

# G3J Learn Semantic Kernel

On LIVE!

# 시멘틱커널의 모든 것

박구삼

Microsoft MVP

# Agenda

01

Semantic Kernel(SK)이란?

02

G3J Learn SK  
라이브 세션 하이라이트!

03

가이드 프로젝트 소개

# 01

## Semantic Kernel(SK)이란?

Semantic Kernel은 AI 언어 모델을  
App에 쉽게 통합할 수 있는 SDK입니다.

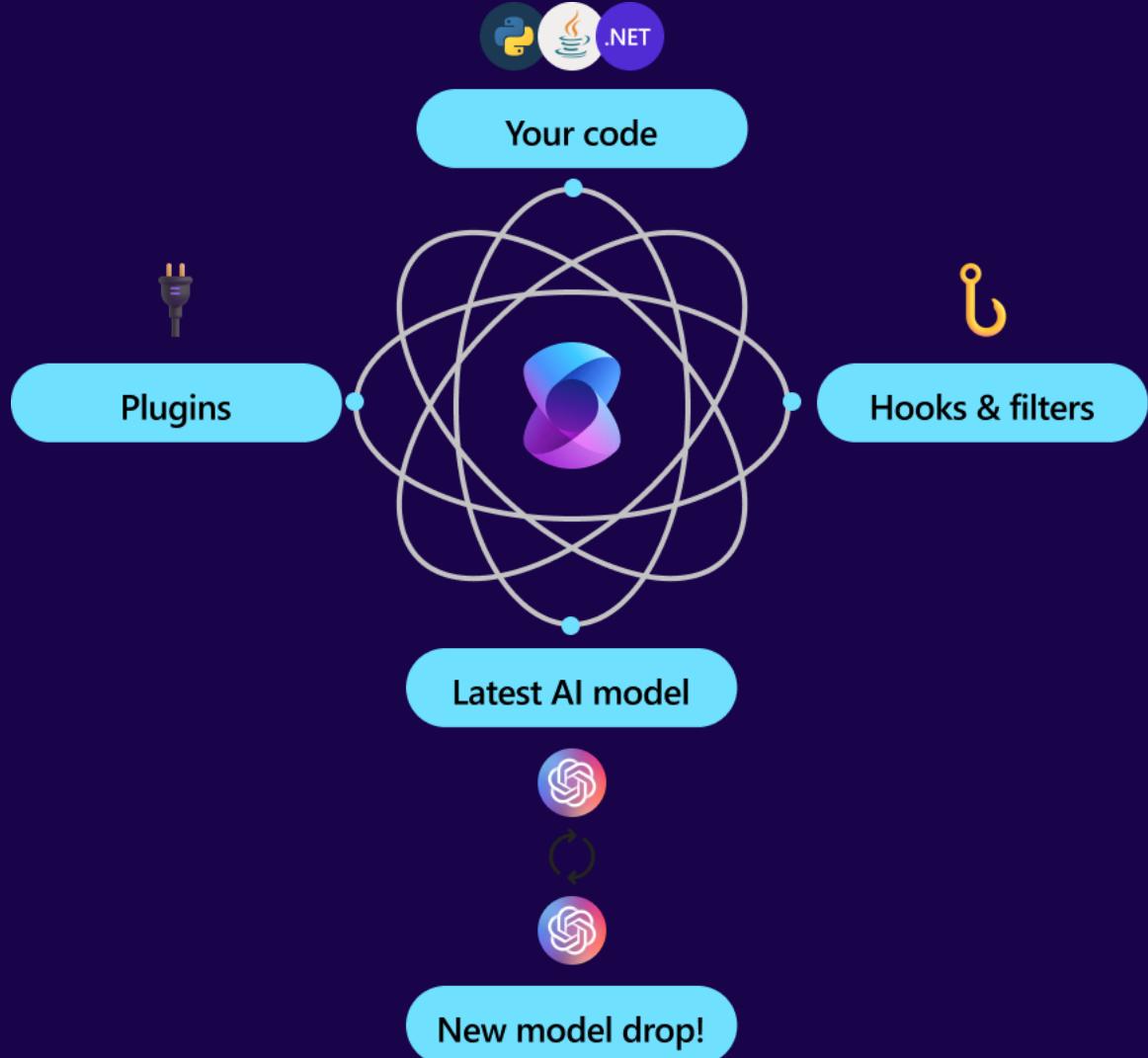
**그리고 Open Source 프로젝트입니다!**

<https://github.com/microsoft/semantic-kernel>



# AI 서비스와 연결을 유연하게!

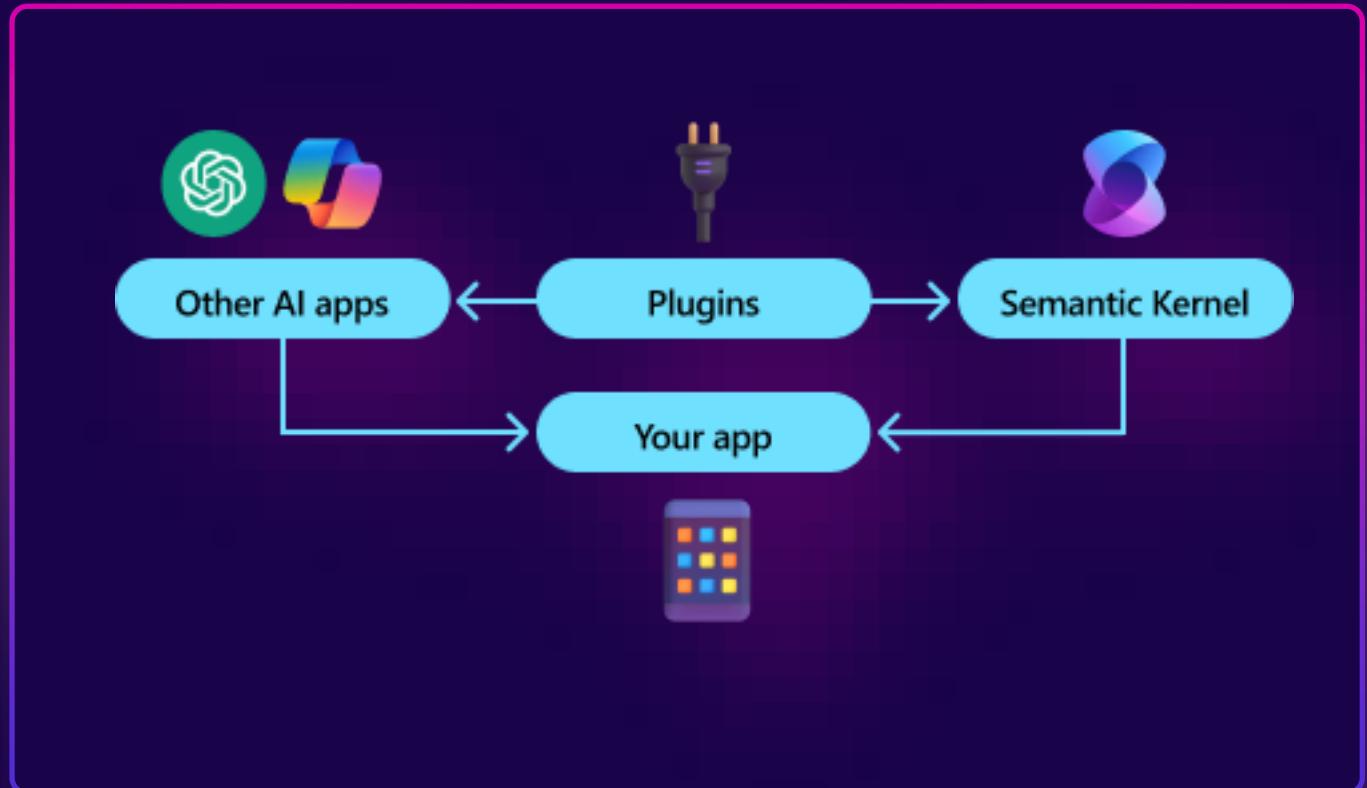
사용중인 AI 서비스의 모델을 쉽게 연결할 수 있고 또 안전하면서 기능 확장이 필요하다면 Semantic Kernel을 통해 플러그인 형태로 가공 및 원하는 상황에 이용할 수 있도록 디자인이 가능합니다.



# 모듈식 확장 가능

## 기존 App에 모듈을 붙이는 확장 가능

- 기존에 사용하는 AI 서비스가 있고
- 기존에 작성한 여러분의 App이 있는 환경이라면
- Semantic Kernel을 모듈식으로 붙여 쓰기 가능!





## Semantic Kernel이란?

Microsoft의 AI 오케스트레이션 프레임워크

"다양한 AI 서비스와 기존 코드를 하나로 연결하는 지휘자"

## 할 수 있는 것!

- OpenAI, Azure OpenAI, Hugging Face 등 다양한 AI 서비스 지원
- C#, Python, Java로 동일한 개발 경험 제공
- 플러그인 시스템으로 기능 확장



# System Requirements



.NET

8.0 이상



Python

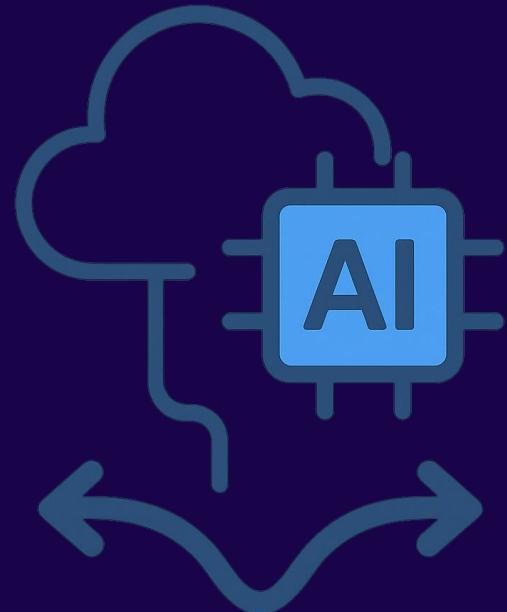
3.10 이상



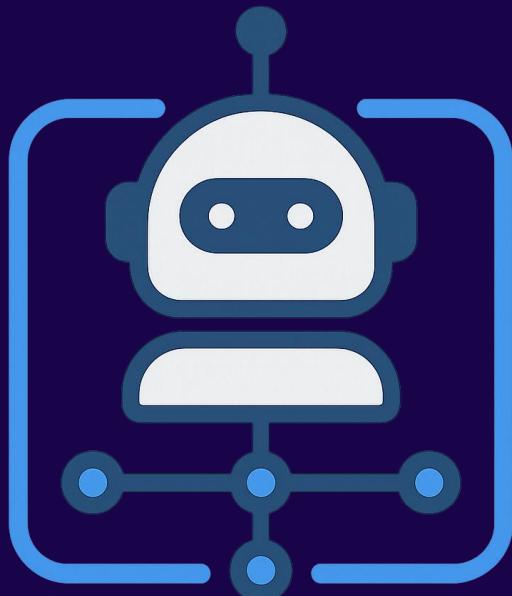
Java

JDK 17 이상

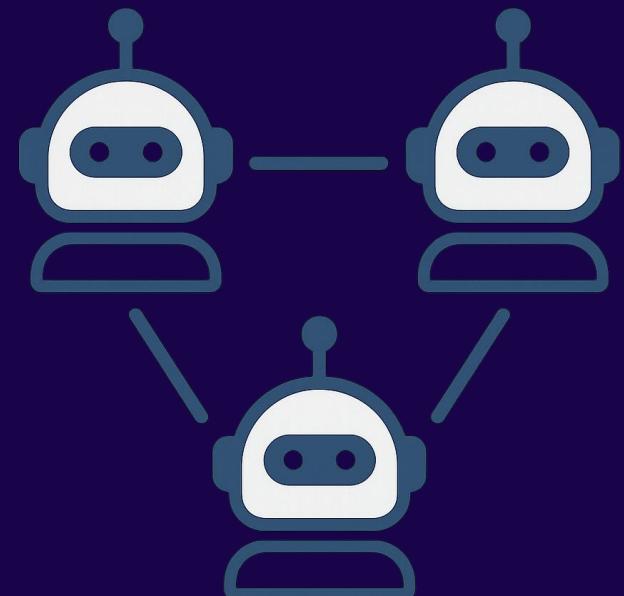
# Key Features



Model Flexibility

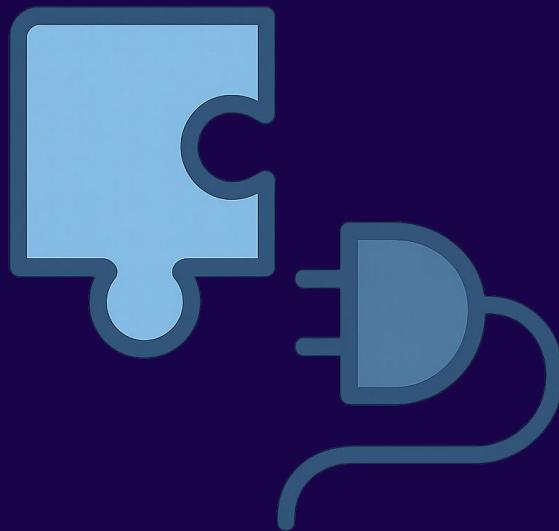


Agent Framework



Multi-Agent Systems

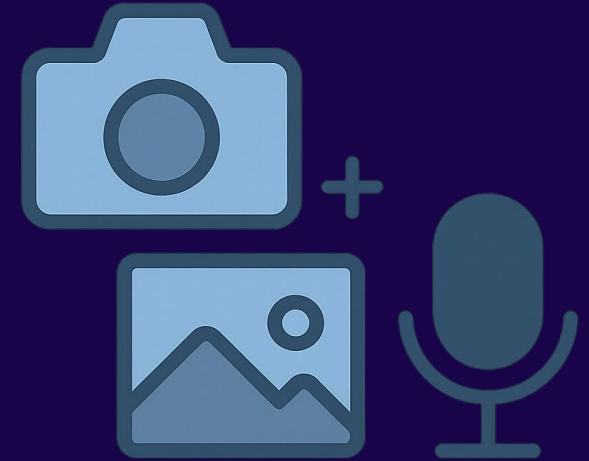
# Key Features



Plugin Ecosystem

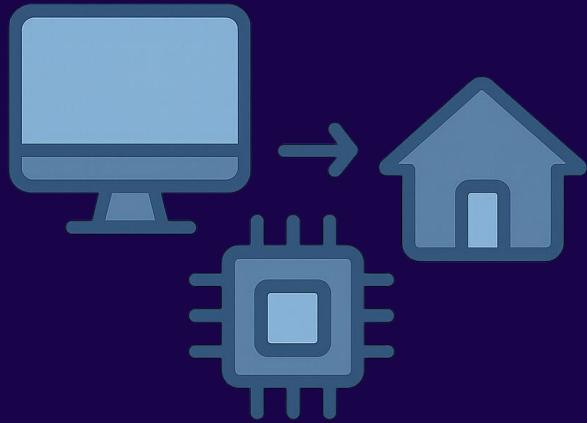


Vector DB Support

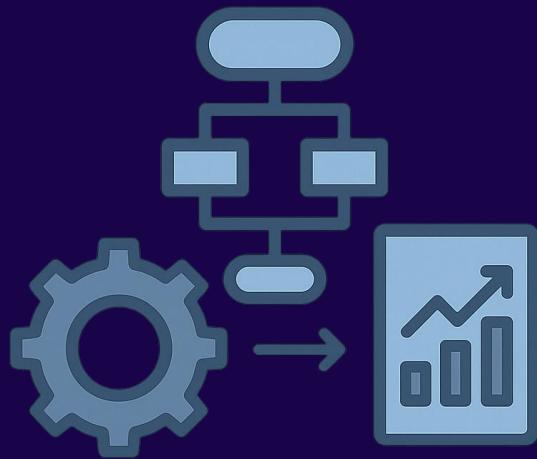


Multimodal Support

# Key Features



Local Deployment



Process Framework



Enterprise Ready



# 안녕! Semantic Kernel!

Semantic Kernel 시작해보기

# AI Agent의 시대

## Semantic Kernel과 함께!

# 02

## G3J Learn SK 라이브 세션 하이라이트!



# G3J 시리즈 콜렉션

이번 G3J Learn 시리즈는 콜렉션을 제공하고 있습니다.  
혹시라도 막히는 부분이 있거나 라이브 중 참고한 자료 및 다시보기를  
알고 싶다면 이 링크를 이용해주세요!

<https://aka.ms/g3jlearnsk>



# G3J Learn Semantic Kernel 시리즈



박구삼  
Microsoft MVP



유저스틴  
Microsoft



김진석  
Microsoft MVP



이종인  
Microsoft MVP

G



박구삼  
Microsoft MVP

J



유저스틴  
Microsoft

J



김진석  
Microsoft MVP

J



이종인  
Microsoft MVP

G



박구삼  
Microsoft MVP

J



유저스틴  
Microsoft

J



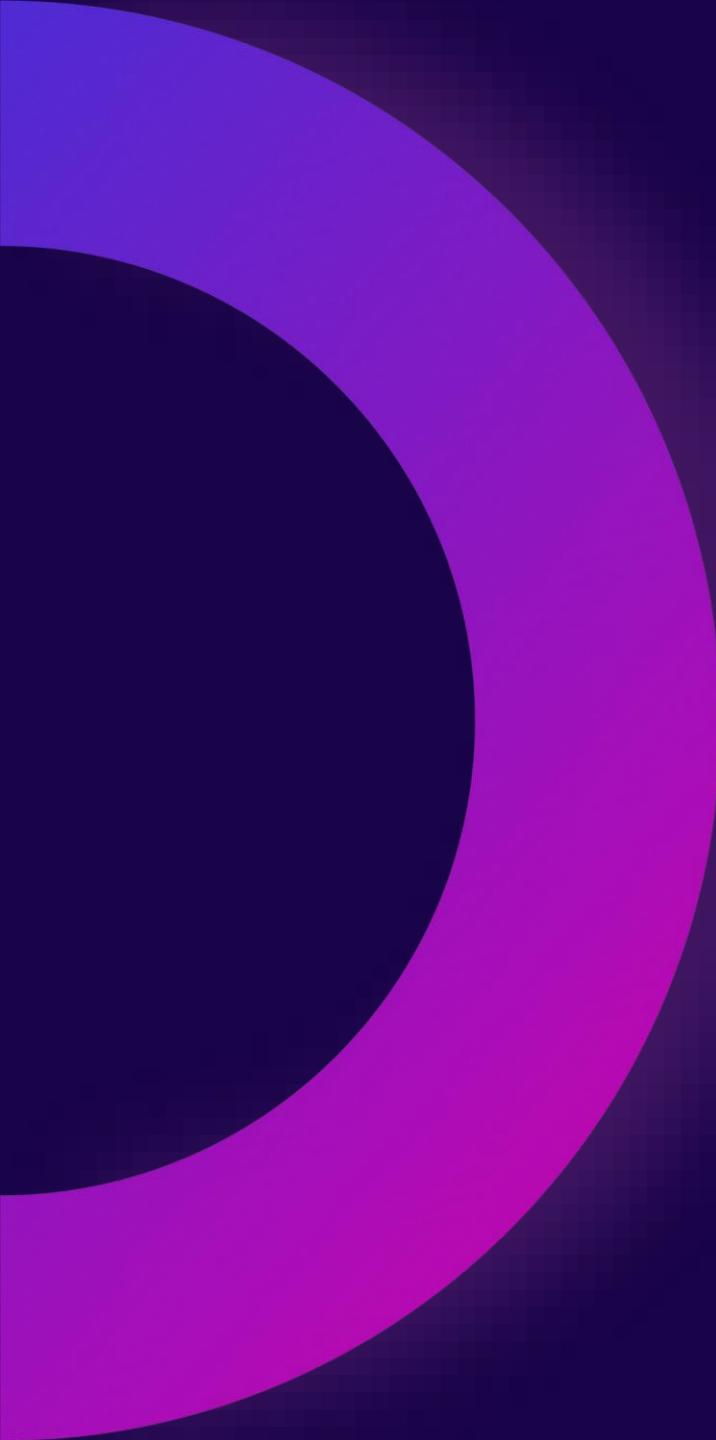
김진석  
Microsoft MVP

J



이종인  
Microsoft MVP

G3J Learn SK!



G3J 12개 세션 중  
가장 인기 있는 5위!

**Session 4.**  
**SK 오토 평션 콜링**



# Agenda

01

Semantic Kernel  
챗 히스토리 구현

02

Semantic Kernel  
페르소나 설정

03

Semantic Kernel  
오토 평션 콜링



```
4  using Microsoft.Extensions.Configuration;
5  using Microsoft.SemanticKernel;
6  using Microsoft.SemanticKernel.ChatCompletion;
7
8
9  > Configuration
19
20  var kernel = Kernel.CreateBuilder()
21      .AddAzureOpenAIChatCompletion(deployName, endPoint, key, model)
22      .Build();
23
24  Console.WriteLine("Hello, Semantic Kernel!");
25  // 챗 컴플리션???
26  // Chat Completion : Generate a response given a list of messages in a conversational context
27  // 주어진 메시지 목록(맥락이 있는)에서 답변을 생성하는 것
28
29  // Chat History
30  // System Message : 사용자가 변경할 수 없는 유용한 정보를 담는 곳. 페르소나, 그라운딩, 세이프티 메시지 등
31  // User Message : 사용자가 입력한 메시지
32  // Assistant Message : LLM이 답변한 메시지
33
34  // 샘플 1 : 개요, 비스트리밍 챗 컴플리션
35  Console.WriteLine("#nSample1");
36  await Sample.showOverview(kernel);
37
38
39  < // 샘플 2 : 챗히스토리 오픽젝트, 스트리밍 챗 컴플리션
40  //Console.WriteLine("#nSample2");
41  //await Sample.showChatHistory(kernel);
42
43  // 샘플 3 : 페르소나
```

내용  
도구  
상자



The screenshot shows a Microsoft Teams video conference with three participants:

- 유저스틴 | Microsoft**: A man wearing glasses and headphones, speaking.
- 이종인 | Microsoft MVP**: A man in a black t-shirt, listening.
- 김진석 | Microsoft MVP**: A man in a white hoodie, listening.

The main focus is a code editor window titled "Appointment.cs". It displays C# code for a chatbot assistant:

```
// Print the result
Console.WriteLine("Assistant > " + result);

// Add the message from the agent to the chat history
chatHistory.AddMessage(result.Role, result?.Content!);
```

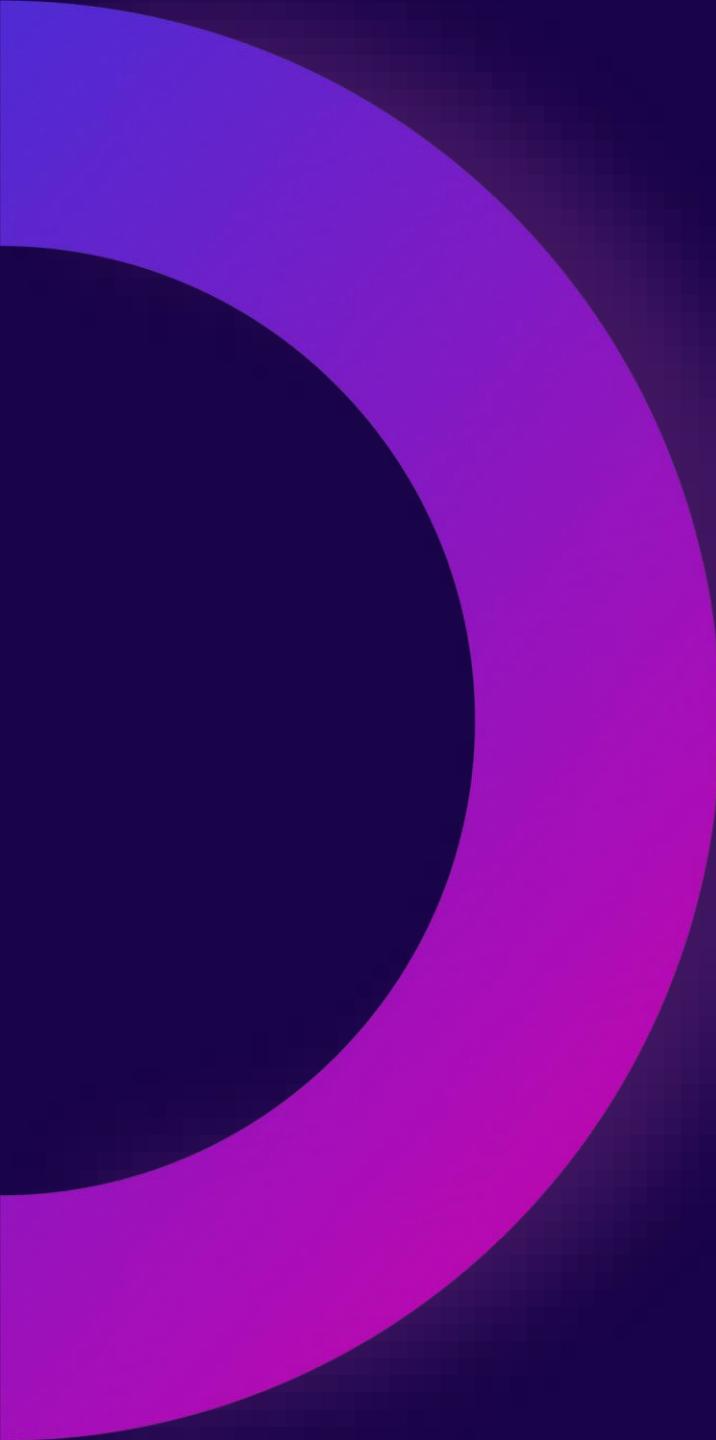
The code includes a comment block and two method calls. The first call prints the result to the console, and the second adds the message to the chat history. Below the code, there is a timestamp "[ ] 2m 45.5s" and a status bar indicating "csharp - C# Script Code".

A modal dialog box is open at the top of the screen, asking "무엇을 도와드릴까요? (Press 'Enter' to confirm or 'Escape' to cancel)". The status bar at the bottom of the screen shows various icons and text, including "main\*", "Launchpad", "No Solution", "CRLF", "Cell 11 of 11", "kiota 1.20.0", and "Prettier".



## Planners: 자동 계획 수립





G3J 12개 세션 중  
가장 인기 있는 4위!

Session 9.  
Kernel Memory





박구삼 | Microsoft MVP

# Agenda

01

커널 메모리 소개

02

동기식 메모리 API

03

비동기식 메모리 API



# Kernel Memory

## Import

`await memory.ImportTextAsync(@"물리학에서 질량-에너지 등가성은 시스템의 정지 상태에서 질량과 에너지 사이의 관계를 말하며,  
이 두 양은 곱셈 상수와 측정 단위에 의해서만 차이가 납니다.  
이 원리는 물리학자 알베르트 아인슈타인의 공식인  $E = m*c^2$ 로 설명됩니다.");`

`await memory.ImportDocumentAsync("file4-KM-Readme.pdf");`

## Ask

$E = m*c^2$ 는 알베르트 아인슈타인이 제안한 질량-에너지 등가성의 공식입니다. 이 공식은 에너지( $E$ )와 질량( $m$ ) 사이의 관계를 설명하며,  $c$ 는 빛의 속도(약 299,792,458 미터/초)의 제곱을 나타냅니다. 이 공식에 따르면, 질량이 있는 물체는 그 질량에 비례하는 에너지를 가지고 있으며, 이 에너지는 물체가 정지해 있을 때의 질량과 관련이 있습니다. 즉, 질량과 에너지는 서로 변환 가능하며, 질량이 에너지로 변환될 수 있고, 반대로 에너지가 질량으로 변환될 수 있다는 것을 의미합니다. 이 원리는 물리학에서 매우 중요한 개념으로, 핵반응이나 입자 물리학 등 다양한 분야에서 적용됩니다. 질량-에너지 등가성은 물리학의 기본 원리 중 하나로, 우주와 물질의 본질을 이해하는 데 중요한 역할을 합니다.

The screenshot shows a Microsoft Visual Studio interface with the following details:

- Project:** KernelMemoryPractice
- File:** Demo1.cs
- Code:** A snippet of C# code using the `async Task` pattern to interact with a memory service.

```
public async Task Search()
{
    var index = Guid.NewGuid().ToString().Replace("-", "");
    var memory = KernelMemoryGenerator.GetMemory();

    var docId = await memory.ImportDocumentAsync("file");
    var question = "Kernel Memory가 뭐야?";

    var answer = await memory.SearchAsync(question,
        output.WriteLine(JsonSerializer.Serialize(answer
    {
        WriteIndented = true,
        Encoder = JavaScriptEncoder.UnsafeRelaxedJsonEncoder
    }));
}
```

- Test Results:** The left pane shows a list of tests with their execution times. One test, "Search (2)", is highlighted.

테스트	기간
AskToExcel	2.2초
AskToExcel	3.7초
AskToPdf (2)	12초
AskToPdf	4.5초
AskToPdf	7.5초
AskToText (2)	39.4초
AskToText	36.2초
AskToText	3.2초
Chunking_100 (2)	56.6초
Chunking_300 (2)	25초
Download (2)	2.3초
Search (2)	4.1초

- GitHub Copilot:** A small icon in the top right corner indicates the feature is active.
- Bottom Status Bar:** Shows the file name (main), line count (11 0/0), and date (오늘 8:24 2025-02-20).



박구삼 | Microsoft MVP



이종인 | Microsoft MVP



박구삼 | Microsoft MVP

이종인 | Microsoft MVP



## 실무 패턴: RAG (검색 증강 생성)



### 문서 수집

PDF, Word, 웹페이지 등 다양한 소스에서 텍스트 추출



### 청킹

긴 문서를 의미 있는 단위로 분할



### 임베딩

텍스트를 벡터로 변환하여 의미론적 검색 가능

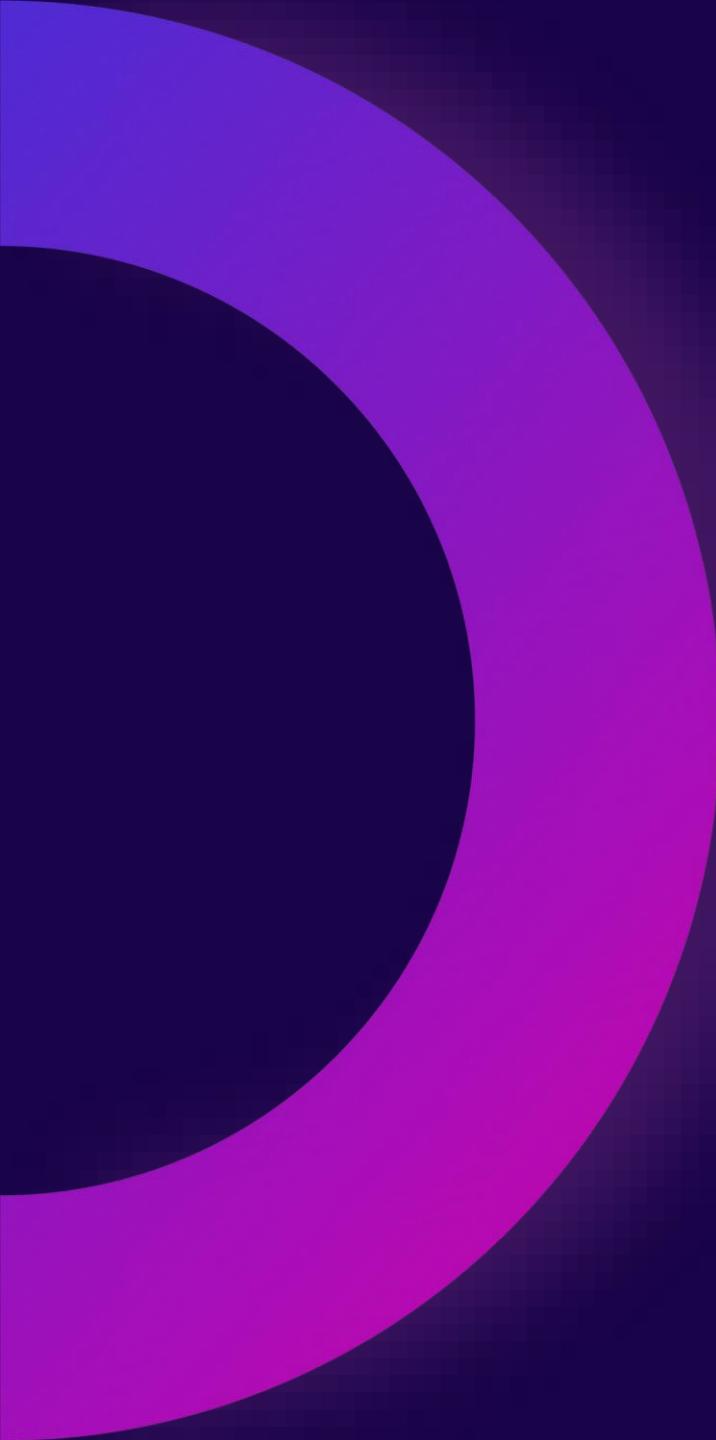


### 관련성 검색

질문과 가장 관련성 높은 문서 조각 찾기

## 실제 활용 사례

- 기업 내부 문서 검색 시스템
- 고객 지원 챗봇
- 법률/의료 문서 분석



G3J 12개 세션 중  
가장 인기 있는 3위!

**Session 3.**  
**SK Plugin**





박구삼 | Microsoft MVP

# Agenda

01

Semantic Kernel  
플러그인 소개

02

프롬프트 플러그인  
제작 및 연동

03

네이티브 플러그인  
제작 및 연동





```
18 var plugins = kernel.CreatePluginFromPromptDirectory("Prompts");
19
20 var history = @"""
21     저는 매운 음식을 아주아주 좋아합니다.
22     한식, 일식, 중식 등 아시아권 음식 또한 좋아합니다.
23     그런데 어제는 짜장면, 잡채밥, 돼지국밥, 회초밥을 먹었고
24     오늘은 쌀국수랑 돈까스 나베를 먹었습니다.
25 """
26
27 Console.OutputEncoding = Encoding.UTF8;
28 Console.WriteLine("어떤 음식 종류로 추천해드릴까요?");
29
30 var category = Console.ReadLine();
31
32 var result = await kernel// Kernel
33 .InvokeAsync(
34     plugins["RecommendFood"],
35     new KernelArguments()
36     {
37         { "history", history },
38         { "foodCategory", category }
39     } // Task<FunctionResult>
40     .ConfigureAwait(false); // ConfiguredTaskAwaitable<FunctionResult>
41
42 Console.WriteLine(result);
```

맨 앞줄 내용은 프롬프트에 성격을 부여하는 건가요?



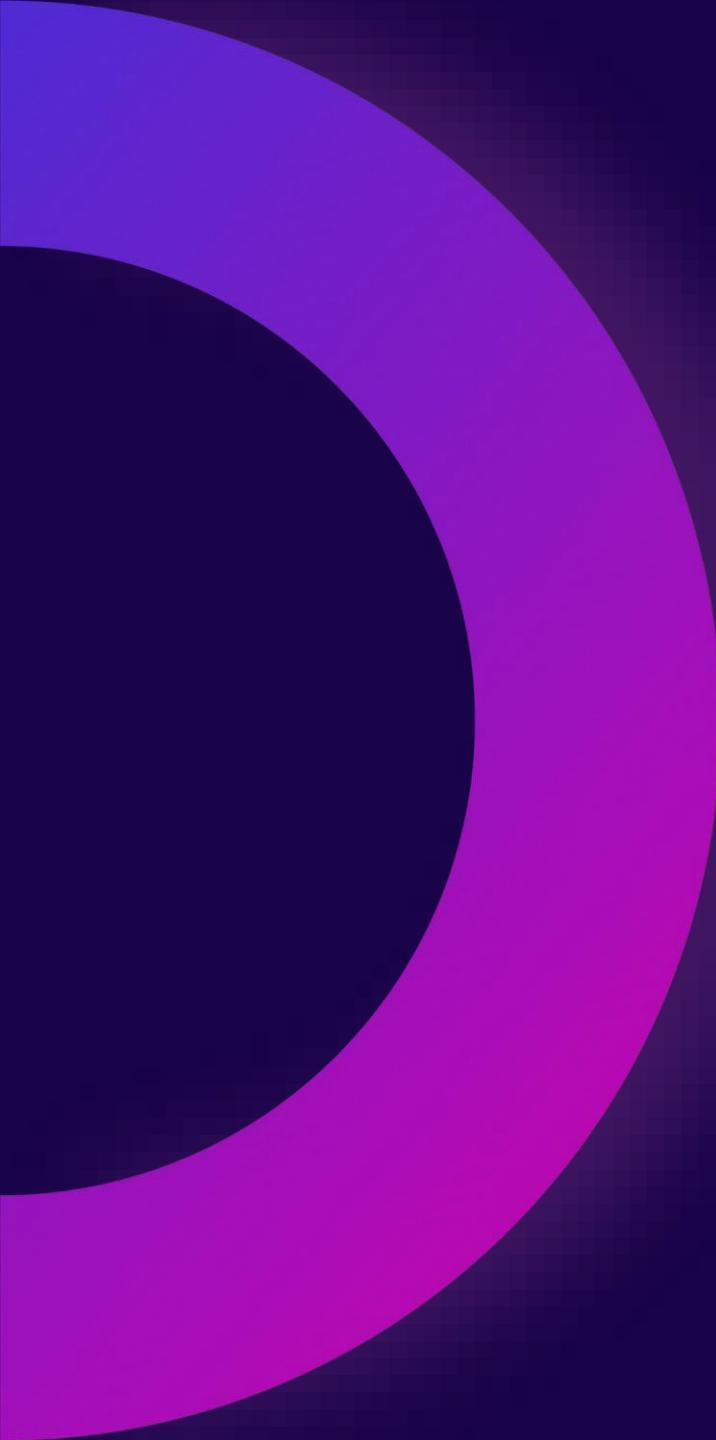
The screenshot shows a Visual Studio interface with the following details:

- Top Bar:** 파일(F), 보기(V), Git(G), 프로젝트(P), 편집(E), 디버그(D), 테스트(S), 분석(N), 도구(T), 확장(X), 장(W), 도움말(H), 검색 - demo02.
- Toolbar:** Debug, Any CPU, demo02.AppHost, https://
- Left Panel:** Programs.cs, GamePlugin.cs, Program.cs, demo02.OpenApiPlugin (selected). The code in demo02.OpenApiPlugin is as follows:

```
1 // See https://aka.ms/new-console-template for more information
2 using Microsoft.SemanticKernel.Plugins.OpenAPI;
3 using Microsoft.SemanticKernel;
4 using Microsoft.Extensions.Configuration;
5
6 var config = new ConfigurationBuilder()
7     .AddEnvironmentVariables()
8     .Build();
9
10 var service = config["services:apiservice:https:0"];
11 var kernel = Kernel.CreateBuilder()
12     .Build();
13
14 var plugin = await kernel.ImportPluginFromOpenAPIAsync("apiPlugin", new Uri($"{service}/openapi/v1.json"));
15 var result = await kernel.InvokeAsync(plugin["FlipCoin"]);
16 Console.WriteLine($"FlipCoin : {result}");
17 Console.WriteLine("End...");
```

- Right Panel:** 솔루션 탐색기 (Solution Explorer) showing the project structure for 'demo02' (6 projects):
  - demo02.ApiService
  - Properties
  - 증속성
  - appsettings.json
  - Program.cs
  - demo02.AppHost
  - demo02.NativePlugin
  - demo02.OpenApiPlugin
  - 증속성
  - Program.cs
  - demo02.ServiceDefaults
  - demo02.Web
- Bottom Status Bar:** 100%, 0/1, SPC, CRLF, 속성, 솔루션 탐색기, GitHub Copilot 재생, 오류 목록, 기호 찾기 결과, 즐겨 찾기, 79, main, 오후 8:38, 2024-11-21

그러면, 프롬프트와 닷넷 코드를 동시에 호출하는 것이 가능한 건가요?



G3J 12개 세션 중  
가장 인기 있는 2위!

**Session 2.**  
**SK 기본 작동법**



# Agenda

01

Semantic Kernel  
인스턴스 생성하기

02

Semantic Kernel  
인스턴스 연동시키기

03

Semantic Kernel  
인스턴스에 다양한  
LLM 연결하기





A screenshot of a .NET Interactive Jupyter Notebook titled "model-gpt4o-20240806". The notebook interface includes a menu bar (Code, File, Edit, Selection, View, Go, Run, Terminal, Window, Help), a toolbar with icons for selection, run, terminal, and help, and a status bar at the bottom.

The main area displays three code cells:

- Cell 1:** Prints the current version of the Semantic Kernel.
- Cell 2:** Gets Azure OpenAI access details.
- Cell 3:** Builds a Semantic Kernel using the Azure OpenAI Chat Completion provider.

The "OPEN EDITORS" sidebar shows one unsaved file: "demo01.ipynb". The "SOLUTION EXPLORER" sidebar lists the project structure:

- G3J-LEARN-SK
  - ep01
  - ep02
    - config
    - demo01.apiapp
    - demo01.consoleapp
  - demo01.ipynb
  - demo02.ipynb
- README.md
- demo01.sln
- ep03
- ep04
- ep05



demo02.ipynb

에서 여러 LLM 사용해보기 > Google Gemini 모델 사용 > var prompt = await Microsoft.DotNet.Interactive.Kernel.GetInputAsync("Prompt");

+ 코드 + Markdown | ▶ 모두 실행 ⏪ 다시 시작 ⏴ 출력 모두 지우기 | 변수 ⏴ 개요 ... .NET Interactive

```
var prompt = await Microsoft.DotNet.Interactive.Kernel.GetInputAsync("Prompt");

var result = await kernel.InvokePromptAsync(prompt);

Console.WriteLine(result);
```

[57] ✓ 9.2s

... 저는 구글에서 훈련된 대규모 언어 모델입니다.

csharp - C# Script Code

## Hugging Face 모델 사용

Semantic Kernel로 Hugging Face의 다양한 모델을 사용할 수 있습니다.

Microsoft.SemanticKernel.Connectors.HuggingFace 누겟 패키지 설치가 필요하며 Kernel을 빌드할 때

g3j-learn-sk

ep02/demo2-github-models\*

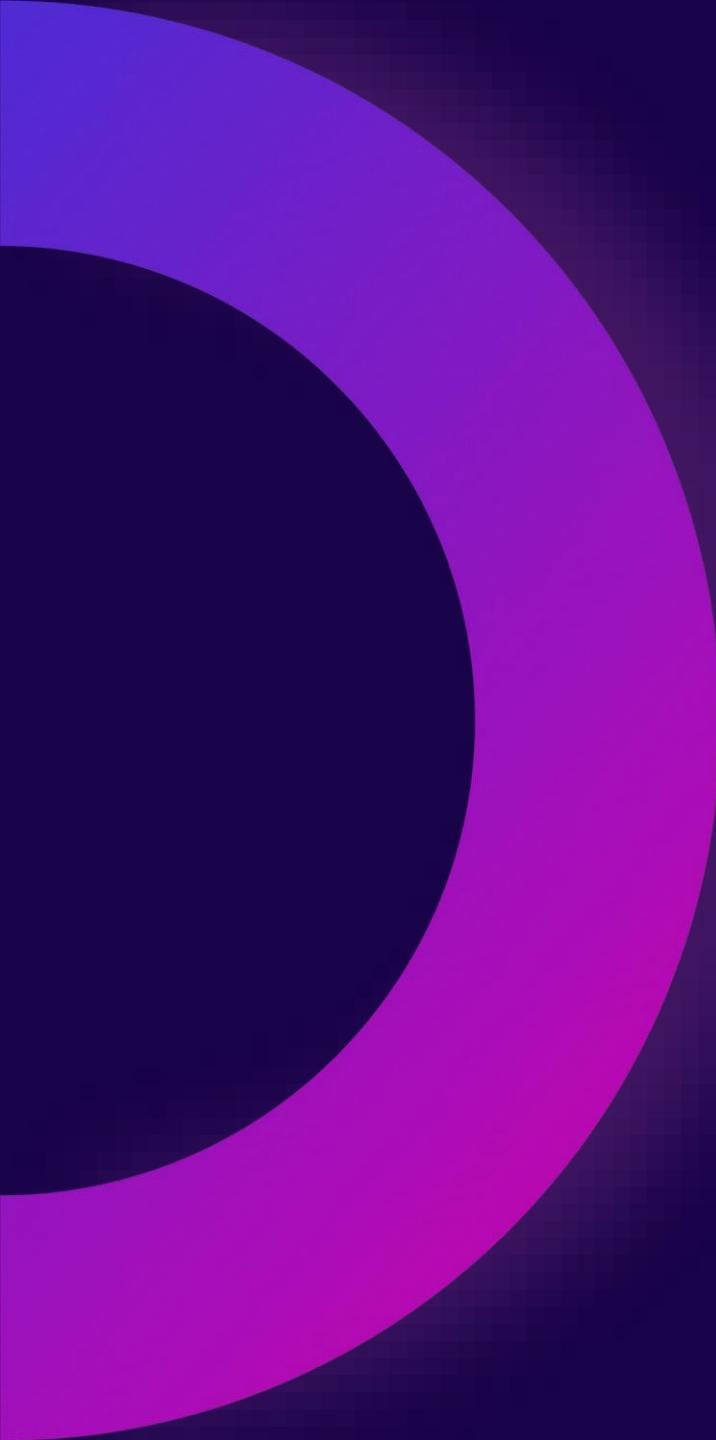
Spaces: 4 CRLF 셀 9/19

9°C 구름 조금

검색

Spaces: 4 CRLF 셀 9/19

오후 8:21 2024-11-07



G3J 12개 세션 중  
가장 인기 있는 1위!

**Session 5.**  
**SK 멀티 에이전트**





박구삼 | Microsoft MVP

# Agenda

01

Agent 소개

02

Agent 만들기

03

Agent 협업하기





출처: 언커버 더 스모킹 건, 스팀 상점페이지

Visual Studio IDE showing a C# code editor with a test runner interface. The code is for a class named `Demo2_03` which implements `ITestOutputHelper`. The code uses AI-generated text for menu reviews and developer instructions.

```
17     {
18         참조 0개 | Jongin Lee, 20시간 전 | 만든 이 1명, 변경 내용 1개
19         public class Demo2_03(ITestOutputHelper output) : BaseAgentsTest(output)
20     {
21         private const string ReviewerName = "MenuReviewer";
22         private const string ReviewerInstructions =
23             """
24             너는 메뉴 개발자가 개발한 메뉴를 리뷰하는 메뉴리뷰어입니다.
25             너는 새로운 메뉴가 식당의 이미지와 잘 어울리는지 판단하는 것이 목표입니다.
26             적합하다면 해당 메뉴가 승인되었다고 반드시 명시적으로 말하고 다음 새로운 메뉴를 제안해 달라고 하세요.
27             그렇지 않다면 예시 없이 개선 방법에 대한 인사이트를 제공하세요.
28             """
29
30         private const string MenuDeveloperName = "MenuDeveloper";
31         private const string MenuDeveloperInstructions =
32             """
33             너는 식당에게 식당의 상황, 특색에 맞게 최선의 메뉴를 개발하는 메뉴 개발자입니다.
34             너는 고객이 좋아할 만한 새로운 음식을 생각해냅니다.
35             너는 식당의 분위기와 고객의 취향에 맞게 다양한 음식을 조합하여 메뉴를 구성합니다.
36             한 번에 하나의 메뉴만 제공합니다.
37             현재 목표에 집중합니다.
38             끝데없는 잡담으로 시간을 낭비하지 않습니다.
39             아이디어를 개선할 때 제안을 고려합니다.
40             """
41
42         [Fact]
43         ● | 참조 0개 | Jongin Lee, 20시간 전 | 만든 이 1명, 변경 내용 1개
44         public async Task UseKernelFunctionStrategiesWithAgentGroupChatAsync()
45     {
46         // 에이전트 정의
        ChatCompletionAgent agentReviewer =
```

박구삼 | Microsoft MVP

이종인 | Microsoft MVP Kernel

# 03

## 가이드 프로젝트 소개



# Repository 살펴보기!

G3J Learn SK 쇼 샘플 코드 리포지토리



# G3J 시리즈 콜렉션

이번 G3J Learn 시리즈는 콜렉션을 제공하고 있습니다.  
혹시라도 막히는 부분이 있거나 라이브 중 참고한 자료 및 다시보기를  
알고 싶다면 이 링크를 이용해주세요!

<https://aka.ms/g3jlearnsk>



# G3J 시리즈 샘플

이번 G3J Learn 시리즈는 라이브에 사용한 샘플을 제공하고 있습니다.  
혹시라도 궁금한 부분이 있거나 라이브 중 소개한 코드를 알고 싶다면  
이 링크를 이용해주세요!

<https://aka.ms/g3jlearnsk/samples>



Thank you

