

# 생성형AI-랭체인(LLM)활용 백엔드 (풀스택) 엔지니어(자바,파이썬) 양성

## ▼ 프로그래밍 기초(자바) [24/72]

### ▼ 자바 개발 환경 설정 및 기본 프로젝트 실습

#### ▼ IntelliJ 설치

<https://www.jetbrains.com/ko-kr/idea/download/?section=windows>

#### ▼ Visual Studio Code

<https://code.visualstudio.com/>

#### ▼ Java 기본 문법

#### ▼ 배열 및 문자열 기본

#### ▼ 객체지향 프로그래밍

#### ▼ 상속과 다형성

#### ▼ 인터페이스와 추상 클래스

#### ▼ 예외 처리

#### ▼ Collections

#### ▼ 파일 입출력

#### ▼ 멀티스레드 프로그래밍

#### ▼ 람다 표현식 및 스트림 API

#### ▼ 스트림 API 고급 기능

#### ▼ Java 11 새로운 기능

#### ▼ 데이터베이스 연동

#### ▼ GUI 프로그래밍

#### ▼ 네트워크 프로그래밍

#### ▼ 디자인 패턴

#### ▼ 유닛 테스트와 디버깅

#### ▼ 빌드 도구 및 프로젝트 관리

## ▼ 프로그래밍 기초(파이썬) [16/56]

- ▼ 파이썬 개발 환경 설정 및 기본 프로젝트 실습
- ▼ 기본 문법
- ▼ 데이터 구조
- ▼ 함수
- ▼ 문자열 처리
- ▼ 파일 처리
- ▼ 예외 처리
- ▼ 객체지향 프로그래밍
- ▼ 모듈과 패키지
- ▼ 심화 데이터 구조
- ▼ 반복 및 제너레이터
- ▼ 데코레이터와 컨텍스트 매니저
- ▼ 데이터 시각화
- ▼ 데이터 분석
- ▼ NumPy와 고속 연산
- ▼ 알고리즘
- ▼ 웹 크롤링
- ▼ API 활용
- ▼ GUI 프로그래밍
- ▼ 게임 프로그래밍
- ▼ 데이터베이스
- ▼ 네트워크 프로그래밍
- ▼ 멀티스레딩과 멀티프로세싱
- ▼ 테스트와 디버깅
- ▼ 자동화 스크립트 작성
- ▼ 데이터 과학 도구
- ▼ 클라우드와 배포

## ▼ 웹 개발 기초 [16/88]

### ▼ HTML5 및 CSS3 기초 / 반응형 웹 디자인

- ▼ HTML과 웹의 기본 개념 이해
- ▼ HTML 기본 요소 활용
- ▼ HTML 폼의 기초
- ▼ HTML 폼 요소 심화
- ▼ 멀티미디어 요소 다루기
- ▼ HTML 표 작성
- ▼ HTML5 시맨틱 태그 이해
- ▼ CSS 기본 개념 및 문법
- ▼ CSS 박스 모델
- ▼ CSS 레이아웃 기초
- ▼ CSS Flexbox 기초
- ▼ CSS Flexbox 심화
- ▼ CSS Grid 기초
- ▼ CSS Grid 심화
- ▼ CSS 애니메이션 기초
- ▼ 반응형 웹 디자인 기초
- ▼ HTML 및 CSS 통합 복습

### ▼ JavaScript 기본 문법 / DOM 조작 및 이벤트 처리

- ▼ 기본 문법
- ▼ 함수 기초
- ▼ 배열과 객체
- ▼ DOM 조작
- ▼ 이벤트 심화
- ▼ ES6+ 문법
- ▼ 비동기 처리
- ▼ 브라우저 내장 객체

- ▼ 웹스토리지
- ▼ 모듈과 패키지
- ▼ 에러 처리
- ▼ 디버깅과 최적화
- ▼ 데이터 처리와 JSON
- ▼ Form과 Validation
- ▼ 웹 애니메이션
- ▼ Canvas와 그래픽 처리
- ▼ 외부 라이브러리 활용
- ▼ JavaScript와 AJAX
- ▼ React 기본 개념 및 컴포넌트 / 상태관리 및 라우팅
  - ▼ React 소개 및 기본 개념
  - ▼ React 컴포넌트 기초
  - ▼ 이벤트 처리 및 상태 관리 기초
  - ▼ 조건부 렌더링과 리스트 렌더링
  - ▼ React Hooks 심화
  - ▼ 스타일링 개요 및 CSS-in-JS
  - ▼ 컴포넌트 재사용성과 고급 패턴
  - ▼ Context API 기초
  - ▼ React Router
  - ▼ 폼 처리와 데이터 유효성 검사
  - ▼ Redux
  - ▼ API와 비동기 처리
  - ▼ 성능 최적화
  - ▼ 테스트
  - ▼ 애니메이션과 트랜지션
  - ▼ TypeScript와 React
  - ▼ 외부 라이브러리 활용

- ▼ 배포 및 운영
- ▼ 데이터베이스 [16/64]
  - ▼ 관계형 데이터베이스 기본개념 / SQL문법 및 데이터 조작 / 데이터베이스 설계 및 모델
  - ▼ 데이터 모델링
  - ▼ 데이터베이스 설계
  - ▼ SQL
  - ▼ 트랜잭션과 무결성
  - ▼ 데이터베이스 아키텍처
  - ▼ 관계형 데이터베이스 설계
  - ▼ 데이터베이스 성능 최적화
  - ▼ NoSQL 개념 및 활용 / MongoDB 실습 / 데이터베이스 성능 최적화
    - ▼ NoSQL 개요 및 기본 개념
    - ▼ NoSQL 아키텍처 이해
    - ▼ 데이터 모델링
    - ▼ 문서형 데이터베이스 (MongoDB)
    - ▼ 키-값 데이터베이스 (Redis)
    - ▼ 열 기반 데이터베이스 (HBase)
    - ▼ 그래프 데이터베이스 (Neo4j)
    - ▼ 데이터베이스 선택 기준
    - ▼ 데이터 마이그레이션 기법
- ▼ 백엔드 개발 [32/128]
  - ▼ 스프링(부트) 기초 및 MVC패턴
    - ▼ Library / Framework
    - ▼ IoC(Inversion of Control) / DI(Dependency Injection) - @Autowired
    - ▼ Bean - @Component
    - ▼ AOP(Aspect Oriented Programming)
    - ▼ PSA(Portable Service Abstraction)

- ▼ SpringBoot 2 / SpringBoot 3
- ▼ Spring Boot 소개
- ▼ 개발 환경 설정
- ▼ Spring Boot 스타터
- ▼ 스프링 MVC 기초
- ▼ RESTful API 설계
- ▼ 데이터베이스 연동
- ▼ JPA
- ▼ 테스트
- ▼ 예외 처리
- ▼ Spring Boot와 프론트엔드 연동
- ▼ Spring Boot와 REST 클라이언트
- ▼ 파일 업로드 및 다운로드
- ▼ 스프링 부트 설정 파일
- ▼ AOP (관점 지향 프로그래밍)
- ▼ 로그 설정
- ▼ Spring Scheduler
- ▼ 이메일 발송 기능
- ▼ 웹소켓(WebSocket)
- ▼ Spring Boot Actuator
- ▼ Spring Boot 배포
- ▼ 메시지 큐 연동
- ▼ Spring Batch
- ▼ 데이터 캐싱
- ▼ 파일 및 이미지 처리
- ▼ 다국어 지원
- ▼ 스프링 시큐리티 OAuth2
- ▼ 모니터링 및 로그 분석

- ▼ 데이터 검증
- ▼ CORS 처리
- ▼ 스프링과 메시지 브로커
- ▼ 데이터 스트림 처리
- ▼ Kotlin과 Spring Boot
- ▼ 비동기 처리
- ▼ 대규모 트래픽 처리
- ▼ Python FastAPI
  - ▼ FastAPI 소개 및 기본 사용법
  - ▼ 요청(Request)와 응답(Response)
  - ▼ Path와 Query 매개변수
  - ▼ 데이터 유효성 검사 (Pydantic)
  - ▼ API 문서화 및 OpenAPI
  - ▼ 라우터(Router)와 모듈화
  - ▼ 의존성 주입(Dependency Injection)
  - ▼ 배포 준비: CORS 및 환경 변수 관리
  - ▼ 데이터베이스 연동 (SQLAlchemy)
  - ▼ 비동기 프로그래밍(Async/Await)
  - ▼ 인증 및 권한 관리
  - ▼ 파일 업로드 및 다운로드
  - ▼ 테스트 코드 작성 및 디버깅
  - ▼ WebSocket과 실시간 통신
  - ▼ 백그라운드 작업 및 스케줄링
  - ▼ 고급 라우팅 및 미들웨어
- ▼ API 개발 및 통합 [16/64]
  - ▼ RESTful API 설계 및 구현
    - ▼ SpringBoot2
      - ▼ RESTful API 개요 및 기초

- ▼ Spring Boot 환경 설정 및 시작
- ▼ 컨트롤러와 기본 API 설계
- ▼ 데이터 모델링 및 JPA 기초
- ▼ 데이터 검증 및 예외 처리
- ▼ API 설계의 모범 사례와 HATEOAS
- ▼ 인증 및 보안
- ▼ 파일 업로드 및 다운로드
- ▼ 성능 최적화 및 모니터링
- ▼ FastAPI
  - ▼ RESTful API 개요 및 Python 기초
  - ▼ API 설계 원칙과 인증/인가
  - ▼ 요청과 응답 처리
  - ▼ 데이터베이스 연동
  - ▼ API 설계 원칙과 인증/인가
  - ▼ 비동기 프로그래밍과 성능 최적화
  - ▼ 파일 업로드 및 외부 API 연동
  - ▼ API 배포 및 모니터링
- ▼ OpenAPI 및 외부 API 연동
  - ▼ SpringBoot2
    - ▼ Spring Boot 및 API 기본 개념
    - ▼ Spring Boot RESTful API 설계
    - ▼ 데이터베이스 연동
    - ▼ OpenAPI 명세와 Swagger
    - ▼ 외부 API 연동 기초
    - ▼ OAuth 2.0 인증 및 보안
    - ▼ 복잡한 API 요청 처리
    - ▼ 에러 처리 및 로깅
    - ▼ 외부 API 연동 심화



- ▼ FastAPI
  - ▼ OpenAPI 문서 이해 및 활용
  - ▼ FastAPI로 CRUD API 구현
  - ▼ 외부 API 연동 기본
  - ▼ FastAPI에서 외부 API 연동
  - ▼ 인증 및 권한 관리
  - ▼ 고급 FastAPI 기능 활용
  - ▼ 에러 처리 및 디버깅
- ▼ 세미 프로젝트 [128H]
  - ▼ 온라인 쇼핑몰 시스템
    - ▼ SpringBoot2
      - ▼ Java v11
      - ▼ Docker 및 Docker Compose
      - ▼ Spring Boot
      - ▼ React.js
      - ▼ Redis와 Cache 활용
      - ▼ MongoDB와 GridFS
      - ▼ Apache Kafka
      - ▼ ELK 스택
      - ▼ Netty 기반 네트워크 프로그래밍
      - ▼ 전체 시스템 설계
      - ▼ 백엔드 개발 및 통합
      - ▼ 프론트엔드 개발 및 통합
      - ▼ 최종 프로젝트 완성 및 배포
  - ▼ FastAPI
    - ▼ 프로그래밍 기초 및 Python 소개
    - ▼ Python 심화
    - ▼ Git 및 버전 관리

- ▼ Docker 및 Docker-compose 기초
- ▼ FastAPI
- ▼ React.js
- ▼ MongoDB 기초 및 GridFS 사용법
- ▼ Redis
- ▼ Kafka
- ▼ ELK 스택
- ▼ Twisted 프레임워크 기초
- ▼ 프로젝트 설계
- ▼ 백엔드 개발
- ▼ 프론트엔드 개발
- ▼ 통합 및 Docker-compose 구성
- ▼ ELK 및 모니터링 설정
- ▼ 대규모 파일 관리 및 최적화
- ▼ 메시징 및 비동기 처리
- ▼ 테스트 및 디버깅
- ▼ 프로젝트 최종 구현 및 배포
- ▼ 블로그 플랫폼
  - ▼ SpringBoot2
    - ▼ Java 및 객체지향 프로그래밍
    - ▼ SpringBoot
    - ▼ SpringBoot 데이터 처리
    - ▼ Docker 및 Docker-Compose
    - ▼ React.js
    - ▼ SpringBoot + React.js 연동
    - ▼ MongoDB와 Redis 기초
    - ▼ GridFS를 이용한 파일 업로드
    - ▼ Kafka를 이용한 비동기 메시징 시스템

- ▼ ELK 스택을 이용한 로깅 및 모니터링
- ▼ Netty를 활용한 실시간 통신
- ▼ 블로그 플랫폼 설계
- ▼ 사용자 인증 및 권한 관리 구현
- ▼ 게시물 CRUD 구현
- ▼ 파일 관리 시스템 구현
- ▼ 실시간 알림 시스템 구축
- ▼ 검색 및 필터링 기능 구현
- ▼ 성능 최적화 및 배포
- ▼ FastAPI
  - ▼ Python 웹 개발 기초
  - ▼ Docker 및 Docker-compose
  - ▼ FastAPI 심화
  - ▼ 데이터베이스 기본과 MySQL 활용
  - ▼ NoSQL 및 MongoDB 심화
  - ▼ React.js 기본 및 블로그 UI 개발
  - ▼ React.js 심화 및 상태 관리
  - ▼ Redis와 캐싱
  - ▼ Kafka와 메시지 브로커 활용
  - ▼ ELK 스택과 로그 관리
  - ▼ Twisted와 실시간 통신 구현
  - ▼ 블로그 플랫폼 백엔드 완성
  - ▼ 블로그 플랫폼 프론트엔드 완성
  - ▼ 배포 환경 구성 및 CI/CD
- ▼ 리스트 관리 애플리케이션
  - ▼ SpringBoot2
    - ▼ 객체지향
    - ▼ 데이터베이스

- ▼ Spring Boot
- ▼ React.js
- ▼ Docker 및 Redis
- ▼ 파일 관리 및 MongoDB/GridFS
- ▼ Kafka와 메시징 시스템
- ▼ ELK 스택을 활용한 로깅 및 모니터링
- ▼ 네트워크 프로그래밍 (Netty)
- ▼ 리스트 관리 애플리케이션 설계
- ▼ 백엔드 애플리케이션 개발
- ▼ 프론트엔드 애플리케이션 개발
- ▼ 통합 및 테스트
- ▼ FastAPI
  - ▼ Python
  - ▼ Docker-compose
  - ▼ FastAPI
  - ▼ React.js
  - ▼ MySQL
  - ▼ MongoDB
  - ▼ Redis
  - ▼ Kafka
  - ▼ ELK Stack
  - ▼ Twisted
  - ▼ 리스트 관리 애플리케이션 설계 및 기본 구현
  - ▼ 백엔드 구현
  - ▼ 프론트엔드 구현
  - ▼ 실시간 데이터 처리 구현
  - ▼ ELK를 활용한 모니터링 시스템 구축
  - ▼ Docker-compose를 활용한 서비스 배포

## ▼ 도서 관리 시스템

### ▼ SpringBoot2

- ▼ 개발 환경 설정 및 기초
- ▼ Java 기본 문법
- ▼ 객체지향 프로그래밍
- ▼ Java 컬렉션 프레임워크
- ▼ 데이터베이스 기초 및 MySQL
- ▼ MyBatis
- ▼ SpringBoot 기초
- ▼ SpringBoot REST API 개발
- ▼ Docker 및 Docker-compose 기초
- ▼ Redis 기초 및 캐싱 적용
- ▼ React.js 기초
- ▼ React.js와 도서 관리 시스템 연동
- ▼ MongoDB 및 GridFS
- ▼ Kafka 기초
- ▼ ELK 스택
- ▼ Netty 기초
- ▼ 도서 관리 시스템 백엔드 설계 및 구현
- ▼ 도서 관리 시스템 프론트엔드 구현
- ▼ 시스템 통합 및 테스트
- ▼ 배포 및 CI/CD
- ▼ 고급 기능 추가

### ▼ FastAPI

- ▼ Python
- ▼ Docker 및 Docker-compose
- ▼ FastAPI
- ▼ React.js

- ▼ 데이터베이스: MySQL 및 MongoDB
- ▼ Kafka 및 메시징 시스템
- ▼ Redis 및 캐싱
- ▼ ELK 스택 소개 및 로그 관리
- ▼ Twisted를 활용한 비동기 네트워킹
- ▼ 도서 관리 시스템 설계
- ▼ 사용자 인증 및 권한 관리
- ▼ 도서 데이터 관리
- ▼ 이미지 및 파일 관리
- ▼ 메시징 및 알림 서비스
- ▼ 로그 관리 및 모니터링
- ▼ 프로젝트 배포
- ▼ 인공지능 및 머신러닝 [24/104]
  - ▼ LangChain 기본개념 및 활용 사례
    - ▼ **PromptTemplate**
    - ▼ **Document Loaders**
    - ▼ **Text Splitters**
    - ▼ **Text Embeddings**
    - ▼ **Vectorstores**
    - ▼ **Retriever**
    - ▼ **LLMChain**
  - ▼ 딥러닝 기초 및 모델 학습 / Tensorflow / PyTorch
    - ▼ Tensorflow
      - ▼ 딥러닝 기본 개념 및 환경 설정
      - ▼ 신경망 기본 구조와 전처리
      - ▼ 모델 훈련 및 평가
      - ▼ 과적합 방지와 모델 튜닝
      - ▼ 합성곱 신경망 (CNN)

- ▼ 고급 CNN 기술
- ▼ 순환 신경망 (RNN)
- ▼ 고급 RNN 기술
- ▼ 생성 모델과 강화학습 개요
- ▼ PyTorch
  - ▼ 딥러닝 개요 및 PyTorch 기초
  - ▼ 선형 회귀와 경사하강법
  - ▼ 신경망의 기본 구조
  - ▼ 역전파 알고리즘과 옵티마이저
  - ▼ 딥러닝의 필수 구성 요소
  - ▼ 합성곱 신경망 (CNN)
  - ▼ 순환 신경망 (RNN) 및 LSTM
  - ▼ 전이 학습과 사전 학습된 모델 활용
  - ▼ 모델 평가 및 최적화
- ▼ 텍스트 분류 및 자연어 처리 모델 실습
  - ▼ PyTorch 기본
  - ▼ 신경망 기초
  - ▼ 자연어 처리 개요
  - ▼ 워드 임베딩
  - ▼ RNN과 LSTM 개념
  - ▼ 텍스트 분류 개요
  - ▼ 간단한 텍스트 분류 모델
  - ▼ 데이터 증강 및 전처리 심화
  - ▼ RNN 기반 텍스트 분류
  - ▼ 어텐션 메커니즘
  - ▼ Transformer 기본
  - ▼ BERT 모델 개요 및 활용
  - ▼ 모델 튜닝 및 하이퍼파라미터 조정

- ▼ 모델 평가 및 성능 개선
- ▼ 다중 레이블 텍스트 분류
- ▼ 모델 배포 및 응용
- ▼ 파이널 프로젝트 [184H]
  - ▼ 지능형 고객 지원 시스템
    - ▼ Python 심화 및 데이터 처리
    - ▼ Docker 및 Docker-Compose
    - ▼ FastAPI
    - ▼ React.js
    - ▼ 데이터베이스 : MySQL 및 MongoDB
    - ▼ Redis 및 Kafka
    - ▼ GridFS 및 파일 스토리지
    - ▼ Twisted 기반의 비동기 프로그래밍
    - ▼ ELK 스택을 활용한 로그 관리
    - ▼ 프로젝트 기획 및 시스템 설계
    - ▼ 백엔드 개발
    - ▼ 프론트엔드 개발
    - ▼ 파일 및 데이터 처리
    - ▼ 로그 관리 및 시스템 모니터링
    - ▼ 프로젝트 통합 및 테스트
    - ▼ 시스템 배포
  - ▼ 맞춤형 뉴스 추천 및 요약 시스템
    - ▼ Python 기초 및 필수 라이브러리
    - ▼ Python 심화 및 웹 개발 기초
    - ▼ Docker 및 Docker-compose 기본
    - ▼ FastAPI를 활용한 백엔드 개발
    - ▼ React.js를 활용한 프론트엔드 개발
    - ▼ MySQL 기초 및 데이터베이스 설계



- ▼ MongoDB 및 GridFS 활용
- ▼ Redis와 캐싱 메커니즘
- ▼ Kafka를 활용한 메시징 시스템
- ▼ ELK 스택 (Elasticsearch, Logstash, Kibana)
- ▼ Twisted로 실시간 데이터 처리
- ▼ 뉴스 데이터 수집 및 전처리
- ▼ 뉴스 요약 알고리즘 구현
- ▼ 추천 시스템 이론 및 구현
- ▼ 시스템 통합 및 Docker-compose로 배포
- ▼ 최적화, 모니터링 및 프로젝트 발표
- ▼ 개인화된 학습 추천 플랫폼
  - ▼ Python 데이터 처리 및 라이브러리 소개
  - ▼ Docker 및 Docker-compose
  - ▼ FastAPI
  - ▼ React.js
  - ▼ MySQL
  - ▼ MongoDB
  - ▼ GridFS
  - ▼ Redis
  - ▼ Kafka
  - ▼ ELK(Stack)
  - ▼ Twisted
  - ▼ 시스템 통합
  - ▼ Docker-compose를 활용한 서비스 통합
  - ▼ 개인화 학습 알고리즘 설계
  - ▼ Redis를 활용한 캐싱 및 세션 관리
  - ▼ Kafka를 활용한 이벤트 기반 시스템
  - ▼ GridFS를 이용한 대용량 파일 관리

- ▼ ELK를 활용한 로그 및 데이터 분석
- ▼ Twisted를 활용한 비동기 통신 구현
- ▼ 프로젝트 설계 및 구현
- ▼ 프로젝트 통합 및 테스트
- ▼ 배포 및 최적화
- ▼ 취업지원 [8H]