Lecture 39: Audio Transcription Options

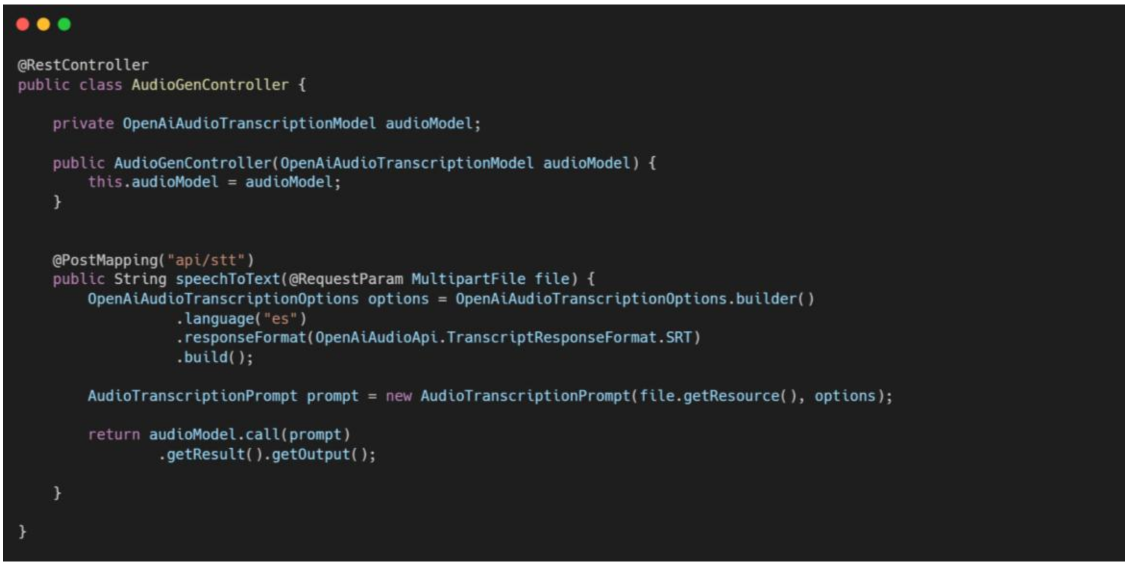
Introduction:

- When working with **audio transcription**, we sometimes need **extra control,** such as
  - Adding **timestamps** (for subtitles).
  - Changing **the language** of the output text.
  - Choosing a specific **response format**.
- By default, the transcription method accepts only a file (resource).

Audio Transcription Prompt:

- It accepts two things:
  1. **Resource** (audio file).
  2. **Options** (custom settings).
- OpenAI Audio Transcription Options
- Provides configuration using **the builder pattern**.
- Common options include:
  - language("es") → Output transcription in Spanish.
  - responseFormat(SRT) → Subtitle format with timestamps.
- This allows us to generate transcriptions with additional details, useful for subtitles and multilingual support.

Code Implementation:



```java
@RestController
public class AudioGenController {

    private OpenAiAudioTranscriptionModel audioModel;

    public AudioGenController(OpenAiAudioTranscriptionModel audioModel) {
        this.audioModel = audioModel;
    }


    @PostMapping("api/stt")
    public String speechToText(@RequestParam MultipartFile file) {
        OpenAiAudioTranscriptionOptions options = OpenAiAudioTranscriptionOptions.builder()
                  .language("es")
                  .responseFormat(OpenAiAudioApi.TranscriptResponseFormat.SRT)
                  .build();

        AudioTranscriptionPrompt prompt = new AudioTranscriptionPrompt(file.getResource(), options);

        return audioModel.call(prompt)
                .getResult().getOutput();

    }

}
```

Key Points:

- **AudioTranscriptionPrompt** is required when using options.

- Options provide flexibility for **format** (e.g., SRT for subtitles) and **language translation**.
- The call(prompt) method returns an **AudioTranscriptionResponse**, from which the final text is extracted.

Summary:

- **Without options** → transcription works in default plain text.
- **With options** → transcription supports subtitles, multiple languages, and formats.
- This makes the transcription process more powerful for real-world use like **video captions** and **multilingual content**.