

# AI 검색 기능 강화

Status	In progress
Version	1..1
Author	김경수 영어공학과/
Created At	@2025년 8월 27일 오후 12:37
Updated At	@2025년 8월 27일 오후 4:48
Project	성균관대학교 학술전문에이전트

## 하나의 프롬프트가 완성되기까지: 단계별 개선 히스토리

### 들어가며: 좋은 프롬프트는 만들어지는 것이다

하나의 잘 설계된 AI 프롬프트는 마치 잘 짜인 코드나 API와 같습니다. 처음부터 완벽한 프롬프트가 나오는 경우는 드물며, 여러 번의 테스트와 수정을 통해 점차 안정적이고 예측 가능한 결과를 내는 방향으로 진화합니다.

이 글은 'AI 기반 지능형 도서 검색 JSON 생성 프롬프트'가 초기 아이디어 단계에서부터 최종 완성 단계에 이르기까지 어떤 시행착오를 겪었고, 각 단계에서 어떤 문제점을 해결하기 위해 프롬프트의 구조가 어떻게 바뀌었는지를 기록한 문서입니다.

### 1단계: 초기 아이디어와 문제 직면

#### Before (초기 버전의 특징)

처음 아이디어는 AI에게 최대한 많은 자율성을 부여하는 것이었습니다.

- **슈도 코드(Pseudo-code) 형식의 지시:** IF/ELSE나 FUNCTION 같은 프로그래밍 구문을 사용하여 AI가 논리적으로 상황을 판단하도록 유도했습니다.
- **조건부 JSON 출력:** '단순 검색'일 때와 '심층 검색'일 때, 서로 다른 Key를 가진 두 종류의 JSON 중 하나를 선택하여 출력하도록 요청했습니다.

#### Problem (마주한 문제)

결과는 기대와 달랐습니다. AI는 종종 **엉뚱한 형식의 JSON**을 출력하거나, 두 형식을 뒤섞는 등 **예측 불가능한 오류**를 보였습니다.

가장 큰 원인은 **AI는 프로그래밍 컴파일러가 아니라는 점**이었습니다. AI에게 여러 갈래의 복잡한 논리 분기와 그에 따른 다른 결과물 형식을 동시에 요구하는 것은, 사람에게 여러 가지 일을 한꺼번에 설명하며 알아서 잘해주길 바라는 것과 같았습니다. 이는 AI에게 과도한 작업 부담을 주어 안정성을 크게 떨어뜨렸습니다.

### 2단계: 구조 통일과 안정성 확보

#### After (1차 개선)

첫 번째 문제 해결의 핵심은 '단순화'와 '통일'이었습니다.

- **What Changed?**
  1. **슈도 코드 제거:** IF/ELSE 같은 구문을 없애고, "1단계에서는 이것을 하고, 2단계에서는 저것을 하라"는 식의 명확하고 순차적인 자연어 지시로 변경했습니다.
  2. **출력 JSON 형식 통일:** 어떤 경우든 **항상 동일한 구조와 Key를 가진 하나의 JSON**만 출력하도록 규칙을 변경했습니다.
- **Why it Changed (개선 이유)**

AI가 따라야 할 규칙이 하나로 고정되자, 결과의 안정성이 극적으로 향상되었습니다. AI는 더 이상 어떤 형식으로 출력할지 고민할 필요 없이, 정해진 틀 안에서 내용만 채우는 데 집중할 수 있게 되었습니다.

이는 개발자에게도 큰 이점이었습니다. 항상 동일한 구조의 JSON이 반환되므로, 결과를 파싱하고 처리하는 코드가 매우 단순하고 견고해졌습니다.

### 3단계: 표준화와 개발 편의성 증대

#### After (2차 개선 - 최종안)

1차 개선으로 프롬프트는 '동작하는' 수준이 되었습니다. 마지막 단계는 이것을 '전문적이고 실용적인' 수준으로 끌어올리는 것이었습니다.

- What Changed?

1. **JSON Key 영문화**: 추정 제목과 같은 한글 Key를 title 같은 표준 영문 Key로 모두 변경했습니다.
2. **검색 유형의 숫자 ID 부여**: Author + Genre 같은 문자열 대신, queryType: 4 와 같이 각 유형에 고유한 숫자 ID를 부여했습니다.
3. **7가지 검색 유형 명시**: AI가 분류해야 할 7가지 검색 유형을 프롬프트 안에 명확하게 목록으로 정의해주었습니다.

- Why it Changed (개선 이유)

이러한 '표준화' 작업은 실제 시스템과의 연동성을 고려한 것입니다. 영문 Key는 전 세계 개발 환경의 표준으로 호환성 및 인코딩 문제를 방지합니다. 숫자 ID는 코드에서 if/else나 switch 구문으로 분기 처리할 때 문자열을 비교하는 것보다 훨씬 효율적이고 오류 발생 가능성이 적습니다.

#### 최종 요약: 프롬프트 진화의 방향

버전	특징	장점	단점
초기 버전	슈도 코드, 조건부 출력	아이디어를 자유롭게 표현	안정성 최악, 예측 불가능
1차 개선	자연어 지시, 구조 통일	안정성 대폭 향상	한글 Key 등 비표준 요소 잔존
2차 개선(최종)	영문 Key, 숫자 ID	안정성/개발편의성 최상	-

#### 결론

이번 프롬프트를 개선하는 과정은 "AI에게 얼마나 똑똑하게 지시할까?"가 아니라 \*\*\*"AI가 얼마나 실수 없이 명확하게 알아들을 수 있게 지시할까?"\*\*가 더 중요하다는 것을 보여주었습니다.

복잡하고 창의적인 아이디어에서 출발하더라도, 최종적으로는

**단순하고(Simple), 명확하며(Clear), 일관된(Consistent)**

구조를 갖출 때 가장 안정적이고 실용적인 프롬프트가 탄생한다는 교훈을 얻을 수 있었습니다.

### 검색 기능 개선을 위한 프롬프트\_최종 버전

```
EXTRACT_PARAMS_PROMPT = """
# [Instruction]
You are an expert AI that analyzes a user's book search query and generates a structured JSON object for optimal database search. Follow the steps below precisely.

# [Input Data]
- user_prompt: {user_prompt}
- user_intent: {user_intent}

---
# [Step 1: Classify Search Type]
First, classify the `user_prompt` into one of the 7 types defined below and find its corresponding number ID.

1. **Title**: The user is looking for a specific book title. (e.g., "데미안 찾아줘")
2. **Title + Author**: The user specifies both title and author. (e.g., "헤르만 헤세의 데미안 있어?")
3. **Author**: The user is looking for works by a specific author. (e.g., "헤르만 헤세 작품 목록 보여줘")
4. **Author + Genre**: The user specifies an author and a genre. (e.g., "한강 작가의 소설 작품을 찾아보고 싶어")
5. **Concept + Target Audience**: The user is looking for books for a specific audience. (e.g., "초등학생이 읽을 만한 우주 관련 책")
```

6. **\*\*Concept + Genre\*\***: The user combines a concept with a genre. (e.g., "인공지능을 다루는 SF 소설 추천해 줘")
7. **\*\*Concept + Synopsis\*\***: The user describes the plot or concept of a book. (e.g., "주인공이 갑자기 고양이로 변하는 소설 알아?")

---

# [Step 2: Generate JSON Output]

Based on the classification from Step 1, fill out the following JSON object.

- **\*\*Keys MUST be in English\*\*** as specified in the format below.
- **\*\*If the type ID is 1-4 (Simple Search)\*\***: **EXTRACT** entities directly from the ``user_prompt``.
- **\*\*If the type ID is 5-7 (In-depth Search)\*\***: **INFER** and **SUGGEST** entities based on the ``user_intent``. If the ``user_intent`` strongly suggests a famous book (e.g., "안네의 일기"), you **MUST** fill in the ``title`` and ``author`` fields.
- If a value cannot be found, leave it as an empty string "".

# [Unified Output JSON Format]

Respond with **ONLY** the following JSON object. Do not include any other text or explanations.

```
{
  "searchType": "",      // String: "Simple Search" (for types 1-4) or "In-depth Search" (for types 5-7).
  "queryType": 0,        // Integer: The number ID from 1 to 7 determined in Step 1.
  "title": "",           // Inferred or extracted title of the book.
  "author": "",          // Inferred or extracted author of the book.
  "genre": "",           // Inferred or extracted genre.
  "concept": "",         // Core concept keywords from the user's query.
  "optimizedQuery": ""   // A 3-4 word keyword query for database search.
}
```

# [Example for In-depth Search (Type 7)]

- user\_prompt: "세계 2차대전에 쓰인 소녀의 일기에 관한 책이 뭐지?"
- user\_intent: "세계 2차 대전을 배경으로 한 소녀의 일기 관련 책 정보 탐색. '안네의 일기'일 가능성이 높음."
- Expected Output:

```
{
  "searchType": "In-depth Search",
  "queryType": 7,
  "title": "안네의 일기",
  "author": "안네 프랑크",
  "genre": "에세이",
  "concept": "세계 2차대전, 소녀, 일기",
  "optimizedQuery": "세계 2차대전 소녀 일기"
}
```

# [Example for Simple Search (Type 4)]

- user\_prompt: "한강 작가의 소설 작품을 찾아보고 싶어"
- user\_intent: "한강 작가의 소설 장르 도서 목록 검색"
- Expected Output:

```
{
  "searchType": "Simple Search",
  "queryType": 4,
  "title": "",
  "author": "한강",
  "genre": "소설",
  "concept": "한강, 소설",
  "optimizedQuery": "한강 소설"
}
```

? 사용자 질문: "어린 왕자 책 찾아줘."

[INFO] 지정된 의도(user\_intent)로 파라미터 추출 요청...

- 원본 질문: 어린 왕자 책 찾아줘.
- 하드코딩된 의도: 생텍쥐페리의 '어린 왕자' 도서 검색

--- [ 최종 추출된 검색 파라미터 ] ---

```
{
  "searchType": "In-depth Search",
  "queryType": 1,
  "title": "어린 왕자",
  "author": "생텍쥐페리",
  "genre": "소설",
  "concept": "어린 왕자",
  "optimizedQuery": "어린 왕자 생텍쥐페리"
}
```

(pickle\_reader\_env) ubuntu@H100:~/gs\_content\$ python3 searchReal.py

? 사용자 질문: "조정래 작가의 태백산맥 빌려볼 수 있을까?"

[INFO] 지정된 의도(user\_intent)로 파라미터 추출 요청...

- 원본 질문: 조정래 작가의 태백산맥 빌려볼 수 있을까?
- 하드코딩된 의도: 조정래 작가의 '태백산맥' 도서 정보 및 대출 가능 여부 확인

--- [ 최종 추출된 검색 파라미터 ] ---

```
{
  "searchType": "Simple Search",
  "queryType": 2,
  "title": "태백산맥",
  "author": "조정래",
  "genre": "",
  "concept": "조정래, 태백산맥",
  "optimizedQuery": "조정래 태백산맥"
}
```

(pickle\_reader\_env) ubuntu@H100:~/gs\_content\$ python3 searchReal.py

? 사용자 질문: "베르나르 베르베르가 쓴 책들 좀 알려줘."

[INFO] 지정된 의도(user\_intent)로 파라미터 추출 요청...

- 원본 질문: 베르나르 베르베르가 쓴 책들 좀 알려줘.
- 하드코딩된 의도: 베르나르 베르베르 작가의 전체 작품 목록 조회

--- [ 최종 추출된 검색 파라미터 ] ---

```
{
  "searchType": "Simple Search",
  "queryType": 3,
  "title": "",
  "author": "베르나르 베르베르",
  "genre": "",
  "concept": "베르나르 베르베르",
  "optimizedQuery": "베르나르 베르베르 작품"
}
```

```

=====
(pickle_reader_env) ubuntu@H100:~/gs_content$ python3 searchReal.py
? 사용자 질문: "히가시노 게이고의 미스터리 소설 뭐 있어?"

[INFO] 지정된 의도(user_intent)로 파라미터 추출 요청...
- 원본 질문: 히가시노 게이고의 미스터리 소설 뭐 있어?
- 하드코딩된 의도: 히가시노 게이고 작가의 미스터리 장르 소설 추천 및 목록 검색

--- [ 최종 추출된 검색 파라미터 ] ---
{
  "searchType": "Simple Search",
  "queryType": 4,
  "title": "",
  "author": "히가시노 게이고",
  "genre": "미스터리",
  "concept": "히가시노 게이고, 미스터리 소설",
  "optimizedQuery": "히가시노 게이고 미스터리 소설"
}
-----
=====

(pickle_reader_env) ubuntu@H100:~/gs_content$ python3 searchReal.py
? 사용자 질문: "이제 막 경제 공부 시작하는 직장인데, 어떤 책부터 보면 좋을까?"

[INFO] 지정된 의도(user_intent)로 파라미터 추출 요청...
- 원본 질문: 이제 막 경제 공부 시작하는 직장인데, 어떤 책부터 보면 좋을까?
- 하드코딩된 의도: 경제 초심자 및 직장인을 대상으로 한 입문용 경제 도서 추천

--- [ 최종 추출된 검색 파라미터 ] ---
{
  "searchType": "In-depth Search",
  "queryType": 5,
  "title": "",
  "author": "",
  "genre": "경제",
  "concept": "경제 입문, 직장인",
  "optimizedQuery": "직장인 경제 입문"
}
-----

(pickle_reader_env) ubuntu@H100:~/gs_content$ python3 searchReal.py
? 사용자 질문: "조선시대를 배경으로 하는 로맨스 소설 재밌는 거 없어?"

[INFO] 지정된 의도(user_intent)로 파라미터 추출 요청...
- 원본 질문: 조선시대를 배경으로 하는 로맨스 소설 재밌는 거 없어?
- 하드코딩된 의도: 조선시대를 배경으로 한 역사 로맨스 장르의 인기 도서 추천

--- [ 최종 추출된 검색 파라미터 ] ---
{
  "searchType": "In-depth Search",
  "queryType": 6,
  "title": "",
  "author": "",
  "genre": "로맨스 소설",
  "concept": "조선시대, 역사 로맨스",
  "optimizedQuery": "조선시대 역사 로맨스"
}
-----
=====

(pickle_reader_env) ubuntu@H100:~/gs_content$ python3 searchReal.py
? 사용자 질문: "주인공이 혼자 화성에 남겨져서 감자 키우고 살아남는 내용의 책 제목이 뭐야?"

[INFO] 지정된 의도(user_intent)로 파라미터 추출 요청...
- 원본 질문: 주인공이 혼자 화성에 남겨져서 감자 키우고 살아남는 내용의 책 제목이 뭐야?
- 하드코딩된 의도: 화성에서 조난된 주인공이 생존하는 줄거리의 과학 소설 탐색. 앤디 위어의 '마션'일 가능성이 높음.

--- [ 최종 추출된 검색 파라미터 ] ---
{
  "searchType": "In-depth Search",
  "queryType": 7,
  "title": "마션",
  "author": "앤디 위어",
  "genre": "과학소설",
  "concept": "화성, 조난, 생존, 감자 재배",
  "optimizedQuery": "화성 생존 감자"
}
-----
=====

```

## 최초 프롬프트 (Pseudo-code 형식의 프롬프트)

# [Instruction]

You are an expert **AI** that analyzes a user's book search query, classifies its intent, and generates a structured **JSON** object for optimal database search. Follow the two-stage process below.

```

# [Input Data]
- user_prompt: The user's original, natural language query.
- user_prompt = {user_prompt}
- user_intent: Analyzed user_prompt
- user_intent = {user_intent}

---

# [Stage 1: Classify Search Type]
First, internally determine whether the `user_prompt` corresponds to a "Simple Search" (for specific, known items) or an "In-depth Search" (for discovery and recommendation).

FUNCTION classify_search_type(user_prompt):

    // --- 단순 검색 (Simple Search) 조건 확인 ---
    // 사용자가 찾고자 하는 대상(책, 저자)을 명확히 알고 있는지 검사합니다.

    IF user_prompt contains a specific (Title)
        // 예: "데미안 찾아줘"
    OR user_prompt contains a specific (Title AND Author)
        // 예: "헤르만 헤세의 데미안 있어?"
    OR user_prompt is a request for works by a specific (Author)
        // 예: "헤르만 헤세 작품 목록 보여줘"
    OR user_prompt combines a specific (Author AND Genre)
        // 예: "한강 작가의 소설 작품을 찾아보고 싶어"
    THEN
        RETURN "Simple Search" // Type 0

    // --- 심층 검색 (In-depth Search) ---
    // 위의 조건에 해당하지 않으면, 사용자는 특정 책이 아닌 컨셉 기반의 탐색을 하는 것으로 간주합니다.
    ELSE
        // 예: "초등학생이 읽을 만한 우주 관련 책" (Concept + Target Audience)
        // 예: "인공지능을 다루는 SF 소설 추천해줘" (Concept + Genre)
        // 예: "주인공이 갑자기 고양이로 변하는 소설 알아?" (Concept + Synopsis)
        RETURN "In-depth Search" // Type 1
    END IF

END FUNCTION

---

# [Stage 2: Generate JSON Output based on Classification]
Based on your classification in Stage 1, generate ONLY ONE of the following JSON formats. Do not output anything else.

FUNCTION generate_search_json(user_prompt, user_intent):

    // --- Stage 1 (from previous step) ---
    // First, determine the search type using the previously defined function.
    SET search_type = classify_search_type(user_prompt)

    // --- Stage 2: Generate JSON Output based on Classification ---

    IF search_type IS "Simple Search" THEN
        // For "Simple Search", EXTRACT information directly from the user_prompt.

```

```

// Example user_prompt: "한강 작가의 소설 작품을 찾아보고 싶어"

SET extracted_title = EXTRACT_TITLE_FROM(user_prompt) // ""
SET extracted_author = EXTRACT_AUTHOR_FROM(user_prompt) // "한강"
SET extracted_genre = EXTRACT_GENRE_FROM(user_prompt) // "소설"
SET concept_keywords = COMBINE_KEYWORDS(extracted_author, extracted_genre) // "한강 소설"
SET best_mixture = DETERMINE_MIXTURE_FROM(extracted_title, extracted_author, extracted_genre)
// "Author + Genre"
SET best_search_query = GENERATE_SEARCH_QUERY(concept_keywords) // "한강 소설"

// Construct and return the JSON object using the "Simple Search" format.
RETURN JSON {
  "Title(질문에 포함된 책 제목)": extracted_title,
  "Author(사용자 질문에 포함된 저자명)": extracted_author,
  "Genre(사용자 질문에 포함된 책의 장르)": extracted_genre,
  "Concept(사용자 질문에서 요구하는 핵심적인 컨셉)": concept_keywords,
  "BestMixture(사용자 질문을 기반으로 하는 최고의 검색 조합)": best_mixture,
  "BestSearch(사용자 질문을 기반으로 만들 수 있는 최적의 키토크형 검색어)": best_search_query // 가장 핵심적인
명사 위주로 3-4개의 단어로 조합하라
}

ELSE // search_type IS "In-depth Search"
  // For "In-depth Search", INFER and SUGGEST information based on the user_intent.
  // Example user_intent: "세계 2차 대전에 대한 소녀의 일기 관련 책 정보 탐색", "안네 프랑크의 일기"

  SET inferred_title = INFER_FAMOUS_TITLE_FROM(user_intent) // "안네의 일기"
  SET inferred_author = INFER_FAMOUS_AUTHOR_FROM(user_intent) // "안네 프랑크"
  SET inferred_genre = INFER_GENRE_FROM(user_intent) // "에세이"
  SET inferred_concept = INFER_CONCEPTS_FROM(user_intent) // "세계 2차대전, 소녀, 일기"
  SET best_mixture = DETERMINE_MIXTURE_FROM(user_intent) // "Concept + Synopsis"
  SET best_search_query = GENERATE_SEARCH_QUERY(inferred_concept) // "세계 2차대전 사춘기 소녀의 일
기"

  // Construct and return the JSON object using the "In-depth Search" format.
  RETURN JSON {
    "Title(user_intent를 통해 추론할 수 있는 세계의(혹은 한국의) 유명 책 제목)": inferred_title,
    "Author(user_intent를 통해 추론할 수 있는 세계의(혹은 한국의) 유명 저자명)": inferred_author,
    "Genre(user_intent를 통해 추론할 수 있는 책의 장르)": inferred_genre,
    "Concept(user_intent를 통해 추론할 수 있는 핵심적인 컨셉)": inferred_concept,
    "BestMixture(user_intent를 기반으로 하는 최고의 검색 조합)": best_mixture,
    "BestSearch(user_intent를 기반으로 만들 수 있는 최적의 키토크형 검색어)": best_search_query // 가장 핵심
적인 명사 위주로 3-4개의 단어로 조합하라
  }
END IF

END FUNCTION

```

## 1차 개선 프롬프트

```

# [Instruction]
You are an expert AI that analyzes a user's book search query, classifies its intent, and generates a structured JSON object for optimal database search. Follow the two-stage process below.

# [Input Data]

```

- user\_prompt: The user's original, natural language query.
- user\_prompt = {user\_prompt}
- user\_intent: Analyzed user\_prompt
- user\_intent = {User\_intent}

---

#### # [Stage 1: Classify Search Type]

First, internally determine whether the `user\_prompt` corresponds to a "Simple Search" (for specific, known items) or an "In-depth Search" (for discovery and recommendation).

#### ## Simple Search (Type 0)

The user is looking for a specific item. This includes queries about:

- **Title**: e.g., "데미안 찾아줘"
- **Title + Author**: e.g., "헤르만 헤세의 데미안 있어?"
- **Author**: e.g., "헤르만 헤세 작품 목록 보여줘"
- **Author + Genre**: e.g., "한강 작가의 소설 작품을 찾아보고 싶어"

#### ## In-depth Search (Type 1)

The user is exploring concepts or needs recommendations. This includes queries about:

- **Concept + Target Audience**: e.g., "초등학생이 읽을 만한 우주 관련 책"
- **Concept + Genre**: e.g., "인공지능을 다루는 SF 소설 추천해줘"
- **Concept + Synopsis**: e.g., "주인공이 갑자기 고양이로 변하는 소설 알아?"

---

#### # [Stage 2: Generate JSON Output based on Classification]

Based on your classification in Stage 1, generate **ONLY ONE** of the following JSON formats. Do not output anything else.

#### ## If "Simple Search" (Type 0):

Generate a JSON object by **EXTRACTING** information directly from the prompt.

- **Keys**: Must be in Korean as specified below.
- **Values**: If a value cannot be extracted, leave the string empty "".

#### ### Output Format (Simple Search):

```
{
  "Title(질문에 포함된 책 제목)": "",
  "Author(사용자 질문에 포함된 저자명)": "",
  "Genre(사용자 질문에 포함된 책의 장르)": "",
  "Concept(사용자 질문에서 요구하는 핵심적인 컨셉)": "",
  "BestMixture(사용자 질문을 기반으로 하는 최고의 검색 조합)": "",
  "BestSearch(사용자 질문을 기반으로 만들 수 있는 최적의 키토크형 검색어)": ""
}
```

#### ### Example (Simple Search):

- **user\_prompt**: "한강 작가의 소설 작품을 찾아보고 싶어"
- **Output**:
 

```
{
        "Title(질문에 포함된 책 제목)": "",
        "Author(사용자 질문에 포함된 저자명)": "한강",
        "Genre(사용자 질문에 포함된 책의 장르)": "소설",
        "Concept(사용자 질문에서 요구하는 핵심적인 컨셉)": "한강 소설",
        "BestMixture(사용자 질문을 기반으로 하는 최고의 검색 조합)": "Author + Genre",
        "BestSearch(사용자 질문을 기반으로 만들 수 있는 최적의 키토크형 검색어)": "한강 소설"
      }
```



```
}
```

## If "In-depth Search" (Type 1):

Generate a **JSON** object by **INFERRING** and **SUGGESTING** information based on the prompt's context. For "Title" and "Author", you can recommend famous world or Korean literature if applicable.

- **\*\*Keys\*\***: Must be in Korean as specified below.

- **\*\*Values\*\***: If a value cannot be inferred, leave the string empty "".

### Output Format (In-depth Search):

```
{
  "Title(사용자 질문을 통해 추론할 수 있는 세계의(혹은 한국의) 유명 책 제목)": "",
  "Author(사용자 질문을 통해 추론할 수 있는 세계의(혹은 한국의) 유명 저자명)": "",
  "Genre(사용자 질문을 통해 추론할 수 있는 책의 장르)": "",
  "Concept(사용자 질문을 통해 추론할 수 있는 핵심적인 컨셉)": "",
  "BestMixture(사용자 질문을 기반으로하는 최고의 검색 조합)": "",
  "BestSearch(사용자 질문을 기반으로 만들 수 있는 최적의 키토크형 검색어)": ""
}
```

### Example (In-depth Search):

- **\*\*user\_prompt\*\***: "세계 2차대전을 배경으로 한 소녀의 일기 책이 뭐지?"

- **\*\*Output\*\***:

```
{
  "Title(사용자 질문을 통해 추론할 수 있는 세계의(혹은 한국의) 유명 책 제목)": "안네의 일기",
  "Author(사용자 질문을 통해 추론할 수 있는 세계의(혹은 한국의) 유명 저자명)": "안네 프랑크",
  "Genre(사용자 질문을 통해 추론할 수 있는 책의 장르)": "에세이",
  "Concept(사용자 질문을 통해 추론할 수 있는 핵심적인 컨셉)": "세계 2차대전, 소녀, 일기",
  "BestMixture(사용자 질문을 기반으로하는 최고의 검색 조합)": "Concept + Synopsis",
  "BestSearch(사용자 질문을 기반으로 만들 수 있는 최적의 키토크형 검색어)": "세계 2차대전 일기"
}
```