

책을 출판하려고 한다. 책의 뒷부분의 인덱스를 뽑으려고 하는데, 일반적으로 인덱스는 어떤 것이 좋을까?

책을 출판하려고 하시는군요, 멋진 계획입니다! 인덱스(index)는 책의 뒷부분에서 독자가 주요 주제나 키워드를 빠르게 찾을 수 있도록 도와주는 중요한 도구인데요. 어떤 인덱스가 좋을지는 책의 성격과 독자층에 따라 조금 달라질 수 있습니다. 일반적으로 잘 만들어진 인덱스의 특징과 고려할 점을 말씀드릴게요.

- 주요 키워드와 개념 중심:** 인덱스는 책에서 다루는 핵심 주제, 인물, 사건, 용어 등을 포함해야 합니다. 예를 들어, 소설이라면 주요 캐릭터 이름이나 사건이, 논픽션이라면 개념이나 연구 주제가 들어가야겠죠. 독자가 "이건 어디서 봤더라?" 하고 찾을 만한 단어를 떠올려보세요.
- 간결하면서도 구체적:** 너무 추상적이거나 모호한 단어(예: "사랑", "행복")보다는 구체적인 항목(예: "사랑의 상징, 45쪽" 또는 "행복의 심리학, 120쪽")이 좋습니다. 페이지 번호도 꼭 함께 넣어주세요.
- 알파벳순 또는 주제별 정리:** 일반적으로 인덱스는 알파벳순으로 배열하는 게 가장 보편적이고 찾기 쉽습니다. 하지만 책이 학술적이거나 주제 중심이라면, 큰 주제 아래 하위 항목을 두는 계층적 구조(예: "인공지능 - 역사, - 윤리")도 유용할 수 있어요.
- 독자 맞춤:** 책의 타겟 독자가 전문가라면 더 기술적이고 세부적인 항목을, 일반 대중이라면 좀 더 직관적이고 단순한 항목을 추천드립니다.
- 적당한 분량:** 페이지 수가 200쪽인 책이라면 인덱스가 2~5쪽 정도 되는 게 적당합니다. 너무 길면 오히려 부담스럽고, 너무 짧으면 유용성이 떨어질 수 있어요.

예를 들어, 제가 지금까지 나눈 대화에서 인덱스를 만든다면:

- 개념, 1쪽
- 키워드, 1쪽
- 독자층, 1쪽
- 인덱스 구조, 1쪽

이런 식으로 시작할 수 있겠네요. 책 주제가 구체적으로 뭔지 알려주시면 더 맞춤형으로 제안드릴 수도 있을 것 같아요. 주제가 뭔가요?

책은 제목은 아래와 같고, 다른 목차는 아래와 같다.

[제목] 제대로 시작하는 챗GPT와 AI 활용 with 파이썬

[목차]

PART 1 오픈AI와 챗GPT 제대로 이해하기

1장 생성형 AI 서비스

1.1 생성형 AI 개요

..... 25

1.1.1 생성형 AI의 활용

..... 25

1.1.2 생성형 AI 모델의 아키텍처

..... 26

1.2 대표적인 생성형 AI 서비스

..... 26

2장 오픈AI와 챗GPT

2.1 오픈AI와 챗GPT 개요

..... 29

2.2 GPT 모델 및 사용 사례

..... 33

3장 개발 사전 준비

3.1 구글 코랩 사용 방법

..... 37

3.2 API 개요



.....	40
3.3 오픈AI API 키 얻기	41
3.4 오픈AI API 키 활용 실습	45
3.4.1 openai 라이브러리 설치	45
3.4.2 chatgpt.env 파일 작성	46
3.5 API 사용량 및 요금 확인	56
14	
4장 오픈AI API 활용 첫걸음	
4.1 GPT-4o mini로 시작하는 대화 생성	59
4.2 system, user, assistant 역할의 기본 이해	63
4.2.1 system 역할	63
4.2.2 user 역할	65
4.2.3 assistant 역할	66
4.3 퓨샷 학습 소개	66
4.4 GPT 모델을 활용한 맞춤형 출력 생성	68
4.5 대화 출력 길이 제어	72
4.6 AI 응답 제어	76
5장 오픈AI API 모델 탐색하기	
5.1 사용 가능 모델 확인 및 소개	81
5.1.1 GPT 시리즈: 텍스트 생성 및 이해	85
5.1.2 DALL-E 시리즈: 이미지 생성	85
5.1.3 워스퍼 시리즈: 음성 인식	85
5.1.4 TTS: 텍스트의 음성 변환	85
5.1.5 텍스트 임베딩 모델	86
5.2 사용할 모델 선택	86
5.3 오픈AI 모델 시리즈	87
5.3.1 GPT-4o 시리즈	87
5.3.2 GPT-4o mini 시리즈	88
5.3.3 o1 및 o1-mini 시리즈	89
5.3.4 GPT-4 Turbo 및 GPT-4 시리즈	90
5.3.5 DALL-E 시리즈	90

5.3.6 TTS 시리즈	91
.....
목차	
15	
5.3.7 워스퍼 시리즈	91
.....
5.3.8 임베딩 모델	92
.....
5.4 오픈AI 모델 가격	92
.....
5.4.1 GPT-4o 모델 가격	93
.....
5.4.2 GPT-4o mini	94
.....
5.4.3 o1 및 o1-mini 가격	95
.....
5.4.4 임베딩 모델	96
.....
5.4.5 파인 투닝 모델	96
.....
5.4.6 실시간 API	97
.....
5.4.7 어시스턴트 API	98
.....
5.4.8 이미지 모델	98
.....
5.4.9 오디오 모델	99
.....
5.4.10 기타 모델	99
.....
5.5 오픈AI 모델의 발전과 선택 가이드	100
.....
PART 2 오픈AI GPT 대화 생성 실전 가이드	
6장 대화 생성의 고급 기술	
6.1 창의성 조절 기법	104
.....
6.1.1 temperature	104
.....
6.1.2 top_p 샘플링	107
.....
6.1.3 temperature와 top_p의 차이점	110
.....
6.2 페널티로 독창성과 다양성 확보하기	111
.....
6.3 n 파라미터로 여러 응답 생성하기	115
.....
6.4 실시간 스트리밍으로 즉시 출력하기	117
.....
16	
7장 GPT 모델의 성능을 극대화하는 프롬프트 엔지니어링 기법	
7.1 프롬프트 엔지니어링 개요	119
.....
7.2 퓨샷 학습: 키워드 추출 성능 높이기	121
.....
7.3 프롬프트 체이닝: 트윗 생성 최적화하기	127
.....
7.4 일반 지식 프롬프팅: 올드 스쿨 랩 가사 완성하기	133
.....

7.5 눈백 세우기: AI의 이해도 높이기	138
7.6 동적 토큰 제어: AI 응답 길이 유연하게 조절하기	147
7.7 프롬프트 템플릿 최적화: 리눅스 학습용 CLI 어시스턴트 개발하기	
7.8 다음으로 나아가기	164
PART 3 음성 및 이미지 인식: 워스퍼와 CLIP의 실전 활용	
8장 워스퍼 음성 인식 기초	
8.1 워스퍼 개요	
8.2 워스퍼 설치 방법 및 기본 예제	169
8.3 파이썬에서 워스퍼 SDK 사용하기	171
9장 워스퍼 텍스트 변환 최적화 기법	
9.1 오픈AI 음성-텍스트 API 사용하기	
9.1.1 전사 API	177
9.1.2 번역 API	177
9.2 워스퍼 텍스트 변환 개선하기	
9.2.1 오디오 정리하기	180
9.2.2 프롬프트 파라미터 사용하기	180
9.2.3 전사된 텍스트 후처리하기	181
10장 오픈AI TTS를 활용한 음성 변환	
10.1 오픈 AI TTS 소개	
10.2 오픈 AI TTS 사용 방법	185
11장 오픈AI CLIP을 사용한 이미지 분류	
11.1 CLIP 소개	
11.2 CLIP 사용 방법	191
11.2.1 필수 라이브러리 설치	191
11.2.2 CLIP으로 제로샷 분류하기	193
11.3 이미지를 텍스트로 변환하기	199
PART 4 생성형 AI를 활용한 이미지 제작과 편집 기술	
12장 DALL-E로 이미지 생성하기	
12.1 DALL-E 소개	
12.2 프롬프트 기반의 이미지 생성 기본 예제	203
12.3 다른 크기의 이미지 생성하기	205
12.4 여러 이미지 생성하기	207

.....	208
12.5 이미지 프롬프트 개선 방법	
.....	210
12.5.1 작가 모방	210
.....	210
12.5.2 예술 스타일 모방	213
.....	213
12.5.3 분위기와 감정	215
.....	215
18	
12.5.4 색상	219
.....	219
12.5.5 해상도	221
.....	221
12.5.6 각도와 위치	221
.....	221
12.5.7 렌즈 종류와 촬영 기법	223
.....	223
12.5.8 조명	226
.....	226
12.5.9 이미지 필터	227
.....	227
12.6 프롬프트 조합으로 다양한 랜덤 이미지 만들기	
.....	228
13장 DALL-E로 이미지 편집하기	
13.1 이미지 편집 예제	
.....	236
14장 다른 이미지에서 영감 얻기	
14.1 이미지 변형 방법	
.....	241
14.2 이미지 변형 사용 사례	
.....	246
PART 5 임베딩: 복잡한 데이터를 쉽게 이해하는 방법	
15장 임베딩 소개	
15.1 임베딩의 의미	
.....	251
15.2 임베딩 사용 사례	
.....	252
15.2.1 테슬라: 자율주행차 학습 효율성 향상	
.....	253
15.2.2 Kalender AI: 세일즈 타겟팅 정확도 향상	
.....	253
15.2.3 노션: 검색 기능 향상	
.....	254
15.2.4 DALL-E 3: 텍스트의 이미지 변환 능력 향상	
.....	254
목차	
19	
16장 텍스트 임베딩 활용 방법	
16.1 텍스트 임베딩의 핵심 이해하기	
.....	255
16.2 다중 입력 사용 예제	
.....	259
16.3 사용자의 의도를 읽어내는 시맨틱 검색	
.....	259
16.4 코사인 유사도 쉽게 이해하기	
.....	260
16.5 오픈AI 텍스트 임베딩 사용 예제	

16.6 임베딩의 내부 작동 원리	200
.....	277
17장 고급 임베딩 예제	
17.1 선호하는 커피 예측하기	279
.....	296
17.2 퍼지 검색으로 더 유연한 커피 검색 구현하기	
.....	303
17.3 제로샷 분류기로 뉴스 기사 분류하기	
.....	308
17.4 제로샷 분류기의 정밀도 평가	
.....	316
PART 6 파인 투닝과 모델의 실전 활용	
18장 퓨샷 학습 이해하기	
18.1 퓨샷 학습의 기본 개념과 정의	
.....	319
18.2 파인 투닝으로 퓨샷 학습 성능 향상하기	
.....	320
19장 파인 투닝의 이해 및 구현	
19.1 파인 투닝 실제 적용	
.....	323
20	
19.2 파인 투닝 최적화	
.....	331
19.2.1 모델 선택	
.....	331
19.2.2 데이터셋 검증	
.....	331
19.2.3 토큰 제한	
.....	340
19.2.4 데이터셋 크기	
.....	340
19.2.5 훈련 테스트 및 개선: 하이퍼파라미터 조정	
.....	340
19.2.6 예상 비용 고려하기	
.....	344
19.2.7 데이터셋 품질 높이기	
.....	345
19.2.8 파인 투닝과 다른 기술 결합하기	
.....	346
19.2.9 실험과 학습 지속하기	
.....	346
19.2.10 모델 테스트하기	
.....	347
19.2.11 검증 세트 사용하기	
.....	347
19.2.12 결과 분석하기	
.....	348
20장 파인 투닝 고급 예제: 정신 건강 코치	
20.1 예제에서 사용한 데이터셋	
.....	353
20.2 데이터셋 준비하기	
.....	355
20.3 실제 애플리케이션에서 발생할 수 있는 한계	
.....	366
21장 기억력 및 문맥 문제와 해결책	
21.1 GPT의 기억력 및 문맥 문제	

.....	369
21.2 문맥 유지의 중요성	370
.....	370
21.3 대화 문맥 유지의 해법: 대화 이력 추가	373
.....	373
21.4 대화 이력 관리의 문제점	377
.....	377
21.5 후입선출 방식으로 최신 문맥 유지하기	378
.....	378
21.6 후입선출 메모리의 한계	385
.....	385
21.7 선택적 문맥으로 최적의 대화 문맥 선택하기	385
.....	385
목차	
21	
PART 7 부록	
부록 A 챗GPT 및 클로드 실습	
A.1 코딩 공부에 활용하기	399
.....	399
A.2 유튜브 영상에서 텍스트를 추출하고 요약하기	405
.....	405
A.3 간단한 고객 데이터 분석 수행하기	411
.....	411
A.4 맞춤형 GPT 소개 및 구현	414
.....	414
부록 B 오픈AI 플레이그라운드 실습	
B.1 오픈AI 플레이그라운드 소개	429
.....	429
B.2 취업 진로 상담하기	434
.....	434
B.3 이모지 만들기	437
.....	437
B.4 웹 페이지 작성하기	441
.....	441
부록 C 건강 상담 챗봇 구현하기	
C.1 스트림릿으로 간단한 웹 앱 서비스 구현하기	445
.....	445
C.2 건강 상담 챗봇 웹 서비스 구현	452
.....	452
부록 D 작가 스타일 이미지 생성 앱	
D.1 풍경화 이미지 생성하기	460
.....	460
D.2 특정 작가 스타일의 이미지 생성하기	462
.....	462
D.3 스트림릿 앱으로 작가별 이미지 생성하기	466
.....	466
22	
부록 E RAG로 노트북 추천받기	
E.1 RAG 소개	471
.....	471
E.2 RAG 실습	474
.....	474
부록 F 오토젠을 활용한 AI 에이전트 이해하기	
F.1 오토젠 소개	485
.....	485
F.2 멀티 에이전트 활용 실습	486
.....	486
찾아보기	

책 제목과 목차를 보니, "제대로 시작하는 ChatGPT와 AI 활용 with 파이썬"은 초보자부터 중급 개발자까지를 타겟으로 한 실용적인 AI 활용 가이드북이네요. 오픈AI의 다양한 기술(GPT, DALL-E, 워스퍼, CLIP 등)을 파이썬으로 다루는 방법을 체계적으로 설명하는 만큼, 인덱스는 독자가 기술적 개념, 도구, 실습 항목을 빠르게 찾을 수 있도록 설계하는 게 중요할 것 같습니다. 아래는 이 책에 적합한 인덱스 구성에 대한 제안이에요.

인덱스 구성 제안

1. 핵심 키워드 중심:

책이 기술적이고 실습 중심이니, 주요 개념, 모델 이름, 프로그래밍 용어, API 관련 항목을 중심으로 뽑아야 합니다. 너무 일반적인 단어(예: "AI")보다는 구체적인 용어(예: "GPT-4o", "프롬프트 엔지니어링")를 우선순위로 두세요.

2. 알파벳 순 배열:

한국어와 영어가 섞여 있으니, 한글은 가나다순, 영어는 알파벳순으로 통합하거나, 한글과 영어를 별도 섹션으로 나눠도 괜찮습니다. 다만 독자가 주로 파이썬 개발자일 가능성이 높으니 영어 용어를 많이 참조할 거라 예상돼요.

3. 하위 항목 활용:

같은 키워드가 여러 맥락에서 등장하면 하위 항목으로 구분해 주면 찾기 쉬워요. 예: "임베딩" 아래 "텍스트 임베딩", "사용 사례" 등을 추가.

4. 페이지 번호 필수:

실습 위주 책이니 정확한 페이지로 바로 연결되도록 페이지 번호를 꼼꼼히 넣어주세요.

5. 추천 분량:

500쪽 분량이라면 인덱스는 5~10쪽(약 200~400개 항목)이 적당할 거예요. 너무 길면 부담스럽고, 너무 짧으면 실용성이 떨어질 수 있으니 균형을 맞추세요.

예시 인덱스 (일부抜削)

아래는 목차를 바탕으로 인덱스 항목을 샘플로 만들어 본 거예요. 책 전체를 반영한 건 아니고, 방향성을 보여드리기 위한 예시입니다.

text

Wrap Copy

가나다순:

- API
 - 오픈AI API 키 얻기, 41
 - 오픈AI API 활용 실습, 45
 - 실시간 API, 97
 - 어시스턴트 API, 98
- GPT 모델
 - GPT-4o, 87
 - GPT-4o mini, 88, 59 (대화 생성)
 - GPT-4 Turbo, 90
 - 사용 사례, 33
- 구글 코랩, 37
- 대화 생성
 - 고급 기술, 104
 - 출력 길이 제어, 72
 - 실시간 스트리밍, 117
- 오픈AI
 - 개요, 29
 - 모델 가격, 92
 - 플레이그라운드, 429
- 워스퍼
 - 음성 인식 기초, 169

- 임베딩
 - 소개, 251
 - 텍스트 임베딩 활용, 255
 - 코사인 유사도, 260
- 파일썬
 - openai 라이브러리 설치, 45
 - 위스퍼 SDK 사용, 173
- 프롬프트 엔지니어링
 - 개요, 119
 - 퓨샷 학습, 121
 - 프롬프트 체이닝, 127

알파벳순:

- CLIP
 - 소개, 191
 - 제로샷 분류, 193
- DALL·E
 - 이미지 생성, 203
 - 이미지 편집, 236
- temperature, 104
- top_p 샘플링, 107
- TTS
 - 소개, 185
 - 사용 방법, 186

추가 팁

- **모델 이름 강조:** GPT-4o, DALL-E, 위스퍼 같은 오픈AI의 주요 모델은 독자가 자주 찾을 가능성이 높으니 반드시 포함하세요. 각 모델별로 세부 항목(예: "GPT-4o - 가격", "GPT-4o - 대화 생성")을 추가하면 더 좋아요.
- **실습 관련 용어:** "설치", "실습", "예제" 같은 단어는 실습을 따라 하고 싶은 독자에게 유용합니다.
- **부록 포함:** 부록에 실습이나 특정 주제가 많으니 "챗GPT 실습", "RAG 소개" 같은 항목도 넣으면 좋을 것 같아요.
- **중복 관리:** "GPT"처럼 여러 장에 걸쳐 나오는 용어는 페이지 번호를 모두 나열하거나, 주요 페이지만 선별해서 넣으세요.

책 주제가 기술적이고 실습 중심이라 인덱스가 독자의 학습 효율을 크게 좌우할 거예요. 제가 제시한 방향이 마음에 드시면, 목차 전체를 기반으로 더 구체적인 인덱스를 만들어 드릴 수도 있습니다. 어떻게 진행하고 싶으신가요? 아니면 특정 파트에 더 집중해서 제안해 드릴까요?

목차 전체를 기반으로 더 구체적인 인덱스 만들어보련.

알겠습니다! "제대로 시작하는 챗GPT와 AI 활용 with 파일썬"의 전체 목차를 기반으로 구체적인 인덱스를 만들어 보겠습니다. 책이 기술적이고 실습 중심이니, 주요 개념, 모델, 도구, 프로그래밍 용어, 실습 항목을 중심으로 구성하되, 독자가 빠르게 찾을 수 있도록 간결하고 구체적으로 정리하겠습니다. 한글과 영어 용어가 섞여 있으니 가나다순과 알파벳순을 섞어 통합적으로 배열하고, 페이지 번호도 정확히 반영하겠습니다. 분량은 약 200~300개 항목으로, 57쪽 정도를 목표로 하겠습니다.

인덱스

text

Wrap Copy

가나다순:

- API
 - 오픈AI API 개요, 40
 - 오픈AI API 키 얻기, 41
 - 오픈AI API 활용 실습, 45
 - openai 라이브러리 설치, 45
 - chatgpt.env 파일 작성, 46
 - API 사용량 및 요금 확인, 56
 - 실시간 API, 97
 - 어시스턴트 API, 98
 - 음성-텍스트 API, 177
 - 전사 API, 177
 - 번역 API, 179
- CLIP
 - 소개, 191
 - 사용 방법, 191
 - 필수 라이브러리 설치, 192
 - 제로샷 분류, 193
 - 이미지를 텍스트로 변환, 199
- DALL·E
 - 소개, 203
 - 이미지 생성, 205
 - 기본 예제, 205
 - 다른 크기 생성, 207
 - 여러 이미지 생성, 208
 - 프롬프트 개선, 210
 - 작가 모방, 210
 - 예술 스타일 모방, 213
 - 분위기와 감정, 215
 - 색상, 219
 - 해상도, 221
 - 각도와 위치, 221
 - 렌즈와 촬영 기법, 223
 - 조명, 226
 - 이미지 필터, 227
 - 프롬프트 조합, 228
 - 이미지 편집, 236
 - 이미지 변형, 241
 - 사용 사례, 246
- GPT 모델
 - 개요, 33
 - 사용 사례, 33
 - GPT-4o, 87
 - 대화 생성, 59
 - 가격, 93
 - GPT-4o mini, 88
 - 대화 생성, 59
 - 가격, 94
 - GPT-4 Turbo, 90
 - o1 및 o1-mini, 89
 - 가격, 95
 - 맞춤형 출력 생성, 68
 - 대화 출력 길이 제어, 72
 - 응답 제어, 76
 - 기억력 및 문맥 문제, 369
- 구글 코랩
 - 사용 방법, 37
- 대화 생성
 - GPT-4o mini로 시작, 59
 - system, user, assistant 역할, 63
 - system 역할, 63
 - user 역할, 65
 - assistant 역할, 66
 - 고급 기술, 104
 - temperature, 104
 - top_p 샘플링, 107
 - 차이점, 110
 - 패널티, 111
 - n 파라미터, 115
 - 실시간 스트리밍, 117
- 데이터셋
 - 파일 튜닝 데이터셋 준비, 355
 - 검증, 331
 - 품질 높이기, 345
- 캡 가사 완성, 133
- 리눅스 학습용 CLI 어시스턴트, 153

- 끝낸 GPT
 - 소개 및 구현, 414
- 문맥 유지
 - 중요성, 370
 - 대화 이력 추가, 373
 - 후입선출 방식, 378
 - 선택적 문맥, 385
- 부록
 - 챗GPT 및 클로드 실습, 399
 - 코딩 공부, 399
 - 유튜브 텍스트 추출, 405
 - 고객 데이터 분석, 411
 - 오픈AI 플레이그라운드, 429
 - 취업 전로 상담, 434
 - 이모지 만들기, 437
 - 웹 페이지 작성, 441
 - 건강 상담 챗봇, 445
 - 스트림릿 웹 앱, 445
 - 구현, 452
 - 작가 스타일 이미지 생성 앱, 460
 - 풍경화 생성, 460
 - 특정 작가 스타일, 462
 - 스트림릿 앱, 466
 - RAG 실습, 474
 - 오토젠 실습, 486
- 생성형 AI
 - 개요, 25
 - 활용, 25
 - 아키텍처, 26
 - 대표적인 서비스, 26
- 스트림릿
 - 웹 앱 서비스 구현, 445
 - 건강 상담 챗봇, 452
 - 작가 스타일 이미지 생성, 466
- 오디오 모델, 99
- 오토젠
 - 소개, 485
 - 멀티 에이전트 활용, 486
- 오픈AI
 - 개요, 29
 - API 키 활용, 45
 - 모델 탐색, 81
 - 사용 가능 모델, 81
 - GPT 시리즈, 85
 - DALL·E 시리즈, 85
 - 위스퍼 시리즈, 85
 - TTS, 85
 - 텍스트 임베딩, 86
 - 모델 가격, 92
 - 플레이그라운드, 429
- 위스퍼
 - 개요, 169
 - 설치 방법, 171
 - 파이썬 SDK 사용, 173
 - 텍스트 변환 최적화, 177
 - 오디오 정리, 180
 - 프롬프트 파라미터, 181
 - 텍스트 후처리, 183
- 음성 변환
 - TTS 소개, 185
 - TTS 사용 방법, 186
- 이미지 모델, 98
- 임베딩
 - 소개, 251
 - 사용 사례, 252
 - 테슬라 자율주행, 253
 - Kalendar AI, 253
 - 노선 검색, 254
 - DALL·E 3, 254
 - 텍스트 임베딩, 255
 - 디중 입력 예제, 259
 - 시맨틱 검색, 259
 - 코사인 유사도, 260
 - 사용 예제, 265
 - 내부 작동 원리, 277
 - 고급 예제

- 커피 예측, 279
- 퍼지 검색, 296
- 제로샷 분류기, 303
- 자율주행차 학습, 253
- 제로샷 분류
 - CLIP으로 구현, 193
 - 뉴스 기사 분류, 303
 - 정밀도 평가, 308
- 정신 건강 코치
 - 파인 튜닝 예제, 353
- 챗GPT
 - 개요, 29
 - 실습, 399
- 코사인 유사도, 260
- 퓨샷 학습
 - 소개, 66
 - 키워드 추출, 121
 - 기본 개념, 319
 - 파인 튜닝으로 성능 향상, 320
- 파일
 - openai 라이브러리 설치, 45
 - 워스파 SDK 사용, 173
- 파인 튜닝
 - 실제 적용, 323
 - 최적화, 331
 - 모델 선택, 331
 - 데이터셋 검증, 331
 - 토큰 제한, 340
 - 하이퍼파라미터 조정, 340
 - 정신 건강 코치 예제, 353
- 프롬프트 엔지니어링
 - 개요, 119
 - 퓨샷 학습, 121
 - 프롬프트 체이닝, 127
 - 일반 지식 프롬프팅, 133
 - 문맥 채우기, 138
 - 동적 토큰 제어, 147
 - 템플릿 최적화, 153

알파벳순:

- **assistant** 역할, 66
- **chatgpt.env** 파일, 46
- **n** 파라미터, 115
- **RAG**
 - 소개, 471
 - 실습, 474
- **system** 역할, 63
- **temperature**, 104
- **top_p** 샘플링, 107
- **TTS**
 - 소개, 185
 - 사용 방법, 186
- **user** 역할, 65

특징 및 보완 사항

- 구체성:** "GPT-4o" 같은 모델 이름은 단순히 페이지 나열보다는 맥락별로 세분화(예: "대화 생성", "가격")해서 독자가 원하는 정보를 바로 찾도록 했습니다.
- 실습 중심:** "설치", "예제", "실습" 같은 항목은 개발자 독자가 실습을 따라 하기 쉽게 포함했습니다.
- 부록 반영:** 부록의 실용적인 주제(챗GPT 실습, RAG, 오토전 등)를 추가해 실무 활용성을 높였어요.
- 항목 수:** 약 250개 정도로, 500쪽 분량에 적절한 5~6쪽 분량의 인덱스가 될 거예요.

만약 특정 부분을 더 보강하거나 줄이고 싶으시면 말씀해주세요. 예를 들어, 실습 항목을 더 강조하거나, 모델별 세부 항목을 늘릴 수도 있습니다. 어떻게 생각하시나요?

How can Grok help?



DeepSearch



Think

Grok 3 ▾

