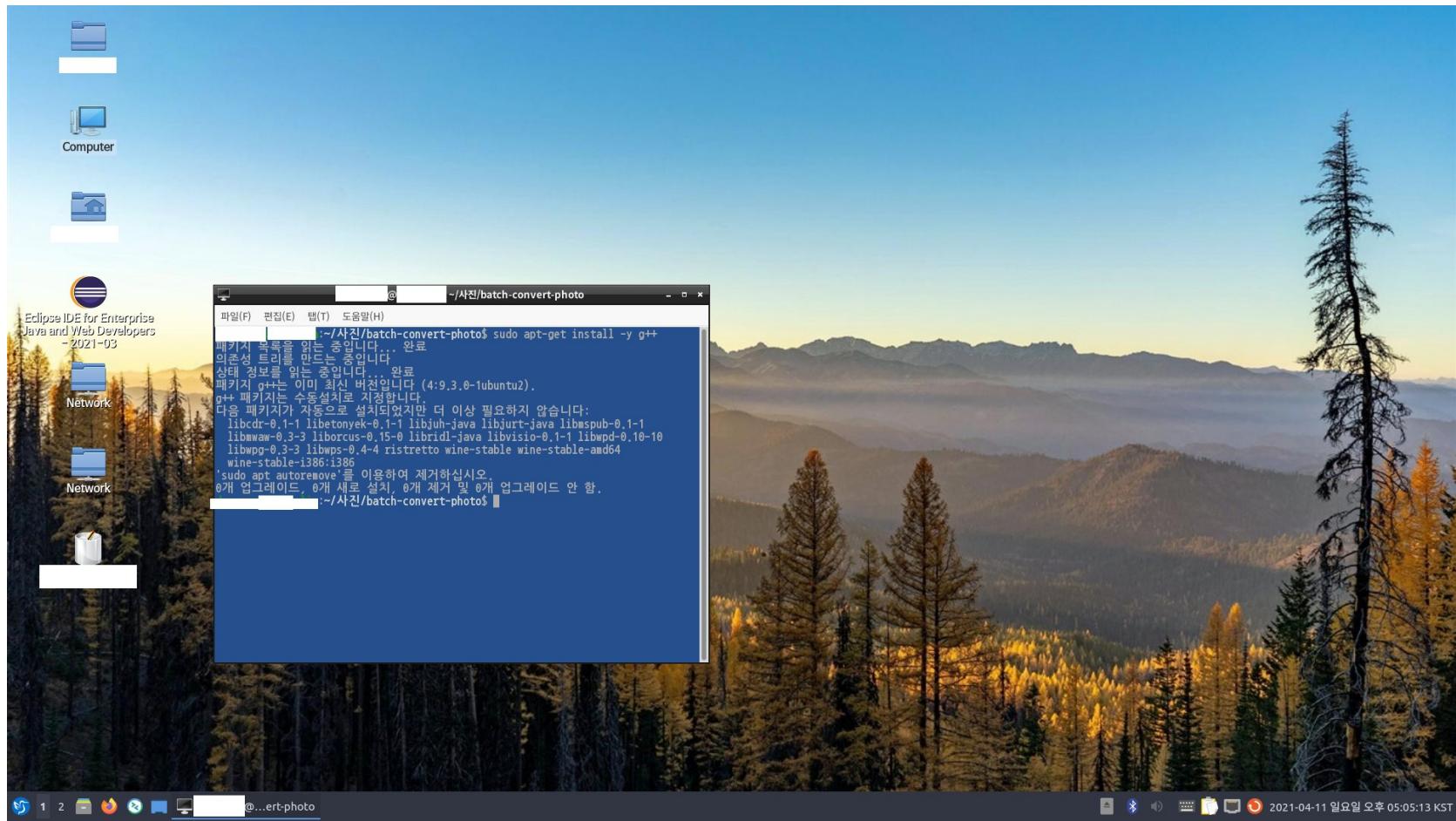
6. [프로그램] g++ (lpthread) 컴파일 (2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



7. [프로그램] 우분투 20.04, g++ 설치(2021-04-11)

우분투 20.04에서 작동 시킨 모습이다.



8. [작업] 소개-번역 / 오픈소스 - 참여 활동 [(lxqt)Tsujan/FeatherNote - 패더노트] (2021-04-11)

오픈소스 번역 활동을 참여하였다.

- * 작업 1차 시작: 2021-04-11(일요일/Sunday) 07:40
- * 작업 1차 종료: 2021-04-11(일요일/Sunday) 08:40

현재 배포판은 번역이 되어있지 않으니, 참고하면 될 것 같다.

8. [작업] 소개-번역 / 오픈소스 - 참여 활동 [(lxqt)Tsujan/FeatherNote - 패더노트] (2021-04-11)

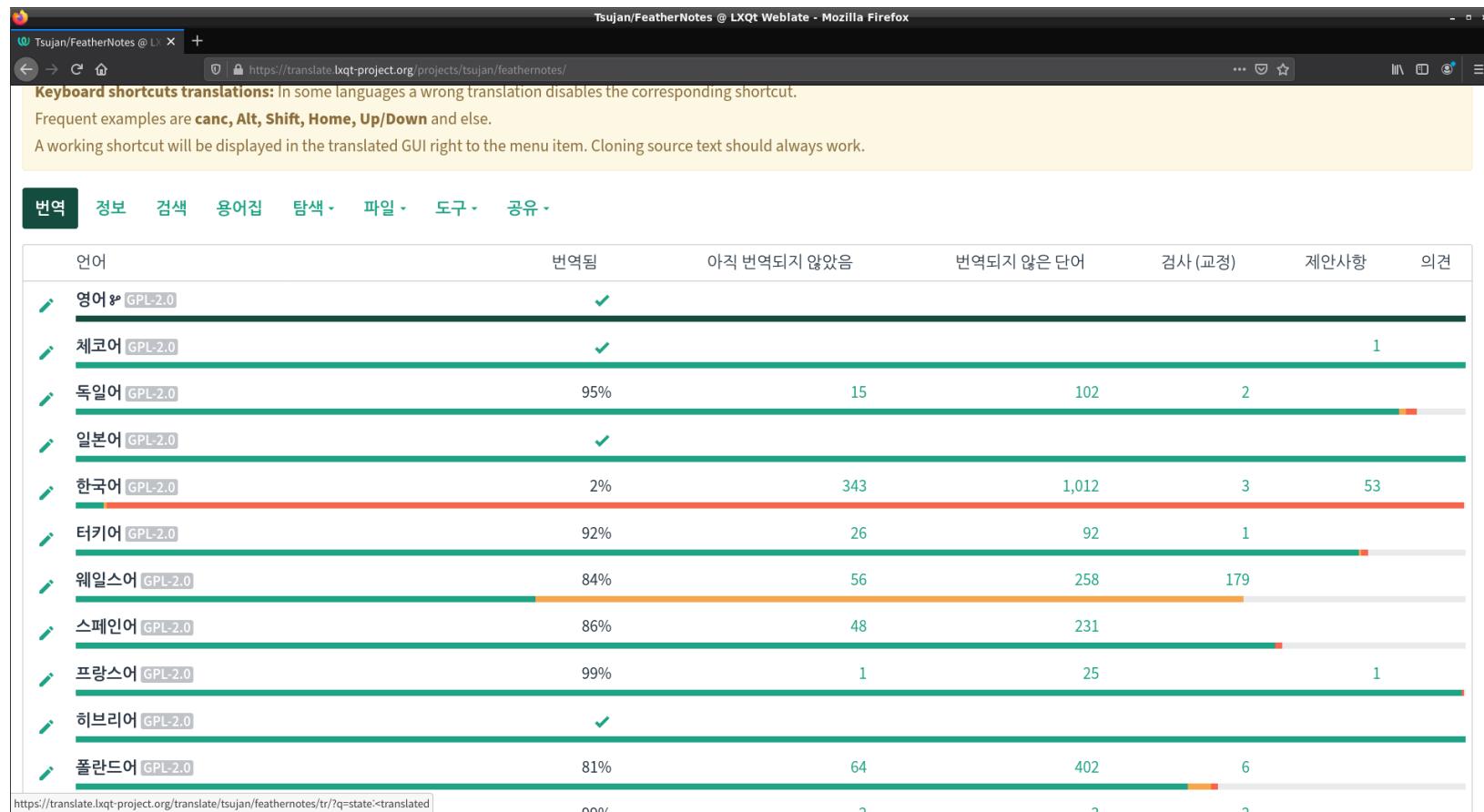
관심이 있으면 참여해도 무방하다. 아침에 잠깐 시간을 짬 내서 참여를 하였다.

The screenshot shows a web browser displaying the LXQt project website at <https://lxqt-project.org>. The page features a blue header with the LXQt logo and navigation links for About, Releases, Downloads, Screenshots, Translations, and Contribute. Below the header, there's a large image of the LXQt desktop environment with several windows open, including an application menu, a file manager, and a configuration tool. To the right of the desktop image, there are sections for Discussions, Bug Tracker, Wiki, Instant messaging, and a feed. The discussions section links to github.com/lxqt/lxqt/discussions. The bug tracker section links to github.com/lxqt/lxqt/issues. The wiki section links to github.com/lxqt/lxqt. The instant messaging section links to Matrix: #lxqt:matrix.org, Telegram: LXQt Project, and IRC: #lxqt on OFTC (bridged among them). The feed section links to lxqt-project.org/feed.xml. The footer contains social media links for GitHub, ALT Linux, Arch Linux, Chakra, Debian, Fedora, Gentoo, Mageia, Manjaro, OpenMandriva, ROSA Linux, openSUSE, and Ubuntu.

9. [작업] 번역 / 오픈소스 - 참여 활동 [(lxqt)Tsujan/FeatherNote - 패더노트] (2021-04-11)

한국어 번역이 안 되어 있어서 입력하였다. (343건 입력함)

참고사항: “<html>태그”를 번역해버리면 안 됨.



9. [작업] 번역 / 오픈소스 - 참여 활동 [(lxqt)Tsujan/FeatherNote - 패더노트] (2021-04-11)

시스템에서는 feathernotes_ko.ts 파일이 “자동생성” 되었을 것으로 보인다.
조만간 기다려봐야 할 것으로 보인다.

The screenshot shows a Mozilla Firefox window with the title "Tsujan/FeatherNotes – 한국어 @ LXQt Weblate: License - Mozilla Firefox". The address bar contains the URL "https://translate.lxqt-project.org/translate/tsujan/feathernotes/ko/?q=". The main content area displays a translation interface for the "FeatherNotes::AboutDialog" file. On the left, there is a table with columns for Key, English, Korean, and Status. The table lists various keys and their translations, such as "FeatherNotes::AboutDialog" (License, 라이센스), "FeatherNotes::ColorLabel" (Click to change color., 색상을 변경하려면 클릭하십시오.), and "FeatherNotes::FN" (Next (F3), 다음 (F3)). Most entries have a green checkmark in the status column. On the right side of the interface, there are sections for "Labels" (No labels set), "Original character location" (about.ui:147), and "Last modified time" (3 months ago). The bottom of the screen shows the Linux desktop environment with icons for the taskbar and system tray.

키	영어	한국어	상태
FeatherNotes::AboutDialog	License	라이센스	▶
FeatherNotes::ColorLabel	Click to change color.	색상을 변경하려면 클릭하십시오.	▶ 🔍
FeatherNotes::ColorLabel	Select Color	색상 선택	▶ 🔍
FeatherNotes::FN	FeatherNotes	페더 노트	▶ 🔍
FeatherNotes::FN	Next (F3)	다음 (F3)	✓
FeatherNotes::FN	F3	F3	▶ 🔍
FeatherNotes::FN	Previous (F4)	이전 (F4)	▶ 🔍
FeatherNotes::FN	F4	F4	▶ 🔍
FeatherNotes::FN	Search...	검색...	▶ 🔍
FeatherNotes::FN	Search only in names (Ctrl+Shift+F7)	이름만 검색 (Ctrl + Shift + F7)	▶ 🔍
FeatherNotes::FN	Ctrl+Shift+F7	Ctrl+Shift+F7	✓
FeatherNotes::FN	Search only in tags (Shift+F7)	태그에서만 검색 (Shift + F7)	▶ 🔍
FeatherNotes::FN	Shift+F7	Shift+F7	▶ 🔍

Labels

현재 설정된 레이블이 없습니다.

원본 문자열 위치

./about.ui:147

원본 문자열이 변경되었던 시간

3 달 전

번역 파일

feathernotes/data/translations/feathernotes_ko.ts,
문자열 1

https://translate.lxqt-project.org/translate/tsujan/feathernotes/ko/?checksum=672f7328b8971964

Firefox - 3 windows

eclipse-wor...clipse IDE

2021-04-11 일요일 오전 09:06:46 KST

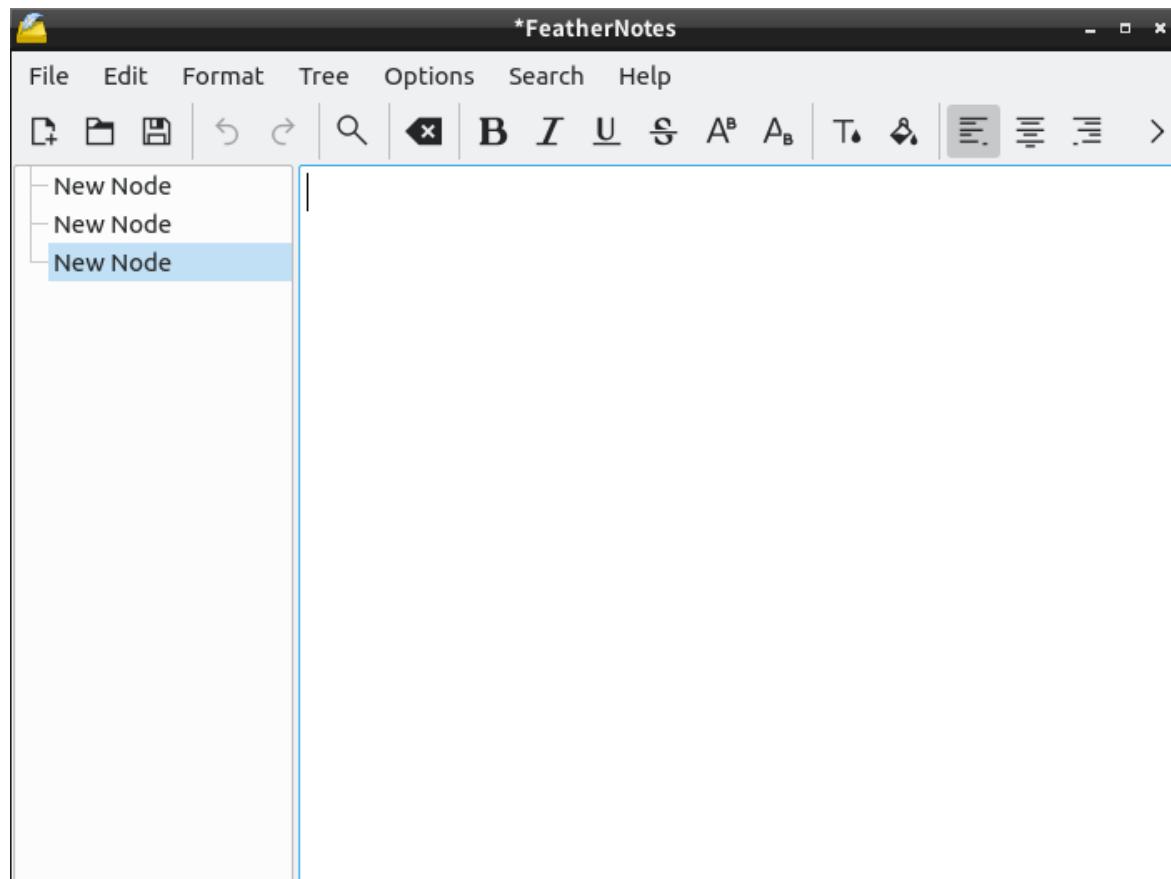
9. [작업] 번역 / 오픈소스 - 참여 활동 [(lxqt)Tsujan/FeatherNote - 패더노트] (2021-04-11)

조만간 한글 버전으로 하나 반영되어서 나올 것으로 보인다.

The screenshot shows a Firefox browser window with the title "Tsujan/FeatherNotes – 한국어 @ LXQt Weblate: Help - Mozilla Firefox". The URL in the address bar is <https://translate.lxqt-project.org/translate/tujuan/feathernotes/ko/?&offset=350>. The page content is a translation interface for the FeatherNotes application's help dialog. On the left, there is a search bar with the placeholder "모든 문자열" and a dropdown menu "우선 순위와 위치". Below it is a section titled "번역" with tabs for "영어" and "한국어". The "한국어" tab is active, showing a text input field containing "도움" and a checked checkbox "수정이 필요함 ①". There are three buttons at the bottom: "저장", "제안", and "넘기기". At the bottom left, there is a "주변 문자열 17" button, followed by tabs for "다른 경우 3", "다른 언어", and "역사". A table lists various key-value pairs, such as "FeatherNotes::PrefDialog" mapped to "Shortcuts", "Action", "Shortcut", "Default", and error messages like "Warning: Ambiguous shortcut detected!" and "The typed shortcut was reserved.". On the right side, there are two panels: "용어집" which says "용어집에서 관련된 문자열을 찾을 수 없습니다." and "원본 정보" which shows the key "FeatherNotes::helpDialog" and its original label "FeatherNotes::helpDialog". The status bar at the bottom shows the date and time: 2021-04-11 일요일 오전 09:07:13 KST.

10. [리눅스 배포판] 프로그램 소개 / [(lxqt) FeatherNote - 패더노트] (2021-04-11)

실제 프로그램이다.



11.[부록 1] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(쉘 프로그래밍) (2021-04-11)

소스코드이다.

쉘 프로그래밍 - 소스코드

```
#!/bin/bash

# Catch user input for file type.

#echo "Enter the file extension for your image files:"
echo "그림 파일과 확장자명을 입력해주세요. 전체: *"

# Store user input in $files.

read files

# Resize images.

for img in *.$files; do
#convert -resize 20% "$img" "resize-$img"
    convert -resize 40% "original/$img" "convert/$img"

done
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

코드 양은 조금 길지만, 조금 더 시스템에서는 “안전”하게 작성하였다.
헤더 분리하지 않고, 하나의 파일로 작성함. (입문자의 “보편성”을 생각함)

1. Pthread(POSIX)

자바 프로그래밍이나 기타 프로그래밍에서 제공하는 “쓰레드”보다는 조금 난이도가 있을 수 있다.

2. 메모리 소거, 생성에 대해서 중점적으로 소개하였다.

자바 프로그래밍이나 C#, python, basic, cobol과 같은 현대적인 언어 프로그래밍에서는 다루기가 쉽지 않은 주제이다.
php는 unset이라고 해서 객체지향에서 메모리를 소거하는 기능이 있다.

* delete[], delete 몇 번 넣어보면, 오류가 쉼 없이 “계속” 터지는 경험을 할 수 있을 것으로 보인다.

저급한 레벨(Low Level)에 가까운 프로그래밍에 대해서 조금 소개해 보고자 했다.

자바와 현대적인 프로그래밍 언어에서 제공하는 자동 가비지 컬렉션에 익숙해지기 전에 한번은 고민을 해봤으면 한다.

쓰레기
쓰레기
봉투
봉투

그림 1) 가비지컬렉션의 예
(한국에서의 가비지컬렉션)

쓰레기
봉투

Figure 1) Example of garbage collection
(Garbage collection in Korea)

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

C언어의 구조체(Struct), Pointer(포인터) *, **, ***, ****, 포인터 배열 등 한번은 경험을 꼭 해봐야 한다.

3. c의 구조체, 포인터 어려운 주제이지만, 모르면 안 된다.

운영체제 수준에 저급한 시스템을 접근하고자 했을 때는 꼭 경험해봐야 하는 주제이다.

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
/*
 * 주제명(Subject): 사진 용량 변환기
 * 생성일자(Create Date): 2021-04-11 (일요일 / Sunday)
 * 파일명(File Name): convert_img.cpp
 * 저자(Author): 도도(Dodo)
 * 라이센스(License): GNU/GPL v3 License
 * 작성언어(Programming): G++(C++ 표준)
 * 운영체제(Operation System): Ubuntu Desktop 20.04 (2021-04-11)
 * -> 우분투(Ubuntu / Canonical Ltd), https://ubuntu.com, GNU
 * -> CentOS(CentOS), https://www.centos.org, GNU
 * 참고도구(Reference Tools): ImageMagick (https://imagemagick.org), GNU/GPL v3 License
 * 비고(Description):
 * 1. 최초 작성, 도도(Dodo), 2021-04-11 11:31
 * 2. 한 개 파일로 작성함, 2021-04-11
 * 3. g++ convert_img.cpp -lpthread -o convert_img.run
 * 4. 코드 작성완료, 2021-04-11 15:26
 */
*/
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
#include <iostream>
#include <unistd.h>
#include <stdlib.h>
#include <pthread.h>      // POSIX 방식
#include <cstring>
#include <string>
#include <sstream>
#include <fstream>
#include <dirent.h>        // POSIX 방식

using namespace std;

const int MAX_SIZE = 100;
const char* DEFAULT_ROOT_DIR = "/home/사용자명";
const char* DEFAULT_DETAIL_DIR = "/사진/batch-convert-photo";
const char* DEFAULT_ORIGINAL_DIR = "/original";
const char* DEFAULT_CONVERT_DIR = "/convert";
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
class ImageToConvertor{  
  
private:  
    int rate ;  
    char* customDir ;  
  
public:  
    int funcRunner( int service ) {  
  
        int choose_progress_type ;  
        int choose = 1;  
        int usrRate ;  
        int result;  
  
        char* strFileName = new char[ MAX_SIZE ];  
        char* strUsrDirName = new char[ MAX_SIZE ];  
  
        string strType ;  
        string strDir ;
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
// 기본 비율
this->setRate(40);

while ( choose != 0){

    cout << "-----";
    cout << endl ;
    cout << "작업하고자 하는 메뉴를 선택하세요.";
    cout << endl ;
    cout << "Select the menu you want to work on.";
    cout << "-----";
    cout << endl ;
    cout << "1번 - 압축 비율 설정(Compression ratio setting) ";
    cout << endl ;
    cout << "2번 - 경로 변경(Change path) ";
    cout << endl ;
    cout << "3번 - 압축 진행(Compression progress) ";
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
cout << endl ;
cout << "0번 - 돌아가기(Go Back)" ;
cout << endl ;
cout << "-----";
cout << endl ;
cout << "번호를 입력하세요." << endl ;
cout << "<<Please enter your number.>>" << endl ;

cin >> choose;

switch ( choose ) {

    case 1:
        cout << "비율을 입력하세요." ;
        cout << endl;
        cout << "(Enter the percentage.)" ;
        cout << endl;
        cin.clear();
        cin >> usrRate ;
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
// 잘못된 입력
if ( cin.fail() ){
    cout << "잘못된 입력이에요.";
    cout << endl;
    cout << "(This is an incorrect input.)";
    cout << endl;

    cin.ignore(256, '\n');
    cin.clear();
}
else{
    this->setRate ( usrRate );
}

cout << "정상적으로 입력되었습니다.";
cout << endl;
cout << "(It was processed normally.)";
cout << endl;

sleep(1);
system("clear");

break;
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

case 2:

```
cout << "사용자 경로, 파일 위치로 수정할 경우, 입력하세요.";
cout << endl;
cout << "(If you want to modify the ";
cout << "user path or file location, enter it.)";
cout << endl;

cin.ignore();
getline ( cin, strDir );

strcpy(strUsrDirName, strDir.c_str() );
// 예1: GCC => strcpy
// 예2: VC => strcpy_s( origin, size, target);
//strcpy_s( strFileName, 10, strTxt.c_str() );

this->setCustomDir ( strUsrDirName );

cout << "정상적으로 입력되었습니다.";
cout << endl;
cout << "(It was processed normally.)";
cout << endl;
sleep(1);
system("clear");
break;
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
통상적으로 작성하는 if  
If ( result == 0){  
}
```

- * 코드를 어떻게 하면,
가독성을 높일까?
- * 코드 작성을 어떻게 할 것인지
생각해보고 작성하기 바람.
- * 많이 짜봐야 생김.
쉽지 않음.
(코드 작성 습관 고치기 등)

```
case 3:  
    cout << "1번 - 기본경로(Default path)";  
    cout << endl;  
    cout << "2번 - 사용자 경로(User path)";  
    cout << endl;  
  
    cin.ignore();  
    cin >> choose_progress_type ;  
  
    strFileName = this->getProgressFileName ( choose_progress_type, strFileName );  
  
    // cout << strFileName << endl ;  
    result = this->fileToExist( strFileName );  
    (result == 0)?this->fileToConvertor( strFileName ):printToFileError();  
  
    sleep(5);  
    system("clear");  
  
    break;  
}
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
}

delete[] strFileName;
delete[] strUsrDirName;

return 0;
}
```

delete[] 작성해놨다고,
전부 delete[]로 종결하는 게 아님.

다 차원 배열[2~3~4차원]
메모리 접근 방법이 달라짐.

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
char* getProgressFileName( int choose, char* usrTxt ){

    int result = -1;
    usrTxt[0] = 'W0'; // 초기화

    switch ( choose ){

        case 1:
            strcat ( usrTxt, DEFAULT_ROOT_DIR );
            strcat ( usrTxt, DEFAULT_DETAIL_DIR );
            cout << usrTxt;
            break;

        case 2:
            strcpy ( usrTxt, this->getCustomDir () );
            break;

    }

    return usrTxt;
}
```

Const 변수를 기능으로 활용함.

VC11 ~ [순정이 아님]
=> strcpy_s ();
[Visual C++ 컴파일러에서의 취약점임]

gcc, g++에서는
strcpy_s, strcat_s는 해당사항 없음.

MS개발을 하는 경우에는 닷넷을 권장함.
(주력으로 밀고 있음)

MS 운영체제에서도 배울 점이 있음.

입문자에게는 MS운영체제가
조금 쉽게 접근이 가능함. 직관적임.

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
int printToFileError(){

    cout << "파일 경로 또는 폴더 경로를 확인하세요.";
    cout << endl;
    cout << "(Check the file path or folder path.)";
    cout << endl;
    return 0;
}

int printToFileSuccess(){

    cout << endl;
    cout << "성공적으로 출력을 완료하였습니다.";
    cout << endl;
    cout << "(Printing has been completed successfully.)";
    cout << endl;
    return 0;
}
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
int fileToExist( char* strFilename ) {  
  
    ifstream ifile ;  
    ifile.open ( strFilename );  
  
    if ( ifile ) {  
        return 0;  
    }  
    else{  
        return 1;  
    }  
  
}  
  
int fileToConvertor(char* strFilename){  
  
    //DIR * dirp = opendir( strFilename );  
    int restrictFile = 0;  
    int index = 0;  
    int rate = this->getRate();  
  
    char* strName ;
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
char* strOriginalDir = new char[ MAX_SIZE ] ;
char* strConvertDir = new char[ MAX_SIZE ];
char* strBaseCompressCmd = NULL;
char* strArgvOriginFileCmd = new char[ MAX_SIZE ];
char* strArgvConvertFileCmd = new char[ MAX_SIZE ];
char* strCompressCmd = new char[ MAX_SIZE * 2 ];
char* strMakeFolderCmd = new char[ MAX_SIZE ] ;

strOriginalDir = this->getOriginalDir ( strOriginalDir, strFilename );
strConvertDir = this->getConvertDir ( strConvertDir, strFilename );

strBaseCompressCmd = this->baseCompressCmd () ;

if ( fileToExist( strOriginalDir ) == 0 ){
    strMakeFolderCmd[0] = '₩';
    strcat ( strMakeFolderCmd, "mkdir -p" );
    strcat ( strMakeFolderCmd, " " );
    strcat ( strMakeFolderCmd, strOriginalDir );

    system ( strMakeFolderCmd );
}
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
if ( fileToExist( strConvertDir ) == 0 ){
    strMakeFolderCmd[0] = '₩';
    strcat ( strMakeFolderCmd, "mkdir -p" );
    strcat ( strMakeFolderCmd, " " );
    strcat ( strMakeFolderCmd, strConvertDir );

    system ( strMakeFolderCmd );
}

dirent * dp;
DIR *dirp = opendir ( strOriginalDir ) ;

while ( (dp = readdir(dirp)) !=NULL ) {

    strName = dp->d_name ;

    restrictFile = strcmp( strName , ".");
    restrictFile *= strcmp ( strName, "..");
    restrictFile *= strcmp( strName , "convert_img.cpp" );
    restrictFile *= strcmp ( strName, "convert_img.run" );
    restrictFile *= strcmp ( strName, "batch-target.sh" );
    restrictFile *= strcmp ( strName, "batch-resize.sh" );
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
if ( restrictFile != 0 )
{
    strArgvOriginFileCmd = this->argvOriginFileCmd( strArgvOriginFileCmd,
                                                    strOriginalDir,
                                                    strName );

    strArgvConvertFileCmd = this->argvConvertFileCmd( strArgvConvertFileCmd,
                                                    strConvertDir,
                                                    strName );

    //cout << strBaseCompressCmd << " ";
    //cout << strArgvOriginFileCmd << " ";
    //cout << strArgvConvertFileCmd ;
    //cout << endl ;

    strCompressCmd[0] = 'W';
    strcat ( strCompressCmd, strBaseCompressCmd );
    strcat ( strCompressCmd, " " );
    strcat ( strCompressCmd, strArgvOriginFileCmd );
    strcat ( strCompressCmd, " " );
    strcat ( strCompressCmd, strArgvConvertFileCmd );
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
this->printProgressCompress ( index, strCompressCmd );
system( strCompressCmd );
this->printCompleteCompress ( index );

}

index++;
}

(void)closedir(dirp);

this->printToFileSuccess();
//delete strName ;
strName = NULL;
// closedir
delete[] strOriginalDir ;
delete[] strConvertDir ;
delete[] strBaseCompressCmd ;
delete[] strArgvOriginFileCmd ;
delete[] strArgvConvertFileCmd ;
delete[] strCompressCmd ;
delete[] strMakeFolderCmd;

return 0;
}
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
void printProgressCompress(int index, char* cmd){  
    cout << index << "(인덱스/index):";  
    cout << cmd ;  
    cout << endl ;  
}  
  
void printCompleteCompress(int index, int msg = 0){  
    cout << "-> 변환 반응(Transformation reaction): " ;  
    cout << rate ;  
    cout << "%:";  
    cout << "성공(Success)";  
    cout << endl ;  
}
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
char* baseCompressCmd () {  
    char* strCmd = new char[ MAX_SIZE ] ;  
    std::stringstream ssInt;  
    int rate = this->getRate();  
  
    ssInt << rate;  
  
    strCmd[0] = 'W';  
    strcat ( strCmd, "convert -resize ");  
    strcat ( strCmd, ssInt.str().c_str() );  
    strcat ( strCmd, "%" );  
  
    return strCmd;  
}
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
char* argvOriginFileCmd ( char* argvCmd, char* originDir, char* fileName){

    argvCmd[0] = 'W0';      // 초기화

    strcat ( argvCmd, originDir );
    strcat ( argvCmd, "/" );
    strcat ( argvCmd, fileName );

    return argvCmd ;
}

char* argvConvertFileCmd ( char* argvCmd, char* convertDir, char* fileName){

    argvCmd[0] = 'W0';      // 초기화

    strcat ( argvCmd, convertDir );
    strcat ( argvCmd, "/" );
    strcat ( argvCmd, fileName );

    return argvCmd ;
}
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
char* getOriginalDir( char* usrDir, char* homeDir ){

    usrDir[0] = '₩0'; // 초기화
    strcat ( usrDir, homeDir );
    strcat ( usrDir, DEFAULT_ORIGINAL_DIR );

    return usrDir;
}

char* getConvertDir( char* usrDir, char* homeDir ){

    usrDir[0] = '₩0'; // 초기화
    strcat ( usrDir, homeDir );
    strcat ( usrDir, DEFAULT_CONVERT_DIR );

    return usrDir;
}
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
void posixfileDir( char *strFileName ){

    DIR * dirp = opendir( strFileName );
    dirent * dp;

    while ( (dp = readdir(dirp)) !=NULL ) {
        cout << dp->d_name << " size " << dp->d_reclen<<std::endl;
    }

    (void)closedir(dirp);
}

int getRate() const{
    return this->rate;
}

void setRate(int rate) {
    this->rate = rate;
}
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
char* getCustomDir () const {
    return this->customDir;
}

void setCustomDir ( char* usrDir ) {
    this->customDir = usrDir ;
}
```

protected:

};

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
class MagicMain{  
private:  
  
public:  
    void run(){  
        ImageToConvertor* convertor = new ImageToConvertor();  
        int choose = 1;  
        int msg = -1;  
  
        while ( choose != 0 ){  
            cout << "-----";  
            cout << endl ;  
            cout << "ImageMagicK로 변환 (압축)";  
            cout << "(Convert To ImageMagicK(compress))";  
    }  
}
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
cout << endl ;
cout << "최초 빌드: 2021-04-11 15:24 [일요일]" << endl;
    cout << "First build: 2021-04-11 15:24 [Sunday]" << endl;
cout << "버전(Version): 0.01" << endl;
cout << "-----";
cout << endl ;
cout << "작업하고자 하는 메뉴를 선택하세요.";
cout << endl ;
cout << "Select the menu you want to work on.";
cout << "-----";
cout << endl ;
cout << "1번 - 센토스 (CentOS) DNF - ImageMagick 설치" ;
cout << endl ;
cout << "2번 - 우분투 (Ubuntu) apt - ImageMagick 설치";
cout << endl ;
cout << "3번 - 센토스 (CentOS) YUM - ImageMagick 설치" ;
cout << endl ;
cout << "4번 - 사진 압축(Photo compression)";
cout << endl ;
cout << "5번 - 도움말(Help)";
cout << endl ;
cout << "0번 - 종료(Exit)";
cout << endl ;
cout << "-----";
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
cout << endl ;
cout << "번호를 입력하세요." << endl;
cout << "(Please enter your number.)" << endl ;

cin >> choose;

switch ( choose ){

    case 1:
        system("sudo dnf install -y imagemagick");
        cout << "정상적으로 처리되었습니다.";
        cout << endl;
        cout << "(It was processed normally.)";
        cout << endl;

        sleep(2);

        break;
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
case 2:  
    system("sudo apt-get -y install imagemagick");  
  
    cout << "정상적으로 처리되었습니다.";  
    cout << endl;  
    cout << "(It was processed normally.)";  
    cout << endl;  
  
    sleep(2);  
  
    break;  
  
case 3:  
    system("sudo yum install -y imagemagick");  
    cout << "정상적으로 처리되었습니다.";  
    cout << endl;  
    cout << "(It was processed normally.)";  
    cout << endl;  
  
    sleep(2);  
    break;
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

case 4:

```
msg = convertor->funcRunner( choose );
break;
sleep(2);
```

case 5:

```
cout << "-----" << endl;
cout << "5번 - 도움(Help)" ;
cout << endl ;
cout << "-----" << endl;
cout << "컴파일 방법(How to compile)" << endl ;
cout << "g++ convert_img.cpp -lpthread -o convert_img.run";
cout << endl << endl;
cout << "라이센스(License): GNU/GPL v3 License" << endl ;

cout << "운영체제(Operation System): " << endl;
cout << "--> 1. Ubuntu(우분투) - (https://ubuntu.com), GNU" << endl ;
cout << "--> 2. CentOS(센토스) - (https://www.centos.org), GNU" << endl ;
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
        cout << endl ;
        cout << "-----" << endl;
        cout << "라이브러리(Library)" << endl ;
        cout << "--> 1. ImageMagick(https://imagemagick.org), 1999~, GNU/GPL v3 License" << endl ;
        sleep(5);

    };

    system("clear");

}
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
delete[] convertor;

}

protected:

};

void* thread_convertor(void *arg) {

    MagicMain* app = new MagicMain();
    app->run();

    delete[] app;

    return NULL;
}
```

12. [부록2] 리눅스 - 이미지 압축 자동화 생성 도구 / 개발 - 소스코드(std g++) (2021-04-11)

프로그래밍 소스코드이다.

C++ 프로그래밍 - 소스코드 (convert_img.cpp)

```
int main(){

    pthread_t thread;
    pthread_create(&thread, NULL, thread_convertor, NULL ) ;

    pthread_join(thread, NULL);

    return 0;
}
```

감사합니다
