

**REPORT**

|  |  |  |
| --- | --- | --- |
| 과목명 | : | 스크립트프로그래밍01 |
| 제출일 | : | 06월 01일 |
| 학번 | : | 2020213311 |
| 성명 | : | 신중근 |
|  |  |  |

**목차**

**소개3**

**기능 설명3**

화면 표시3

파일4

파일 열기4

파일 저장4

이미지 편집5

**데이터베이스 구축6**

저장6

조회8

로드9

삭제10

**참고자료10**

**결론10**

1. **소개**

이 코드는 이미지 편집을 위한 프로그램입니다. 프로그램은 tkinter 라이브러리를 사용하여 GUI 환경에서 작동하며, 사용자는 이미지 파일을 선택하고 다양한 이미지 편집 작업을 수행할 수 있습니다. 편집 기능에는 이미지 필터링, 명암 조절, 반전 및 회전 등 다양한 기능이 구현되어 있습니다. 또한, 사용자가 작업한 이미지는 SQLite 데이터베이스에 저장되어 추후에도 접근할 수 있도록 합니다.

1. **기능 설명**
   1. 화면 표시

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

**이미지를 화면에 표시하는 함수**

14: 창 크기 설정: 인자로 받아온 width와 height를 기반으로 창 크기 설정

15-16: canvas가 이미 존재한다면 제거

18-20: 캔버스와 이미지 객체 생성

21-28: RGB 문자열 생성: convert(‘RGB’)를 사용해 img를 RGB 형식으로 변환 후 각 픽셀의 값을 for문과 getpixel()을 사용해 문자열로 변환

29-30: 생성된 RGB 문자열을 이용하여 이미지 생성 후 캔버스에 배치. 캔버스는 창에 배치. padx와 pady로 여백 설정

* 1. 파일
     1. 파일 열기

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

**이미지 파일을 열고 화면에 표시하는 함수**

35-37: askopenfilename() 함수를 사용해 파일 읽음. jpg, png, gif, jpeg, bmp, tif 형식의 파일 선택 가능

38-40: 읽어온 파일을 Image.open() 함수로 열고, convert(‘RGB”)로 RGB 형식으로 변환 후 이미지의 너비와 높이 정보 저장

42-44: 이미지와 이미지 정보 복사

45: 복사한 이미지를 창에 표시

* + 1. 파일 저장

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

**화면에 표시중인 이미지를 저장하는 함수**

53: asksaveasfile() 함수를 사용해 이미지 저장 경로 선택. 확장자는 jpg 형식으로 저장

56: save() 함수를 사용해 photho2 이미지를 저장. saveFp는 경로와 이름을 나타낸다.

* 1. 이미지 편집

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

**이미지를 확대하는 함수**

78-91: 확대 배율을 선택하는 창 열기. 사용자는 수치 조정 바를 이용하여 배율 설정. 설정한 값은 double 자료형으로 변수명 scale\_value에 저장. 값은 1부터 4까지 0.01단위로 설정 가능.

67-76: [확인] 버튼을 클릭하면 시행되는 함수. get() 함수를 사용하여 사용자가 선택한 값 불러와 변수명 value에 저장. 그 후 resize()함수를 이용하여 기존 너비와 높이에 value를 곱한 값으로 photho2의 크기 변경. 변경한 photo2를 displayImage() 함수를 이용하여 화면에 표시.

이 외에도 축소, 반전, 회전, 밝게, 블러, 부드럽게 등 여러 기능이 있다. 기능들 대부분이 이와 비슷하거나 간단하기 때문에 생략

1. **데이터베이스 구축**
   1. 저장

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

**이미지의 이름을 입력받는 함수**

364-365: 이름을 입력받는 Entry 위젯 생성

367-369: input\_name 함수를 실행시키는 버튼

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

**이름 입력이 빈 문자열인지 확인하는 함수**

빈 문자열이 입력되었다면 경고 메시지 출력 후 종료. 아니라면 insert\_database 함수에 입력한 이름을 삽입하여 실행

텍스트, 스크린샷, 문서, 폰트이(가) 표시된 사진

자동 생성된 설명

**이미지를 데이터베이스에 저장하는 함수**

304: os.path.expanduser() 함수를 사용하여 경로를 Documents/imgDB로 설정

306-307: sqlite3를 사용해 데이터베이스 연결

309-310: imageDB라는 이름의 테이블 생성. 변수는 id, name, extension, image가 있다.

313-321: 이름 중복 체크. SELECT를 사용하여 imageDB 테이블에 name 열을 조회 후 WHERE을 사용하여 name 값이 변수 name과 일치하는 행을 찾는다. 그 후 fetchone()을 사용하여 찾은 행들 중 첫 번째 행을 existing\_name 변수에 저장. 만약 값이 있다면 중복되는 이름이 있다는 뜻이므로 덮어쓰기 확인 창을 생성. [no]를 클릭한 경우 함수 종료

323: format을 사용해 원본 파일의 확장자를 가져온다

325-327: 메모리 내에서 바이트 데이터를 읽고 쓰기 위한 임시 버퍼를 생성. 편집한 이미지(photo2)를 JPEG 형식으로 버퍼에 저장. getvalue()를 사용하여 버퍼의 데이터를 바이트로 가져온다.

329-334: 이미지 이름이 기존에 존재하는 경우(중복) UPDATE를 사용하여 기존에 있던 데이터에 업데이트하고, 이름이 기존에 존재하지 않는 경우에 INSERT를 사용하여 새로운 레코드를 삽입한다.

336-338: 저장 완료 창을 띄운 뒤 변경사항을 커밋하고 DB 연결을 종료

* 1. 조회

텍스트, 스크린샷, 문서, 폰트이(가) 표시된 사진

자동 생성된 설명

**데이터베이스에서 이미지 이름들을 가져와 이미지를 선택할 수 있는 창을 생성하는 함수**

377-380: path 경로의 데이터베이스에 연결

382-384: SELECT를 사용하여 imageDB의 모든 이름을 가져오고 연결 종료

386-407: 이미지가 있는 경우 새 창 생성.

395-396 라인의 for문을 통해 리스트박스에 이미지 이름을 추가. insert() 함수를 사용하여 리스트박스 마지막(END)에 현재 이미지 이름(row[0]) 값을 추가.

408-409: DB에 저장된 이미지가 없다면 로드 실패 메시지 표시

* 1. 로드

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

**선택한 이미지를 데이터베이스에서 로드하여 화면에 표시하는 함수**

415: 사용자가 리스트 박스에서 선택한 값을 가져온다.

416-419: 지정된 경로의 데이터베이스에 연결

421-423: SELECT를 사용하여 선택한 이름의 이미지 데이터를 가져오고 데이터베이스 연결 해제

425-435: 데이터를 가져오는데 성공했을 경우. 로드 완료 메시지 표시. io.BytesIO를 사용하여 row[0]에 저장된 이미지 데이터를 이미지 객체로 변환하고, convert()를 사용하여 RGB 형식으로 변환. 가져온 이미지를 원본 이미지 변수(photo)에 저장. photo2 변수에 원본 이미지를 복사하고 displayImage()를 사용하여 복사한 이미지를 창에 표시

436-437: 가져온 데이터가 없는 경우 로드 실패 메시지

* 1. 삭제

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

**선택한 이미지를 데이터베이스에서 삭제하는 함수**

443: 사용자가 리스트 박스에서 선택한 값을 가져온다.

444-447: 지정된 경로의 데이터베이스에 연결

449: DELETE를 사용하여 선택한 이미지 이름의 데이터를 삭제

450-451: 데이터베이스에 변경사항을 적용 후 연결 해제.

453-455 삭제 완료 메시지 표시 후 기존 이름 선택 창을 닫고 load\_database() 함수를 사용하여 리스트 박스를 업데이트

1. **참고자료**

파이썬 for Begginer(지정도서), 구글링, chatGPT

1. **결론**

이 프로그램은 사용자가 이미지 편집 기능을 쉽고 편리하게 수행할 수 있는 기능을 제공합니다. tkinter를 통해 제공되는 사용자 친화적인 인터페이스는 이미지 선택 및 처리 과정을 간단하게 만들어줍니다. 또한, SQLite 데이터베이스를 활용하여 사용자가 작업한 이미지를 보관하고 추후에도 사용할 수 있도록 합니다. 이를 통해 사용자는 이미지 처리 작업을 보다 효율적으로 수행할 수 있으며, 필요한 경우 작업 이력을 확인하고 수정할 수 있습니다.