**1. 파이썬 설치 방법 (Windows 10)**

**📥 1단계: 파이썬 다운로드**

* 공식 사이트 접속 →  
  👉 [Python 공식 다운로드 페이지](https://www.python.org/downloads/windows/)
* 바로 가기 링크:  
  ➡️ [Python 3.12.3 다운로드 (64bit Windows용)](https://www.python.org/ftp/python/3.12.3/python-3.12.3-amd64.exe)

(※ 요즘은 Python 3.12 버전이 가장 안정적이고 좋아)

**⚙️ 2단계: 설치하기**

1. 다운로드한 python-3.12.3-amd64.exe 파일을 실행해.
2. **첫 화면에서 "Add Python 3.12 to PATH" 체크박스 반드시 체크!**  
   (이거 체크 안 하면 나중에 명령어 인식 안 돼서 고생함)
3. 그 다음 **"Install Now" 버튼 클릭**.
4. 설치 완료될 때까지 기다리기 (2분 정도)

**🧪 3단계: 설치 확인**

* 키보드 윈도우 키 + R → "cmd" 입력 → 명령 프롬프트 실행
* 아래 명령어 입력해봐:

bash

복사편집

python --version

* 화면에 예를 들면 Python 3.12.3 이렇게 나오면 설치 성공!

**🛠️ 추가로 추천하는 세팅 (아주 간단)**

* **pip 최신 버전으로 업데이트**

bash

복사편집

python -m pip install --upgrade pip

* (이건 나중에 FastAPI, chromadb 같은 라이브러리 설치할 때 더 빠르고 깔끔하게 하려고 하는 거야)

## 1. 파이썬 주요 라이브러리 설치

커맨드 프롬프트(cmd) 열고 순서대로 입력하면 돼.

bash

복사편집

pip install fastapi uvicorn

pip install sentence-transformers

pip install chromadb

pip install openai

pip install pdfminer.six python-docx

pip install langchain

pip install tqdm

**설명:**

* FastAPI + uvicorn : API 서버용
* sentence-transformers : 문서 임베딩 생성
* chromadb : 벡터DB 구축
* openai : LLM 호출
* pdfminer.six, python-docx : 문서(PDF, Word) 텍스트 추출
* langchain : 문서 쪼개기(chunking) 도와주는 툴
* tqdm : 진행상황(progress bar) 예쁘게 보여줌

👉 이거 한 번에 묶어서 쭉 설치해도 괜찮아.

## 2. (옵션) VSCode 설치

코딩할 때 편하게 쓰려고 설치 추천.

* 다운로드 링크:  
  👉 [Visual Studio Code 다운로드](https://code.visualstudio.com/)

설치할 때 "Add to PATH" 옵션 체크하는 것도 잊지 말고!

**Microsoft C++ Build Tools**를 설치해주면 끝나.  
(5~10분 정도 걸려)

#### 설치 방법:

1. **공식 링크 들어가기**  
   👉 [Visual C++ Build Tools 다운로드](https://visualstudio.microsoft.com/visual-cpp-build-tools/)
2. 페이지에서 Download Build Tools 버튼 클릭.
3. 설치 프로그램 실행 후,
   * "C++ build tools" 선택
   * "기본 C++ 컴파일러 도구" 체크되어 있는지 확인
   * 바로 설치 진행 (5~7GB 정도)
4. 설치 완료되면, **컴퓨터 재부팅 한 번 해주고**.
5. 다시 터미널(cmd) 열어서 아까 했던 pip install chromadb 명령어 재실행.

bash

복사편집

pip install chromadb

# 🛠️ 준비할 것 (설치할 라이브러리)

커맨드(cmd) 열고 이걸 설치해줘:

bash

복사편집

pip install pdfminer.six python-docx python-pptx tqdm

| **라이브러리** | **용도** |
| --- | --- |
| pdfminer.six | PDF 텍스트 추출 |
| python-docx | Word(docx) 텍스트 추출 |
| python-pptx | PPT(pptx) 텍스트 추출 |
| tqdm | 진행상황(Progress bar) 표시 |

# 📚 문단 쪼개기(Chunking) 스텝 기본 계획

**목표:**

raw\_texts/ 폴더 안의 텍스트 파일을 **500~700자 정도로 쪼개서**  
chunks/ 폴더에 저장한다. (RAG 챗봇 학습용으로 쓰기 좋게!)

# 🛠️ 필요한 라이브러리 설치

터미널(cmd) 열고:

bash

복사편집

pip install langchain

(※ LangChain 안에 좋은 텍스트 쪼개기 도구가 들어있어!)

# 🚀 다음 단계는 바로 **임베딩(Embedding)**

**이제부터**:

쪼갠 문단(chunk)들을 "숫자 벡터(embedding)"로 변환해서  
"벡터 데이터베이스(Vector DB)"에 저장할 거야.

(챗봇이 질문하면 비슷한 문단을 찾아주는 핵심 파트야!)

# 📚 임베딩 스텝 기본 계획

| **단계** | **설명** |
| --- | --- |
| 1 | 쪼개진 문단을 하나하나 읽는다 |
| 2 | 문단을 임베딩(숫자 벡터)로 변환한다 |
| 3 | 벡터DB(ChromaDB)에 저장한다 |

# 🛠️ 필요한 라이브러리 설치

터미널(cmd) 열고:

bash

복사편집

pip install chromadb sentence-transformers

| **라이브러리** | **역할** |
| --- | --- |
| chromadb | 로컬 벡터 DB |
| sentence-transformers | 문장 임베딩 생성 (all-MiniLM-L6-v2 모델 사용) |

좋습니다. FastAPI 백엔드에 연결할 수 있는 웹 UI는 세 가지 중 하나로 만들 수 있습니다:

1. **React (추천)** – 대중적이고 생태계가 풍부하며, 확장성과 사용자 경험이 우수합니다.
2. **Vue** – 상대적으로 배우기 쉽고, 구성 요소 기반 개발이 간결합니다.
3. **Streamlit** – 빠르게 프로토타입을 만들 수 있지만 사용자 경험은 제한적입니다.

**추천 흐름** (React 기준):

* 프론트엔드에서 사용자의 질문 입력
* FastAPI의 /ask 엔드포인트로 POST 요청
* 응답 받아서 화면에 출력

원하시는 프레임워크가 있나요, 아니면 제가 React로 기본 샘플부터 시작해 드릴까요?

4o

##### 나의 말:

React 기본샘플부터 시작하자. 혹시 vs code말고 다른 프로그래 또 설치해야하나? 설치해야하면 설치방법도 자세히 알려줘. 초등학생 수준으로

###### ChatGPT의 말:

저장된 메모리 업데이트됨

좋아요! React 기본 샘플부터 시작하는 데 필요한 준비물과 설치 방법을 **초등학생도 이해할 수 있게** 아주 쉽게 설명할게요.

## 🎒 준비물 (설치해야 하는 것들)

1. **Node.js (노드)** - React 앱을 만들고 실행하는 데 꼭 필요해요.
2. **VS Code (이미 설치했으니 OK!)**
3. **터미널 (명령어 입력하는 창)** - VS Code 안에서 사용하면 돼요.

## 1단계: Node.js 설치하기

✨ Node.js는 React의 엔진 같은 거예요. 자동차에 엔진이 없으면 못 달리듯, React도 Node.js가 필요해요!

### 📥 설치 방법

1. 인터넷에서 아래 주소를 누르거나 복사해서 브라우저에 붙여넣어요:  
   <https://nodejs.org/>
2. 화면에서 **LTS**라고 적힌 초록색 버튼을 클릭해서 다운로드해요.
3. 다운로드가 끝나면 더블클릭해서 설치해요.
   * 중간에 나오는 창에서는 그냥 **계속(Next)**, **설치(Install)** 버튼만 누르면 돼요.
4. 설치가 끝나면 컴퓨터를 껐다 켜면 더 좋아요! (꼭은 아님)

#### 1. PowerShell 실행 정책 수정

PowerShell에서 npm 스크립트 실행을 허용합니다.

##### 1.1 실행 정책 확인

* PowerShell(관리자 권한) 실행:
  + Windows 키 → "PowerShell" 검색 → "관리자 권한으로 실행".
* 현재 정책:

powershell

복사

Get-ExecutionPolicy

* + 출력: Restricted (기본값).

##### 1.2 정책 변경

* **RemoteSigned** 설정 (권장):

powershell

복사

Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy RemoteSigned

* + 프롬프트에서 Y 입력.
* **확인**:

powershell

복사

Get-ExecutionPolicy

* + 출력: RemoteSigned.

### ✅ 해결 방법: Git 설치부터 하기

1. 아래 링크로 이동해서 Git을 다운로드하세요:  
   👉 <https://git-scm.com/download/win>
2. 설치 프로그램 실행 후 기본 설정 그대로 **계속 "Next" → "Next"** 하면서 설치하세요.
   * "Git from the command line" 같은 옵션은 그대로 두면 됩니다.
   * 마지막에 **"Git Bash Here"** 같은 항목도 기본 설정이면 돼요.
3. 설치가 끝나면, **컴퓨터를 재시작하거나, 새 CMD 창을 열어** 다시 시도하세요:

bash

복사편집

git --version

이 명령어가 잘 되면 이제 준비 완료입니다.

### ✅ 해결 방법: Git 설치부터 하기

1. 아래 링크로 이동해서 Git을 다운로드하세요:  
   👉 <https://git-scm.com/download/win>
2. 설치 프로그램 실행 후 기본 설정 그대로 **계속 "Next" → "Next"** 하면서 설치하세요.
   * "Git from the command line" 같은 옵션은 그대로 두면 됩니다.
   * 마지막에 **"Git Bash Here"** 같은 항목도 기본 설정이면 돼요.
3. 설치가 끝나면, **컴퓨터를 재시작하거나, 새 CMD 창을 열어** 다시 시도하세요:

bash

복사편집

git --version

이 명령어가 잘 되면 이제 준비 완료입니다.

### 다시 시도:

bash

복사편집

cd C:\Users\BIS02\Documents\construction-chatbot

git config --global user.name "onepmis-redmine"

git config --global user.email "yonepmis-redmine@naver.com"

git에 변경분 커밋

git add .

git commit -m " requirements.txt"

git push origin main

### 🚀 다음 배포 단계 (Render에서 진행)

1. **다른 PC에서** [**https://render.com**](https://render.com) **에 로그인**
   * GitHub 계정으로 로그인 또는 계정 만들기
2. **New → Web Service 선택**
   * 연결할 GitHub 리포지토리: construction-chatbot
3. **설정 정보 입력**
   * **Name**: construction-chatbot-api
   * Root Directory: api\_server
   * **Build Command**: pip install -r requirements.txt
   * **Start Command**: uvicorn main:app --host 0.0.0.0 --port 10000
   * **Python Version**: 3.10 이상
   * **Environment**: Python