



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



Note			



### **SPRING AI**

### GENERATIVE ARTIFICIAL INTELLIGENCE CON JAVA

#### Simone Scannapieco

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

Settembre 2025

Note		

	JSON OUTPUT CONVERTER PROMPT TEMPLATES
Note	

Note			

# PROGETTO SPRING AI



- 1 Nuovo string template per richiesta definizione da dizionario con formato di output automatizzato
- Creazione modelli per riposta da LLM
- 3 Modifica file GeminiFromClientService.java e GeminiFromClientServiceImpl.java
- 4 Modifica file QuestionFromClientController.java
- 5 Test delle funzionalità con Postman/Insomnia

Note		

# PROGETTO SPRING AI MODELLO DI RICHIESTA



⚠ Il path resources/templates viene creato automaticamente da Spring AI in fase di creazione del progetto

File get-json-output-converter-format-definition-prompt.st

Dammi una definizione del termine {lemma}, come riporterebbe un dizionario. {formato}

🐸 Simone Scannapieco

Spring AI - Corso base

m Venis S.p.A, Venezia, IT

4/10

Note		

# PROGETTO SPRING AI RISPOSTA



### Modello di risposta

### Modello di sotto-risposta (esempi)

```
package it.venis.ai.spring.demo.model;
public record DefinitionExample(String example) {}
```

🐸 Simone Scannapieco

Spring AI - Corso base

m Venis S.p.A, Venezia, IT

5/10

# PROGETTO SPRING AI SERVIZIO GEMINI



### Interfaccia servizio Gemini (da client)

```
package it.venis.ai.spring.demo.services;
import it.venis.ai.spring.demo.model.Answer;
import it.venis.ai.spring.demo.model.DefinitionRequest;
import it.venis.ai.spring.demo.model.DefinitionResponse;
import it.venis.ai.spring.demo.model.Question;

public interface GeminiFromClientService {
    String getAnswerFromClient(String question);
    Answer getAnswerFromClient(Question question);
    Answer getDefinitionFromClient(DefinitionRequest definitionRequest);
    Answer getCustomFormatDefinitionFromClient(DefinitionRequest definitionRequest);
    DefinitionResponse getJSONOutputConverterFormatDefinitionFromClient(DefinitionRequest definitionRequest definitionRequest);
}
```

🐸 Simone Scannapieco

Spring AI - Corso base

🧰 Venis S.p.A, Venezia, IT

6/10

# PROGETTO SPRING AI SERVIZIO GEMINI



### Implementazione servizio Gemini (da client) - I

```
package it.venis.ai.spring.demo.services;
 import java.util.Map;
{\tt import org.springframework.ai.chat.client.ChatClient};
 {\tt import org.springframework.ai.template.st.StTemplateRenderer;}
{\tt import\ org.springframework.beans.factory.annotation. Value;}
import org.springframework.core.io.Resource;
import org.springframework.stereotype.Service;
{\tt import it.venis.ai.spring.demo.model.Answer;}
{\tt import it.venis.ai.spring.demo.model.DefinitionRequest;}
import it.venis.ai.spring.demo.model.Question;
\verb"public class GeminiFromClientServiceImpl" implements GeminiFromClientService \{ in the content of the conten
              private final ChatClient chatClient;
              public GeminiFromClientServiceImpl(ChatClient.Builder chatClientBuilder) {
                          this.chatClient = chatClientBuilder.build();
             public String getAnswerFromClient(String question) {
                      return this.chatClient.prompt()
                                                    .user(question)
                                                     .call()
                                                    .content();
             }
```

Simone Scannapieco

Spring AI - Corso base

🏛 Venis S.p.A, Venezia, IT

7/10

# PROGETTO SPRING AI SERVIZIO GEMINI



### Implementazione servizio Gemini (da client) - II

Simone Scannapieco

Spring AI - Corso base

m Venis S.p.A, Venezia, IT

8/10

# PROGETTO SPRING AI MVC DEL SERVIZIO GEMINI



9/10

m Venis S.p.A, Venezia, IT

### Implementazione controllore REST (da client)

Spring AI - Corso base

```
package it.venis.ai.spring.demo.controllers;
import org.springframework.web.bind.annotation.RestController;
import it.venis.ai.spring.demo.services.GeminiFromClientService;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;
public class QuestionFromClientController {
    {\tt private\ final\ GeminiFromClientService\ geminiService;}
    @PostMapping("/client/definition/custom")
    {\tt public \ Answer \ getCustomFormatDefinition(@RequestBody \ DefinitionRequest \ definitionRequest) \ \{ \ \ \ \} }
         \textbf{return this}. \texttt{geminiService}. \texttt{getCustomFormatDefinitionFromClient} (\textbf{definitionRequest});\\
    @PostMapping("/client/definition/json/user")
    public Answer getJSONUserFormatDefinition(@RequestBody DefinitionRequest definitionRequest) {
         return this.geminiService.getJSONUserFormatDefinitionFromClient(definitionRequest);
    @PostMapping("/client/definition/json/converter")
    public DefinitionResponse getJSONOutputConverterFormatDefinition(@RequestBody DefinitionRequest definitionRequest) {
         \textbf{return this}. \texttt{geminiService}. \texttt{getJSONOutput} \texttt{ConverterFormatDefinitionFromClient} (\texttt{definitionRequest}); \\
```

N	lata	

Simone Scannapieco

Note			