



PlayList of Artwork

Data structure design – Personal Project

20164204

컴퓨터공학과

이상민



Goal of Program

- Dynamic Graph 기반의 예술작품 PlayList 구현
 - Dynamic Graph 그래프가 갖는 성질을 활용
 - 노드의 생성과 소멸에 따라 Edge 가 Dynamic 하게
 - 변형되는 그래프
 - 실용성과 효율성에 중점을 둔 프로그램 구현
 - 쉽게 접할 수 있는 PlayList 를 예술작품에 접목
 - 플레이리스트로써 필요한 기능 구현



Dynamic Graph in PlayList

- 재생 목록 = Dynamic Graph
 - 재생목록의 군집
 - (구현부) 동일한 Category (Information) 를 갖는 작품들이 연결된 그래프
 - (추가 , 삭제) User Add // User Delete 에 따라 Dynamic 한 그래프

Node :

Artwork

Edge :

Artwork Information



군집화 ? - 구현부



Edge : 고희



Edge : 서양



Edge : 동양



Edge : 건축물

○ . 작품 정보중 한 항목이 동일 -> 해당 Edge 로 연결 // 마치 군집



군집화 ? - 추가 , 삭제

- 추가

- Artwork – Information
- Information
 - 1. 작가 2. 연도 3. 장르 4. 동양 / 서 양
- Information 을 기준으로
- 기존의 그래프에서 동일한 Information 을 갖는
- 작품들과 Edge 연결

- 삭제

- Artwork
- 해당 Artwork 삭제
- 해당 Artwork 와 다른 Artwork 의
- Edge 모두 연결 해제



Programming - python

- Dynamic Graph

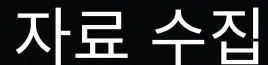
- Dictionary

- Python data structure
 - Key() : Artwork
 - Values() : Information

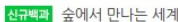
- Programming

- PyQt5 GUI tool

- 입력 , 출력 , 삭제 선택
 - 입력
 - 출력 : What Category
 - 삭제



- 네이버
- 미술 백과
- 조회수에 따라
- 다양한 작품에 대한
- 정확한 정보 제공



정보 제공처

국립고궁박물관

국립중앙박물관

프랑스국립박물관연합(RMN)

한국데이터베이스진흥원 공공저작권 신탁관리시스템

한국사전연구소 한국미술오천년

전체펼치기 ▾

현대미술이 궁금하다면

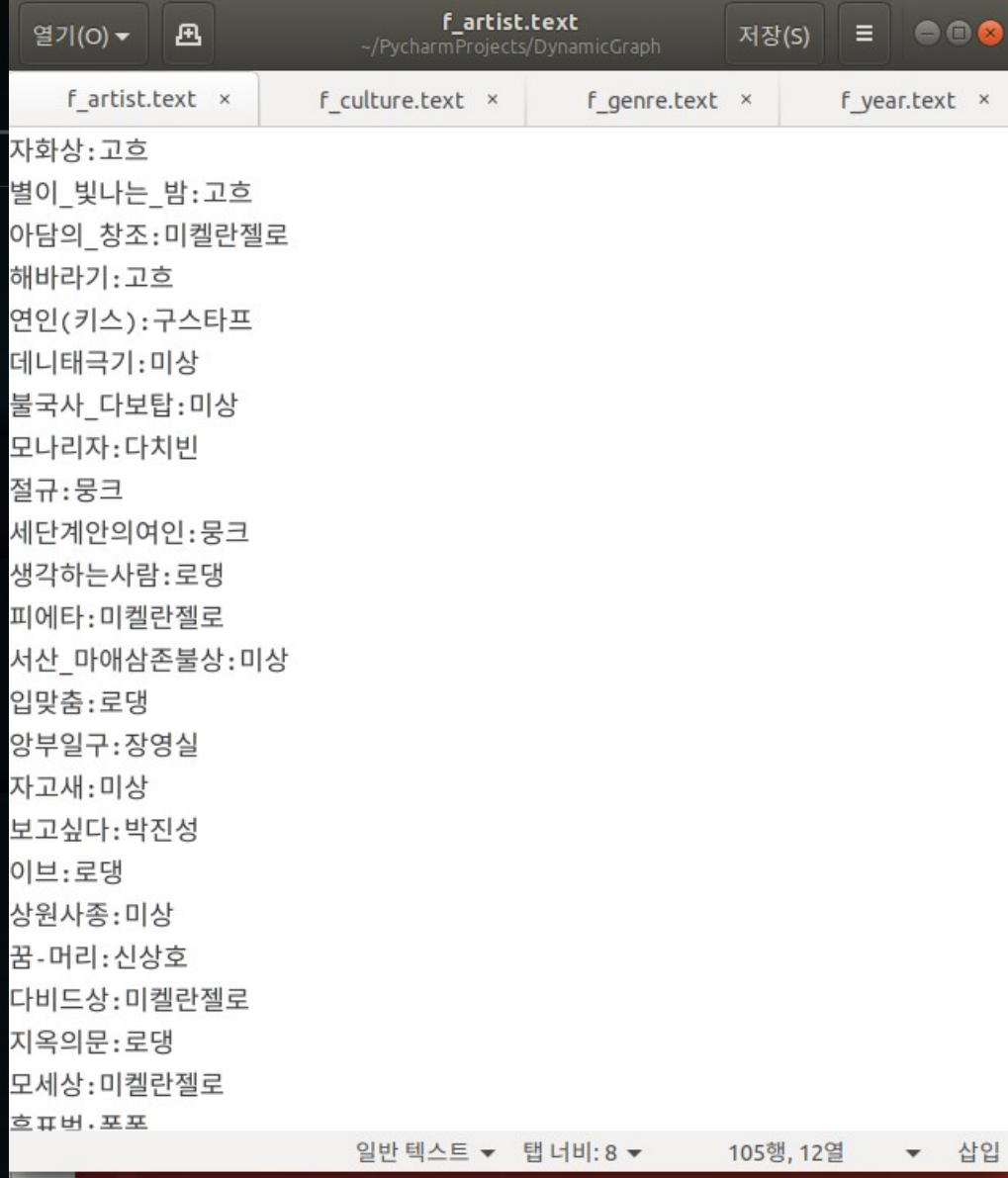
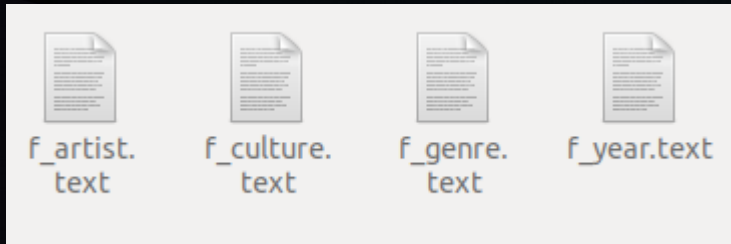
hello!ARTIST

우리 시대 작가들과의 만남



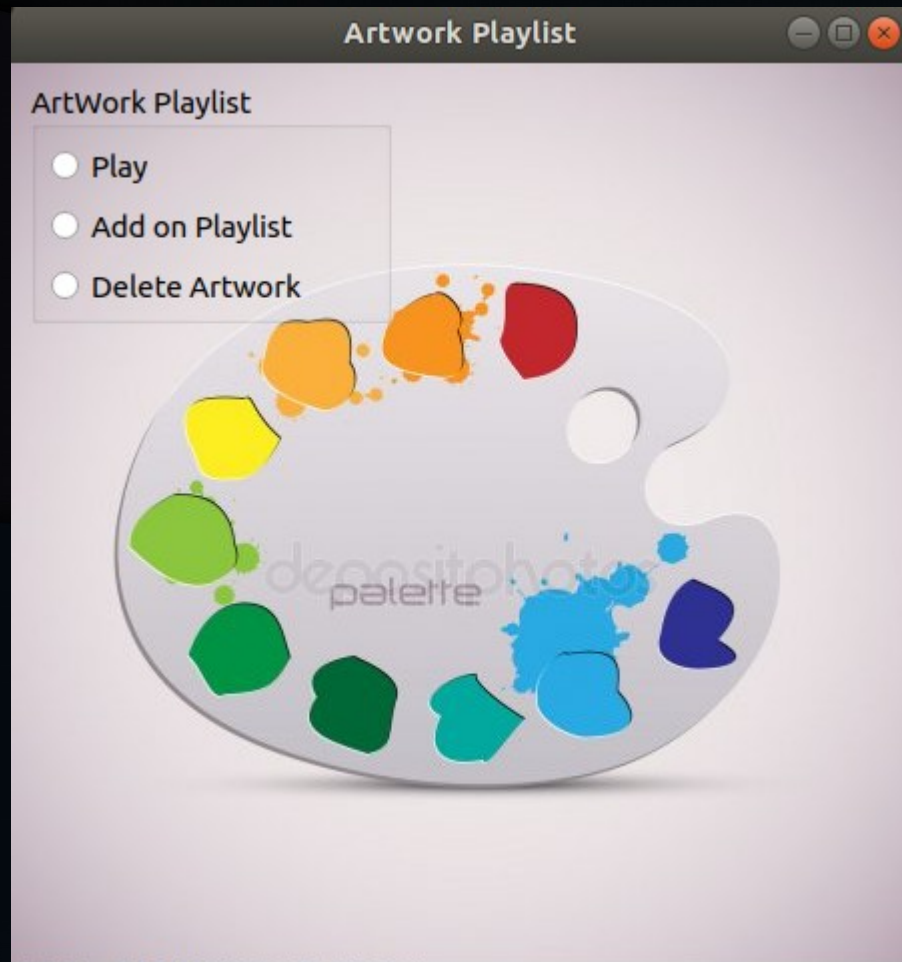
자료 수집 - 파일 연동

-
-
- 총 100 개의 node 수집
- Node 는 사용자의 입력에 따라 끊임없이 추가 - 파일에 저장
- 장르별 조회수가 높은 작품 위주로
- 일차적 자료수집





핵심 코드 - 1. 초기화면 -1



- 프로그램 메인 화면
- 노드 입력 / 삭제 / 그래프 출력 결정
- 선택에 따라 해당 모듈 실행



핵심 코드 - 2. Artwork 추가

The screenshot shows a window titled "Select.py" with standard OS window controls (minimize, maximize, close). Inside the window, there are five text input fields labeled "Artwork", "Artist", "Year", "Genre", and "Culture". To the right of the "Artwork" field is a button labeled "Add on Playlist". To the right of the "Culture" field is a button labeled "Not anymore".

- 프로그램

- 각 값들을 입력 → 기존에 입력받은 Artwork 로 이루어진 재생목록 (dynamic-graph) 구동 → 그래프에서 같은 information(ex: 고흐, 동양 ...) 을 갖는 Artwork 와 모두 연결

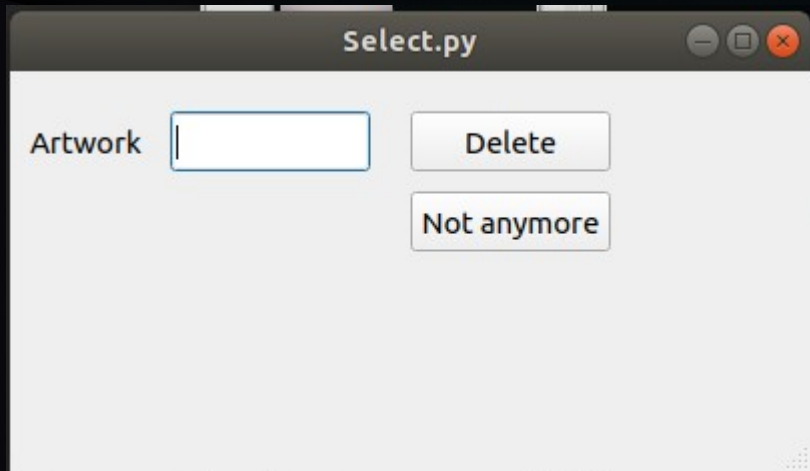
- Node 추가

- Information(Artist/Year/Genre/Culture) : edge
- 같은 Information 값 (ex: 고흐, 동양 ...) 을 갖는 node 끼리 연결
- Information 은 총 4 가지이기 때문에 4 가지 종류의 edge 를 가지며 해당 종류에 따라 다양한 weight 값 (ex: genre : 그림, 조각, 건축 ...) 을 갖는

- Not anymore : 프로그램 종료



핵심 코드 - 3. Artwork 삭제



- Node 삭제 → Node 는 Information 값 lose → edge 해제 → 재생목록 (Dynamic Graph) 에서 Node 삭제

- Node 삭제
 - 다른 node 와 모든 edge 해제
- Not anymore : 프로그램 종료



핵심 코드 - 4. 재생 -1

- ☐ Artist
- ☐ Year
- ☐ Genre
- ☐ Culture

- Dynamic graph 출력 : 특정 edge 로 연결된 작품 node 선별적 출력 ; 입력값에 따라 다이나믹한 Dynamic Graph 표현 가능

- ☒ Artist
- ☐ Year
- ☐ Genre
- ☐ Culture

What do you Want to Play?

미켈란젤로

Play



핵심 코드 - 5. 재생 -2

- ☒ Artist
- ☐ Year
- ☐ Genre
- ☐ Culture

What do you Want to Play? 미켈란젤로

Play

아담의_창조

피에타

다비드상

모세상

라우렌시오도서관

아담의창조

인생의꿈

최후의심판

- Display
- 특정 Edge 로 연결된 Node 출력
- 보고싶은 예술 작품들 선별적 감상 혹은 재생 가능



Q&A

- def Next() :

노드 입력 삭제에 따른 그래프 변화 파일과 연동

프로그램 모듈 연결