"""Base class for vector stores."""

from abc import ABC, abstractmethod

from typing import List, Dict, Any, Optional

class BaseVectorStore(ABC):

"""Abstract base class for vector stores."""

@abstractmethod

def add\_documents(self, documents: List[Any], embeddings: Optional[Any] = None) -> None:

"""Add documents to the vector store."""

pass

@abstractmethod

def similarity\_search(self, query: str, k: int = 4) -> List[Any]:

"""Search for similar documents."""

pass

@abstractmethod

def persist(self) -> None:

"""Persist the vector store."""

pass