



# ICT Training Center

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



### Note

[illegible]

# 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 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 the right side, suggesting it's resting on a surface. There is no handwriting or other markings on the paper.



## **SETUP PROGETTO SPRING AI**

## MULTI-CONFIGURAZIONE

### Note

[illegible]

-  Simone Scannapieco

### 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.

-  Simone Scannapieco

[illegible]

- 1 Creazione `application.yml` per applicativo multi LLM
- 2 Creazione `docker-compose.yml` per servizio Docker Ollama
- 3 Creazione *file* variabili di ambiente per servizio Ollama
- 4 Creazione *script* per *start*, *stop* ed eliminazione servizi Docker
- 5 Creazione configurazione multi-LLM
- 6 Creazione modelli per domanda e risposta
- 7 Creazione interfaccia ed implementazione del servizio di richiesta
- 8 Creazione del controllore MVC
- 9 Test delle funzionalità con Postman/Insomnia

[illegible]

```
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
```

### 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, leaving small margins at the top and bottom. There are no vertical margin lines, and the paper is completely blank except for the lines themselves.

```
services:
  spring-ai-llm-gpu:
    image: ollama/ollama:${OLLAMA_VERSION:-latest}
    hostname: spring-ai-llm
    container_name: spring_ai_llm
    environment:
      OLLAMA_HOST: "${OLLAMA_HOST:-0.0.0.0}:${OLLAMA_PORT:-11434}"
      OLLAMA_DEBUG: ${OLLAMA_DEBUG:-false}
      OLLAMA_FLASH_ATTENTION: ${OLLAMA_FLASH_ATTENTION:-false}
      OLLAMA_KEEP_ALIVE: ${OLLAMA_KEEP_ALIVE:-"5m"}
      OLLAMA_MAX_LOADED_MODELS: ${OLLAMA_MAX_LOADED_MODELS:-1}
      OLLAMA_NUM_PARALLEL: ${OLLAMA_NUM_PARALLEL:-1}
    expose:
      - ${OLLAMA_PORT:-11434}
    deploy:
      resources:
        reservations:
          devices:
            - driver: nvidia
              count: all
              capabilities: [gpu]
    volumes:
      - spring_ai_llm:/root/.ollama
    restart: unless-stopped
    profiles: [llm-gpu]
```

### 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. There are no margins, text, or other markings on the paper.



...

```
volumes:
  spring_ai_llm:
    name: spring_ai_llm
```

### 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. There are no margins, text, or other markings on the paper.

```
# MODE:
# "llm-cpu" --> Large Language Model in cpu mode
# "llm-gpu" --> Large Language Model in gpu mode
MODE=llm-cpu
COMPOSE_PROFILES=${MODE}
LOG_LEVEL=WARNING                                     # default: WARNING

# Ollama configuration
#OLLAMA_VERSION=0.1.39                                # default: latest
OLLAMA_HOST=spring-ai-llm                             # default: 0.0.0.0
OLLAMA_PORT=11434                                     # default: 11434
OLLAMA_DEBUG=false                                    # default: false
OLLAMA_FLASH_ATTENTION=false                          # default: false
OLLAMA_KEEP_ALIVE="5m"                               # default: "5m"
OLLAMA_MAX_LOADED_MODELS=2                            # default: 1
OLLAMA_NUM_PARALLEL=1                                 # default: 1
```

### Note

[illegible]

```
#!/bin/bash

stack=spring-ai-demo-services
rmi=local
file=docker-compose.yml
envfile=spring-ai.env

if [ -z ${1+x} ];
then rmi=local;
else rmi=$1;
fi

## --rmi flag must be one of: all, local
docker compose -f $file -p $stack --env-file $envfile up --build --remove-orphans --force-recreate --detach
```

### Note

[illegible]

```
#!/bin/bash

stack=spring-ai-demo-services
rmi=local
file=docker-compose.yml
envfile=spring-ai.env

if [ -z ${1+x} ];
then rmi=local;
else rmi=$1;
fi

## --rmi flag must be one of: all, local
docker compose -f $file -p $stack --env-file $envfile down --rmi $rmi
```

### Note

[illegible]

## **File** `erase_spring_ai_services.sh`

```
#!/bin/bash

stack=spring-ai-demo
rmi=local
file=docker-compose.yml
envfile=spring-ai.env

if [ -z ${1+x} ];
then rmi=local;
else rmi=$1;
fi

## --rmi flag must be one of: all, local
docker compose -f $file -p $stack --env-file $envfile down --rmi $rmi --volumes
```

### 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.ollama.OllamaChatModel;
import org.springframework.ai.openai.OpenAiChatModel;
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;

@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.build();
}
/*
 * or:
 * return ChatClient.create(ollamaChatModel);
 */
}
```

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

import java.util.UUID;

public record Question(UUID id, String question) {

    public Question(String question) {

        this(UUID.randomUUID(), question);

    }

}
```

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

public record Answer(String answer) {
}
```

## 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.

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

import it.venis.ai.spring.demo.model.Answer;
import it.venis.ai.spring.demo.model.Question;

public interface QuestionService {

    Answer getGeminiAnswer(Question question);

    Answer getOllamaAnswer(Question question);

}
```

[illegible]



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

...

@Service
@Configuration
public class QuestionServiceImpl implements QuestionService {

    private final ChatClient geminiChatClient;
    private final ChatClient ollamaChatClient;

    public QuestionServiceImpl(@Qualifier("geminiChatClient") ChatClient geminiChatClient,
                              @Qualifier("ollamaChatClient") ChatClient ollamaChatClient) {
        this.geminiChatClient = geminiChatClient;
        this.ollamaChatClient = ollamaChatClient;
    }

    @Override
    public Answer getGeminiAnswer(Question question) {
        return new Answer(this.geminiChatClient.prompt()
            .user(question.question())
            .call()
            .content());
    }

    @Override
    public Answer getOllamaAnswer(Question question) {
        return new Answer(this.ollamaChatClient.prompt()
            .user(question.question())
            .call()
            .content());
    }
}
```

[illegible]

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

import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RestController;

import it.venis.ai.spring.demo.model.Answer;
import it.venis.ai.spring.demo.model.Question;
import it.venis.ai.spring.demo.services.QuestionService;

@RestController

public class QuestionController {

    private final QuestionService service;

    public QuestionController(QuestionService service) {
        this.service = service;
    }

    @PostMapping("/gemini/ask")
    public Answer geminiAskQuestion(@RequestBody Question question) {
        return this.service.getGeminiAnswer(question);
    }

    @PostMapping("/ollama/ask")
    public Answer ollamaAskQuestion(@RequestBody Question question) {
        return this.service.getOllamaAnswer(question);
    }
}
```

## 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.

18 / 18

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.