

SpringAI

依赖

- 依赖
 - 引入 SpringAI 依赖需要引入 SpringAI 依赖管理

```
<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.springframework.ai</groupId>
      <artifactId>spring-ai-bom</artifactId>
      <version>${spring-ai.version}</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>
```

- 引入ollama

```
<dependency>
  <groupId>org.springframework.ai</groupId>
  <artifactId>spring-ai-ollama-spring-boot-starter</artifactId>
</dependency>
```

- 配置
 - ollama地址配置url

```
spring:
  ai:
    ollama:
      base-url: url
      chat:
        model: 
```

- 配置
 - 配置bean
 - Bean** ChatClient OllamaChatModel ** builder

```
@Bean
public ChatClient chatClient(OllamaChatModel model){
    return ChatClient.builder(model)
        .defaultSystem(" ")
        .build();
}
```

- `chatClient` `chatClient` `chatClient`

```
@Autowired
private ChatClient chatClient;
String content = chatClient.prompt()
    .user("user")
    .call()
    .content();
```

chatClient

- `ChatClient.defaultSystem()` `ChatClient`
- `ChatClient.defaultAdvisors()` `Advisor`
- `chatClient.prompt()` `ChatClient`
- `chatClient.user()` `ChatClient`
- `chatClient.call()` `ChatClient`
- `chatClient.stream()` `Flux<String> content`
- `chatClient.content()` `ChatClient`
- `chatClient.advisors()` `context`
 - `.advisors(a -> a.param(CHAT_MEMORY_CONVERSATION_ID_KEY, chatId))` `context`
 - `CHAT_MEMORY_CONVERSATION_ID_KEY` `AbstractChatMemoryAdvisor`

chatClient

SpringAI `AOP` `Advisor` `chatClient`

image-20250915194026653

Advisor

- `SimpleLoggerAdvisor`
- `MessageChatMemoryAdvisor`
- `QuestionAnswerAdvisor` `RAG`

chatClient

chatClient

- `SpringAI` `Redis` `MongoDB`

```
public interface ChatMemory{
    void add(String conversationId, List<Message> messages);
    List<Message> get(String conversationId, int lastN);
}
```

```
void clear(String conversationId);  
}
```

- SpringAI `InMemoryChatMemory` **`InMemoryChatMemory` **`InMemoryChatMemory`
- `ChatClientAdvisor`
 - `ChatClient.defaultAdvisors()` `ChatClient`
- `chatId`
 - `chatClient.advisors()` `chatId`

`chatId`

`chatId`

- `chatId` `chatId`
- `chatId` `ChatMemory` `get` `chatId`