

以下に、このコードの「入力」と「出力」を簡潔に整理します。

入力

- ユーザー入力/操作

- 質問文: テキストボックス `txtQuestion` (未入力時はエラーメッセージ)

- RAG の利用有無: チェックボックス `chkUseRAG`

- 実行ボタン: 質問 (`btnAsk`)、Word 出力 (`btnExportWord`)、PDF 出力 (`btnExportPDF`)

- 設定値 (`Web.config` 等)

- `OpenAI_API_Key` (API キー)

- `OpenAI_ChatModel` (例: `gpt-5`)

- `OpenAI_EmbeddingModel` (例: `text-embedding-3-small`)

- `OpenAI_Temperature` (数値。gpt-5 系以外に適用)

- 接続文字列: `ConnectionStrings["ConnStr"]`

- 外部 API 入力 (送信ペイロード)

- `embeddings`: { `model`, `input`: ユーザー質問 }

- `chat/completions`: { `model`, `messages`: [{`role`: "user", `content`: 最終プロンプト}], `temperature` (条件付き) }

- データベース (参照)

- テーブル: DocumentEmbeddings
- 取得列（箇所により差異あり）: Title, ChunkIndex, Chunk, Embedding（JSON のベクトル）
- タイトルフィルタ: 質問文から拡張子っぽい語を抽出（.cs/.sql 等）して LIKE 検索に使用（任意）

処理の要点（中間生成物）

- RAG 有効時の最終プロンプト生成
- 質問文の埋め込みベクトル生成
- DB からチャンクとベクトルを取得
- 類似度（コサイン類似度）で上位 topN（デフォルト 20）抽出
- 同一 Title 内で±neighbor（デフォルト 1）を近傍連結、重複 ChunkIndex の補正・重複排除
- 抜粋ブロックを結合してプロンプトに差し込み
- RAG 無効時は質問文をそのまま送信
- リトライ制御: 429/5xx/タイムアウト時に二次遅延で最大 4 回まで再試行

出力

- 画面出力（UI）
- 回答表示: litAnswer（HTML エンコード）、pnlAnswer を表示
- メッセージ: lblMessage（エラーや未入力時の文言）
- ボタン状態: 回答取得後に Word/PDF 出力ボタンを有効化

- スクリプト: Overlay 非表示スクリプトの実行
- セッション
 - Session["ChatAnswer"] にモデルの回答本文を保存
- 外部 API 出力 (受信)
 - embeddings: ベクトル (float[])
 - chat/completions: 回答本文 (最初の choice の message.content)
- ファイル出力 (ダウンロード)
 - Word (.docx)
 - ファイル名: GeneratedDoc_yyyyMMddHHmmss.docx
 - 内容: 先頭に「ChatGPT の回答」、以降回答本文を行ごとに段落化
 - PDF (.pdf)
 - ファイル名: GeneratedDoc_yyyyMMddHHmmss.pdf
 - 日本語フォント: msgothic.ttc (BaseFont.IDENTITY_H, EMBEDDED)
 - 内容: 先頭に「ChatGPT の回答」、以降回答本文を行ごとに段落化
- ログ
 - 例外時に DBLogger.LogError へ出力 (主な箇所: 質問実行、埋め込み取得、類似検索、類似度計算、Word/PDF 出力)

- エラーメッセージ例

- 「質問内容を入力してください。」
- 「Chat API エラー: ...」
- 「Embedding API エラー: ...」
- 「ChatGPT の応答が空でした。」
- 「出力する回答が見つかりません。」
- 「Word 出力エラー: ...」 「PDF 出力エラー: ...」