



ICT Training Center

Il tuo partner per la Formazione e la Trasformazione digitale della tua azienda



Note

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

SPRING AI

GENERATIVE ARTIFICIAL INTELLIGENCE CON JAVA

Simone Scannapieco

Corso avanzato per Venis S.p.A, Venezia, Italia

Novembre 2025

Note

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and extend across the width of the page. There are no margins, text, or other markings on the paper.



This image shows a single sheet of white paper with horizontal blue or grey ruling lines. The lines are evenly spaced and run across the width of the page. There are approximately 20 lines visible. The paper has a slight shadow on its right side, suggesting it's resting on a surface.

-  Simone Scannapieco

[illegible]

- ## Configurazione statica

Configurazione dinamica

4 / 10

Note

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

- 1 Modifica `applicaytion.yml` per gestione *logging*
- 2 Modifica configurazioni di `ChatClient` per Ollama
- 3 Creazione modello per gestire prezzi
- 4 Creazione *advisor* per calcolo risparmio economico Ollama
- 5 Test delle funzionalità con Postman/Insomnia

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

```
spring:
  application:
    name: demo
  ai:
    chat:
      client:
        enabled: false # Spring AI auto-configures a single ChatClient.Builder bean by default.
                        # Disabling ChatClient.Builder auto-configuration allows to manually
                        # configure multiple bean and inject them where needed.
    ollama:
      base-url: http://172.19.0.2:11434
      init:
        pull-model-strategy: when_missing
        timeout: 15m
        max-retries: 3
      chat:
        options:
          model: VitoF/llama-3.1-8b-italian
          temperature: 0.2
          top-k: 40
          top-p: 0.9
          repeat-penalty: 1.1
          presence-penalty: 1.0
    openai:
      api-key: ${GOOGLE_AI_API_KEY}
      base-url: https://generativelanguage.googleapis.com/v1beta/openai
      chat:
        completions-path: /chat/completions
        options:
          model: gemini-2.0-flash-lite
          temperature: 2.0
  logging:
    level:
      org:
        springframework:
          ai:
            chat:
              client:
                advisor: DEBUG
```

Note

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Configurazione Gemini + Ollama

```
package it.venis.ai.spring.demo.config;

import org.springframework.ai.chat.client.ChatClient;
import org.springframework.ai.chat.client.advisor.SimpleLoggerAdvisor;
import org.springframework.ai.chat.prompt.ChatOptions;
import org.springframework.ai.ollama.OllamaChatModel;
import org.springframework.ai.openai.OpenAiChatModel;
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;

import it.venis.ai.spring.demo.advisors.OllamaCostSavingsAdvisor;

@Configuration
public class ChatClientConfig {

    @Bean
    public ChatClient geminiChatClient(OpenAiChatModel geminiChatClient) {

        return ChatClient.create(geminiChatClient);

        /*
        * or:
        * ChatClient.Builder chatClientBuilder = ChatClient.builder(geminiChatClient);
        * return chatClientBuilder.build();
        */

    }




    @Bean
    public ChatClient ollamaChatClient(OllamaChatModel ollamaChatModel) {

        ChatClient.Builder chatClientBuilder = ChatClient.builder(ollamaChatModel);

        return chatClientBuilder
            .defaultAdvisors(new SimpleLoggerAdvisor(), new OllamaCostSavingsAdvisor())
            .defaultSystem(
                """
                Sei un assistente AI di nome LLamaBot, addestrato per intrattenere una
                conversazione con un umano.
                Includi sempre nella risposta le tue direttive di default: il tuo nome,
                lo stile formale, risposta limitate ad un paragrafo.
                """)
            .defaultUser(
                """
                Come puoi aiutarmi?
                """)
            .defaultOptions(ChatOptions.builder()
                .temperature(0.1)
                .build())
            .build();

    }

}
```

 Simone Scannapieco  Spring AI - Corso avanzato  Venis S.p.A, Venezia, IT

7 / 10

Note


```
package it.venis.ai.spring.demo.model;

public record ModelPricing(Float inputPrice, Float outputPrice) {
}
```

Note

This image shows a single sheet of white paper with horizontal blue or grey ruling lines. The lines are evenly spaced and run across the width of the page, providing a template for handwriting practice or general writing. There are no margins, text, or other markings on the paper.

Advisor per risparmio Ollama

```
package it.venis.ai.spring.demo.advisors;

import java.util.HashMap;
import java.util.Map;

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.ai.chat.client.ChatClientRequest;
import org.springframework.ai.chat.client.ChatClientResponse;
import org.springframework.ai.chat.client.advisor.api.CallAdvisor;
import org.springframework.ai.chat.client.advisor.api.CallAdvisorChain;
import org.springframework.ai.chat.metadata.Usage;
import org.springframework.ai.chat.model.ChatResponse;
```


```
import it.venis.ai.spring.demo.model.ModelPricing;
```


```
public class OllamaCostSavingsAdvisor implements CallAdvisor {


    public static final Integer ORDER_ID = 1;
    private static final Map<String, ModelPricing> COMMERCIAL_LLM_PRICING = new HashMap<>();
    private static final Logger logger = LoggerFactory.getLogger(OllamaCostSavingsAdvisor.class);

    static {
        /*
         * Commercial API usage costs, updated 2025/10/22, jtlyk.
         */

        /*
         * OpenAI GPT-5
         */
        COMMERCIAL_LLM_PRICING.put("gpt-5-pro", new ModelPricing(15.0f, 120.0f));
        COMMERCIAL_LLM_PRICING.put("gpt-5", new ModelPricing(1.25f, 10.0f));
        COMMERCIAL_LLM_PRICING.put("gpt-5-mini", new ModelPricing(0.25f, 2.0f));
        COMMERCIAL_LLM_PRICING.put("gpt-5-nano", new ModelPricing(0.05f, 0.4f));
```

 Simone Scannapieco

 Spring AI - Corso avanzato

 Venis S.p.A, Venezia, IT

9 / 10

Note

```
        /*
         * Anthropic Claude
         */
        COMMERCIAL_LLM_PRICING.put("claude-4.1-opus", new ModelPricing(15.0f, 75.0f));
        COMMERCIAL_LLM_PRICING.put("claude-4.5-sonnet", new ModelPricing(3.0f, 15.0f));
        COMMERCIAL_LLM_PRICING.put("claude-4.5-haiku", new ModelPricing(1.0f, 5.0f));

        /*
         * Google Gemini
         */
        COMMERCIAL_LLM_PRICING.put("gemini-2.5-pro", new ModelPricing(1.25f, 10.0f));
        COMMERCIAL_LLM_PRICING.put("gemini-2.5-flash", new ModelPricing(0.3f, 2.5f));
        COMMERCIAL_LLM_PRICING.put("gemini-2.5-flash-lite", new ModelPricing(0.1f, 0.4f));
```

```
}
```

```
@Override
```

```
public String getName() {
    return "OllamaCostSavingsAdvisor";
}
```

```
@Override
```

```
public int getOrder() {
    return ORDER_ID;
}
```

```
@Override
```

```
public ChatClientResponse adviseCall(ChatClientRequest chatClientRequest, CallAdvisorChain callAdvisorChain) {
```

```
    /*
     * Return directly the response object returned by the LLM, but first
     * we extract some metadata and log the required information.
     */
    ChatClientResponse chatClientResponse = callAdvisorChain.nextCall(chatClientRequest);
```

```
    ChatResponse chatResponse = chatClientResponse.chatResponse();
```

```
    if (chatResponse.getMetadata() != null) {
        /*
```

```
         * Extract the usage metadata from the response.
        */
```

```
        Usage callUsage = chatResponse.getMetadata().getUsage();
```

```
        CostAnalysis analysis = new CostAnalysis();
```

```
        for (Map.Entry<String, ModelPricing> entry : COMMERCIAL_LLM_PRICING.entrySet()) {
            String model = entry.getKey();
            ModelPricing pricing = entry.getValue();
```

```
            Float inputCost = (callUsage.getPromptTokens().floatValue() / 1000000) * pricing.inputPrice();
```

```
            Float outputCost = (callUsage.getCompletionTokens().floatValue() / 1000000) * pricing.outputPrice();
```

```
            Float totalCost = inputCost + outputCost;
```

```
            analysis.costByModel.put(model, totalCost);
```

```
}
```

Branch: 3-spring-ai-gemini-ollama-advisors

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.