

## 2024 소프트웨어프로젝트 프로젝트 6

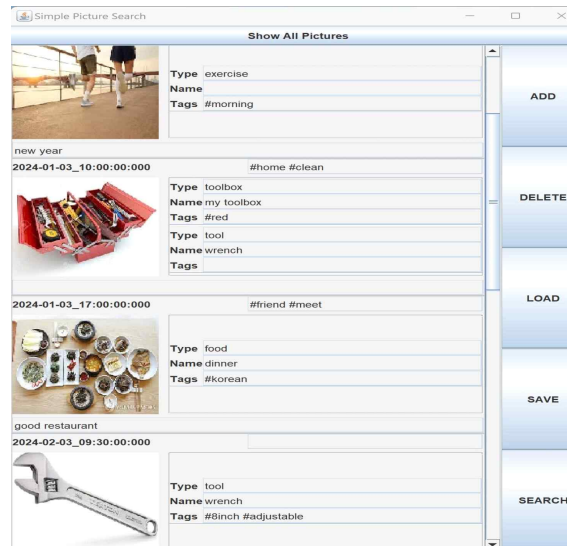
### Picture Search 프로그램 최종 설계 및 구현

#### 1. 내용

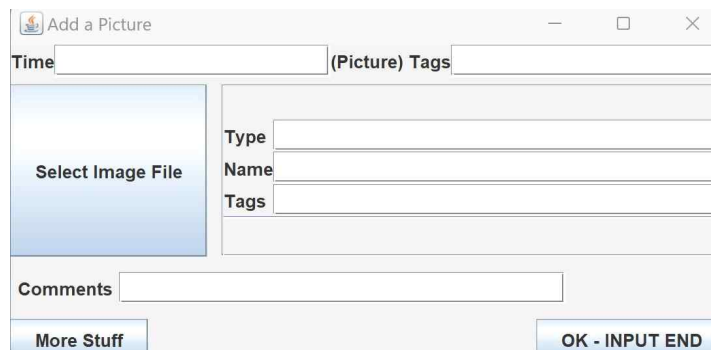
프로젝트 6의 GUI에 이벤트 핸들링을 추가해서 Picture Search 프로그램을 완성한다.

주의. 프로젝트 5는 따로 제출하여야 함. (05/26 마감)

- 프로그램을 수행하면, 정해진 파일 (예, picture-normal.data)에 있는 사진/사물 정보들을 읽어 와서 내부 자료구조에 저장하고, 시간 순으로 표시한다. 파일에서 읽어 오는 프로그램 코드는 프로젝트 3의 것을 사용.



- 메인 화면의 오른쪽에 있는 “ADD”버튼을 누르면, 사진과 사물 정보를 입력해서 새로운 picture 정보를 추가할 수 있는 “Add Picture” 화면이 생성된다.
- “Delete”버튼을 누르면, 선택된 picture를 화면과 메모리에 있는 picture list에서 제거한다. (특정 picture가 선택되지 않은 경우, 먼저 선택하라는 dialogue를 띄운다.)
- “LOAD” 버튼을 누르면, 현재 상태, 즉, 메모리에 있는 picture 정보에 상관없이, 파일에 있는 picture 정보들로 메모리의 picture list를 새로 구축한다. 당연히 화면도 파일에 있는 picture 정보들로 새로 표시되어야 한다.
- “SAVE”버튼을 누르면, 현재 화면에 표시된 picture 들 정보를 정해진 파일에 저장한다. (별도로 “Save”를 하지 않으면, picture 정보의 갱신은 프로그램이 종료되면 무효화된다.)



- “Add Picture” 화면에서 사진 image 파일을 지정하고, 다른 정보들을 키보드로 입력한다.
- “Select Image File” 버튼을 누르면, Java의 FileChooser Dialogue 화면이 나오고 사진 이미지 파일을 선택할 수 있도록 한다.
- 항목 입력은 생략될 수 있으며, “Time” 정보는 입력되지 않으면 현재 시각으로 정해진다.

- 사진 속 사물이 두 개 이상일 경우에는 하단의 “More Stuff”를 입력해서 복수 개의 사물 정보를 입력할 수 있다.
- 입력을 마친 후, “OK - INPUT END” 버튼을 누르면, 해당 입력 내용을 담은 picture 정보가 추가되어 메인 화면을 갱신하고 검색 화면을 닫는다.
- 입력한 정보를 무시하려면 우측 상단의 close 버튼을 이용하여 화면을 닫는다.
- “SEARCH” 버튼을 누르면, 검색 정보를 입력하고 동작시키는 “Search Picture” 화면이 생성된다.

The screenshot shows a window titled "Search Picture". It has three main sections: "Time Search" with "From" and "To" input fields (placeholder: yyyy-MM-dd\_HH:mm:ss), "Keyword Search" with "Tags", "Type", "Name", and "Tags" input fields, and "General Search" with a single input field. At the bottom, there are three buttons: "AND SEARCH", "OR SEARCH", and "CLOSE".

- “Search Picture” 화면에서는 시간 조건과 키워드로 picture 검색을 할 수 있다.
- 원하는 키워드를 해당 항목에 입력하고, 키워드를 모두 만족하는 picture 정보를 원하면 “AND SEARCH” 버튼을, 하나라도 만족하는 picture 정보를 원하면 “OR SEARCH” 버튼을 누른다.
- General Search 항목은 찾으려는 키워드가 어떤 항목에 해당되는지 모를 때 사용한다.
- 메인 화면에는 검색 결과, 즉, 검색 조건에 해당되는 picture들만 표시된다.
- picture 검색은 현재 메인 화면에 나와 있는 picture들을 대상으로 이루어진다. 즉, 검색은 누적으로 진행된다. 누적에서 벗어나서 새로운 검색을 하려면, 메인 화면의 최상단에 있는 “Show All Pictures” 버튼을 눌러서 모든 picture 정보들이 화면에 나오도록 한 후, 검색을 실시한다.

## 2. 목적

- 객체지향 프로그램 설계 연습
  - \* 필요 Class 정의, 클래스별로 기능 나누기
  - \* 프로그램 크기의 증대에 따른 관리 문제 및 Java의 해결법 체득
- 한 학기 배운 모든 주요 Java 기능 체득
  - \* 기본 GUI, File I/O, 스트링 처리 (간단한 DB 프로그래밍)
  - \* Event-driven programming, object reference 처리
  - \* incremental programming
- 개발도구 활용법, 디버깅
- 프로그램 개발 과정 체득

## 3. 추진 방법

- 1) 서로 독립적인 Class들 사이에서 자료 공유하는 방법 연구
  - \* 이번 과제에서는 JFrame을 기반으로 하는 클래스들이 복수 개 생성됨.
  - \* 그들 간의 정보 교환 필요
    - (예, picture info 창에서 “OK” 경우, 입력 내용이 메인 화면으로 전달되어 갱신되어야 함)
  - \* 프로그램 구조의 틀을 유지하면서 정보 교환

- 2) 또한, 프로그램 내부에 있는 picture list (즉, picture DB)와 화면에 보여지는 내용을 항상 일치시켜야 함.
- 3) 클래스 구조를 유지하면서 위 두 문제를 동시에 해결하는 방법을 찾아야 함.
- 4) DELETE 기능 구현을 위해서는, 메인 화면에 나오는 여러 picture 중 어떤 picture가 선택되었는지 파악하는 기법도 필요. (Mouse Event로 처리하는 것이 일반적이지만, 한계가 있음. 따라서 선택이 유효한 영역을 가정하는 것이 좋음)

#### 5) 설계 사항

- 사용자에게 제공되는 GUI 설계 (프로젝트 5)
- 어떤 class 들과 어떤 파일로 구성을 할 것인지 결정
- 다시 한번, **구조 > 기능 > 성능**
- 어떤 기능을 어떻게 구현할 것인가는 설계자의 권한. 단, 결정 후 리포트에 명시.
  - \* 요구되는 최소 기능만 구현하면 되며, 구현 방법은 설계자의 선택

#### 6) 프로그램 완성 제출

### 4. 평가항목

- 프로그램의 동작 여부 (70%)
  - \* **동작 여부를 확연히 알아볼 수 있는 결과**
  - \* 본인이 디버깅해가면서 프로그래밍을 했다면, 설명을 대충해서 오해를 만들겠는가?
  - \* 본인이 작성한 프로그램의 진위 여부를 판단하는데 도움 되는 인증 자료 포함
- 각 단계에 대한 설계 노트 (30%)
  - \* 어떤 이유로 어떤 결정을 내렸는지 명확하게 설명
  - \* 설계 및 구현에 대한 자체 평가: 한계점 및 개선 방안 제시
- 자체평가표 참조
- 평가에 대해 불안해 할 필요 없음. 불안한 사항이 있으면, 리포트에 찾기 쉽게 기술
- **picture data에 적어도 1장 이상의 본인 관련 내용 포함 (본인의 과제 수행 여부를 증명)**

### 5. 리포트 제출

6/9(일)까지. (마감 시간 이후에 제출 불가)

- 최종 테스트 과정 화면, 설계 노트, 소스 프로그램, 자체평가표
- 하나로 묶어서 eClass로 제출