



Intelligent Agents

Agenda:

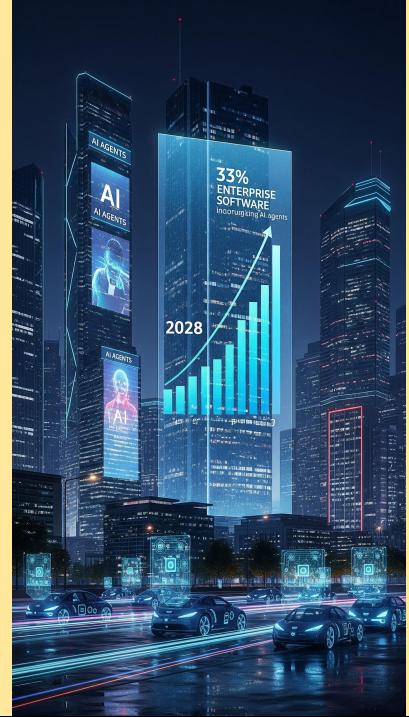
1. AI Agents Summary
2. AI Agents Hands-on
 - o Create Agent
 - o Create Agent with Tool
 - o Create Parallel Agent
3. Questions

AI Agents Summary

By 2028

33%

Enterprise software will incorporate AI agents



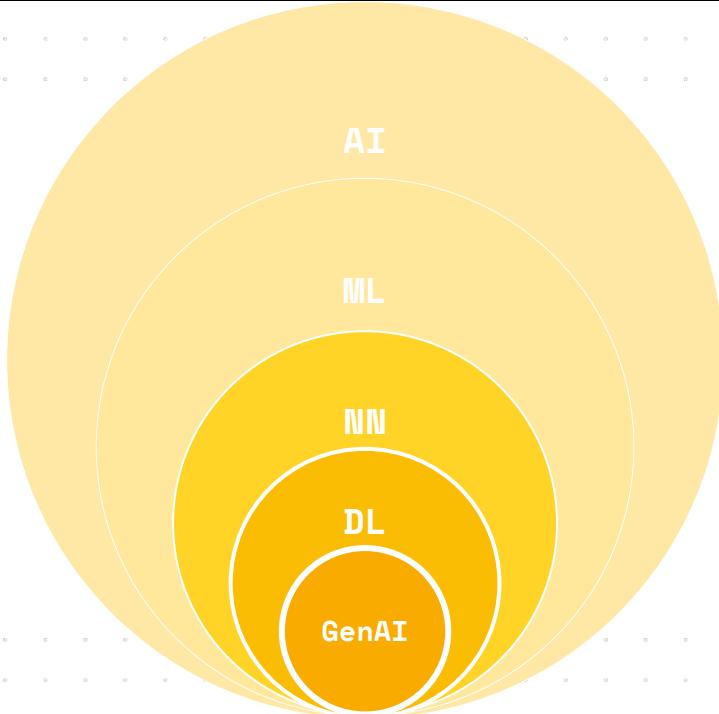
By 2028

15%

AI agents will automate daily work



AI to Gen AI



The most common definition of Artificial Intelligence (AI) is:

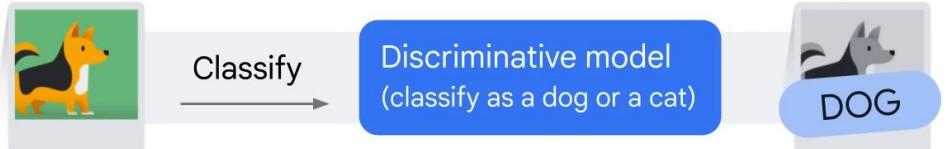
- **AI** is the capability of a computer system to perform tasks typically associated with human intelligence,
 - such as learning, reasoning, problem-solving, perception, and decision-making.
- It is a broad field of computer science focused on creating smart machines that can mimic human cognitive functions.

This diagram places GenAI within the broader context of the other fields.

- GenAI is a subset of Deep Learning (deep neural networks).
- Gen AI is a type of AI that can produce new content, including text, images, audio, and synthetic data.

Generative vs. Non-Generative Models

Discriminative technique



Generative technique



There is a fundamental distinction in ML models.

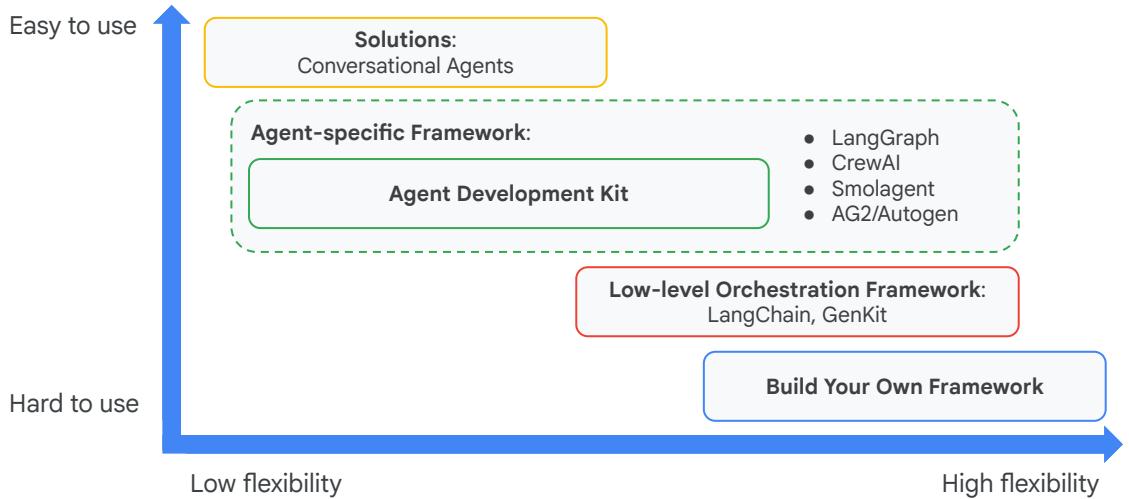
- **Discriminative models** are designed to classify or differentiate between data points, while
- **Generative models** are designed to create new, synthetic data.

This diagram visually explains the difference.

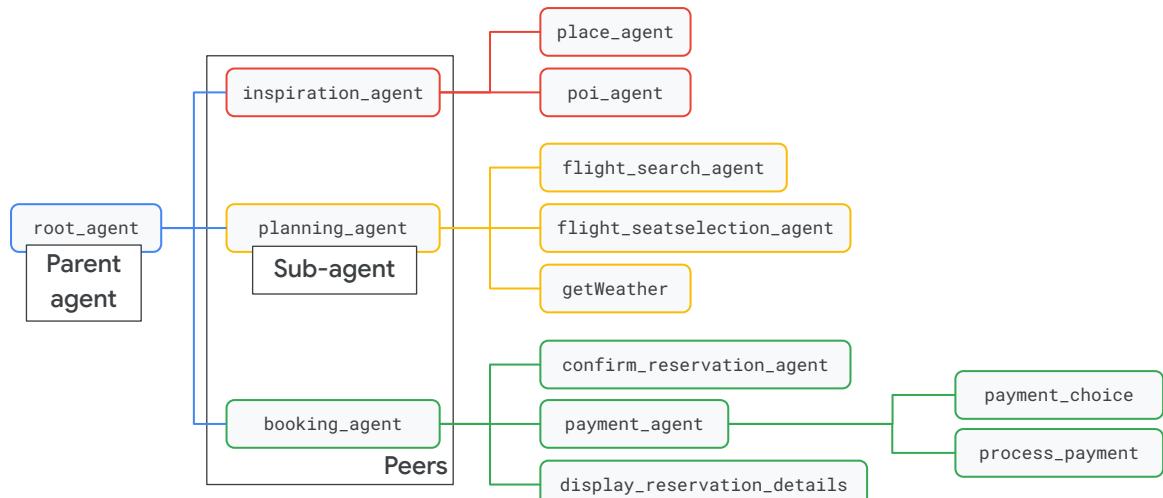
- A **discriminative model** learns the boundary between two classes, while
- a **generative model** learns the underlying distribution of each class, allowing it to generate new data for that class.

A **discriminative model** asks, "What is the boundary between cats and dogs?" while a **generative model** asks, "What is a cat?" or "What is a dog?"

AI Agent Framework



Multi-agent Systems



Google Cloud

The key idea with multi-agent systems in ADK is that there is a structure of parent and sub-agents constructed in a tree. Agents can only transfer the conversation to sub-agents, back to a parent, or to a peer (another agent that shares the same parent). You can disable transferring to peers on an agent-by-agent basis.

This tree structure makes conversations more predictable and reliable. As an alternative, if you could transfer to any agent, and two agents in different parts of your system were each responsible for certain lookup activities, it might be very easy to call the wrong agent. With this approach, only the lookup agent related to your part of the conversation will be invoked.

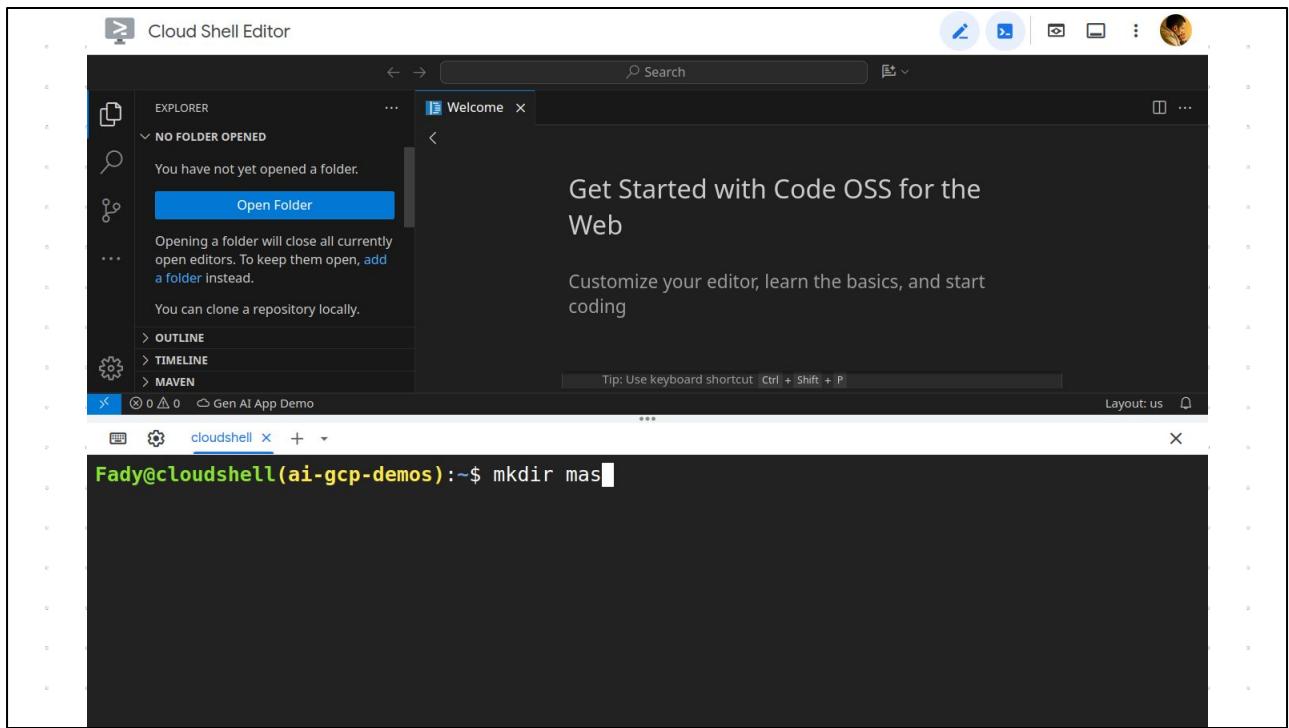


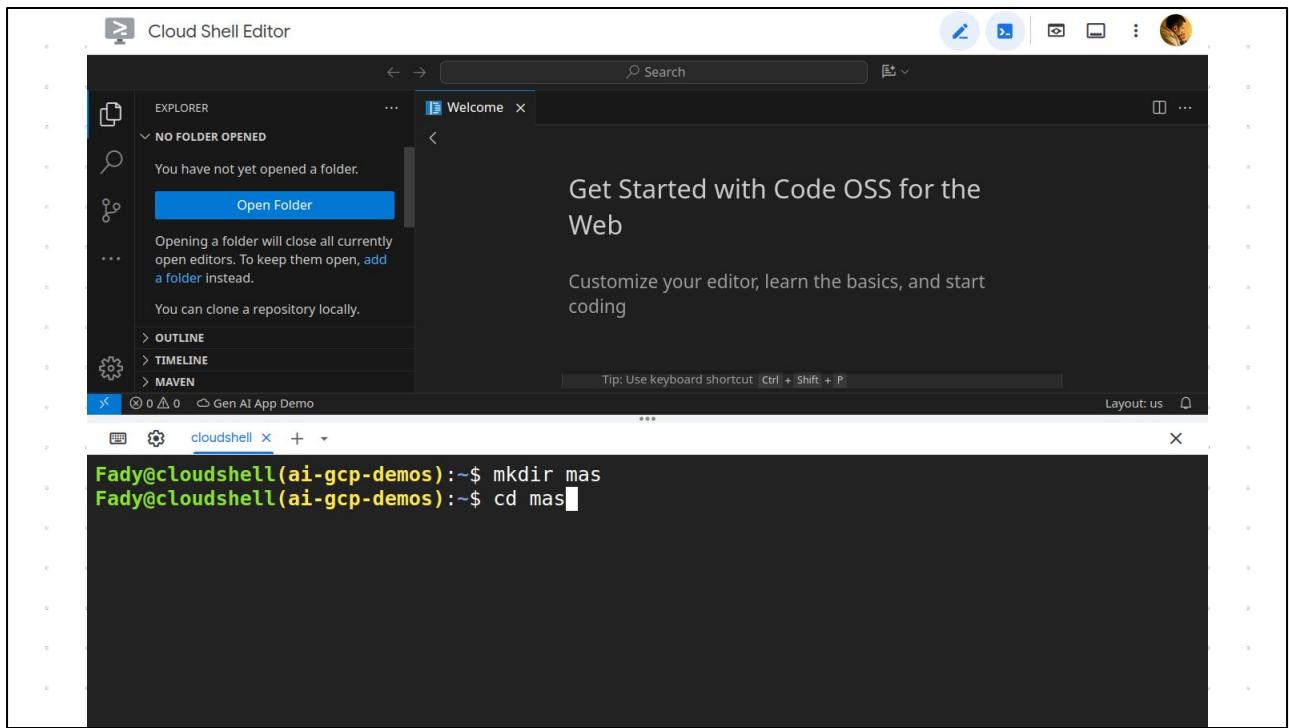
A presentation slide with a white background and a green border. The title 'AI Agents Hands-on' is centered in blue text.

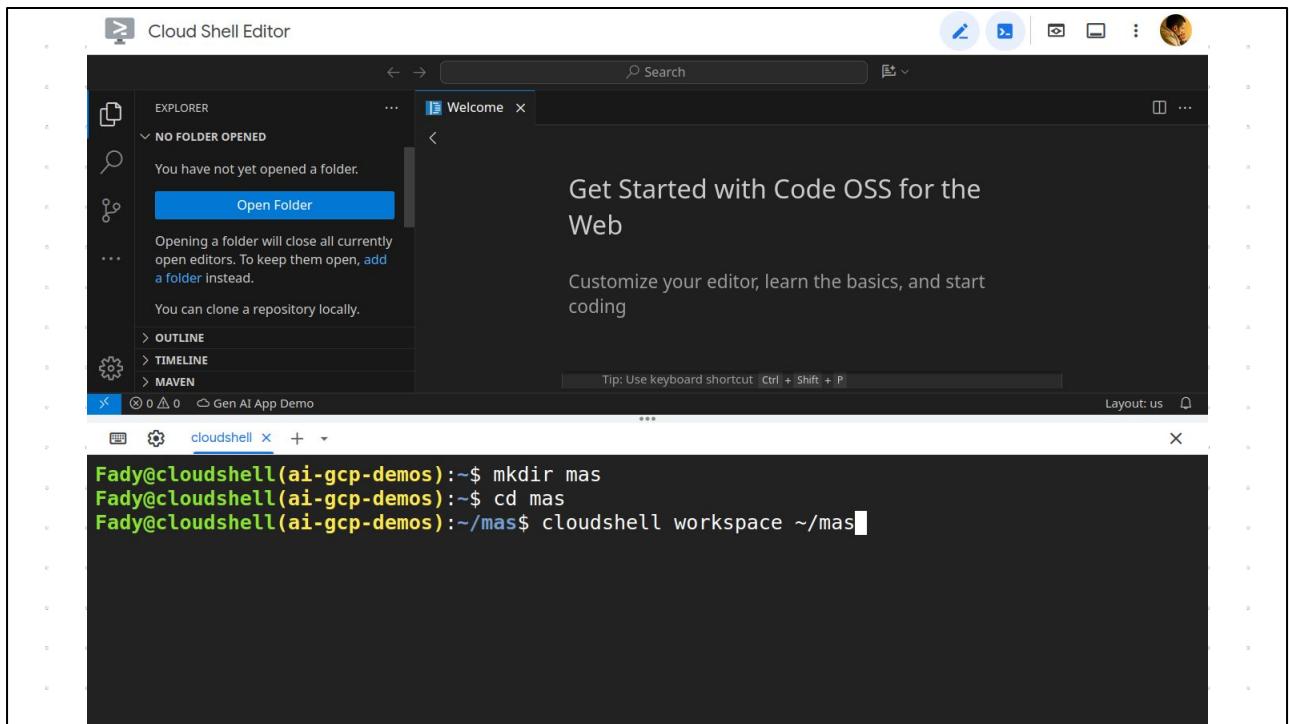
AI Agents Hands-on

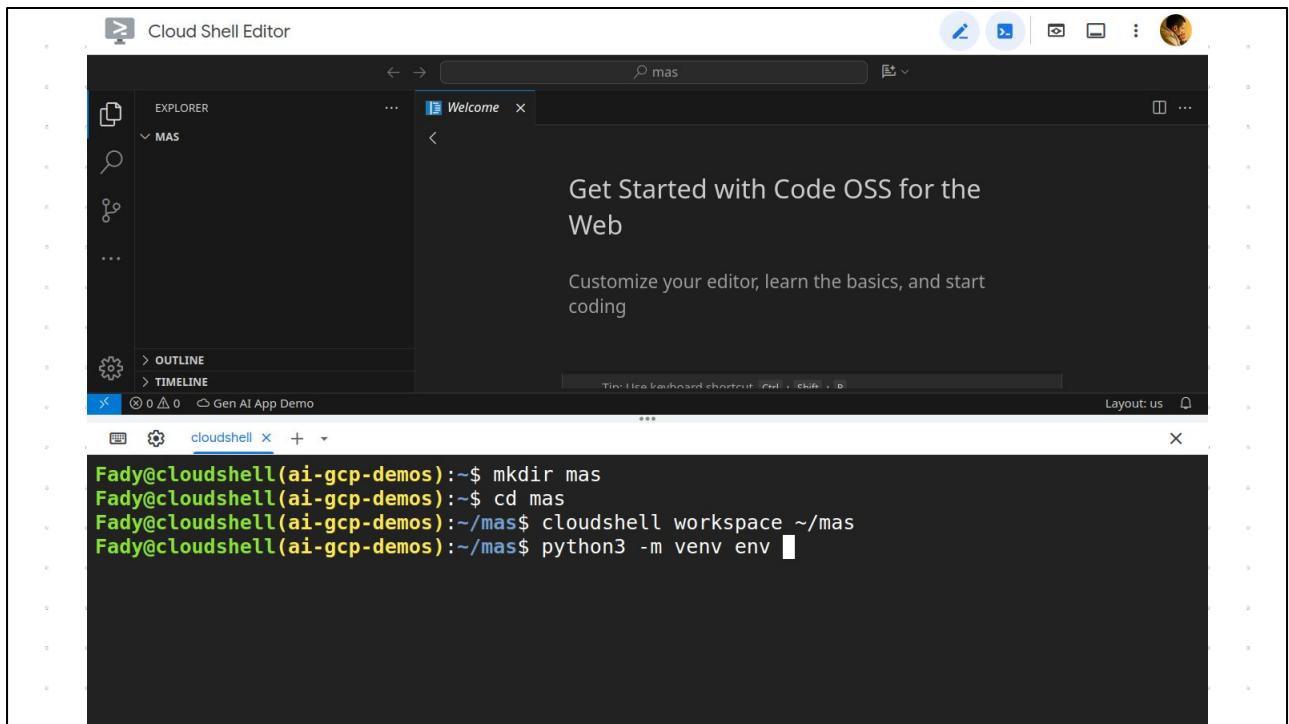
Go to Cloud Shell

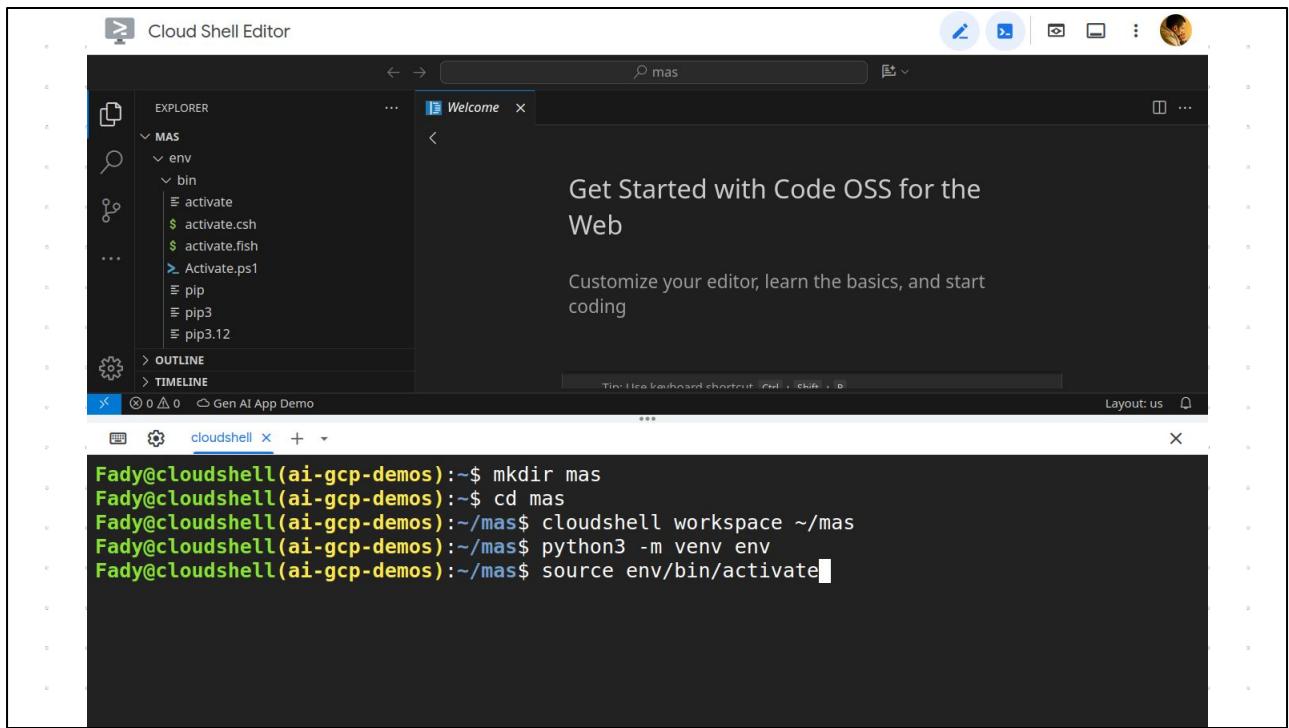
<https://shell.cloud.google.com>











The screenshot shows the Cloud Shell Editor interface. At the top, there's a header bar with icons for file operations like save, copy, and paste, along with a user profile picture. Below the header is the Explorer sidebar, which displays a file structure under the 'MAS' folder. The 'bin' directory contains several files: activate, activate.csh, activate.fish, Activate.ps1, pip, pip3, and pip3.12. There are also 'OUTLINE' and 'TIMELINE' sections. The main workspace is titled 'Welcome' and features a dark theme with white text. It includes a heading 'Get Started with Code OSS for the Web' and a sub-instruction 'Customize your editor, learn the basics, and start coding'. A keyboard shortcut tip is visible at the bottom of the workspace area. At the bottom of the screen, there's a navigation bar with tabs for 'cloudshell' and a '+' icon, along with a 'Layout: us' indicator.

```
Fady@cloudshell(ai-gcp-demos):~$ mkdir mas
Fady@cloudshell(ai-gcp-demos):~$ cd mas
Fady@cloudshell(ai-gcp-demos):~/mas$ cloudshell workspace ~/mas
Fady@cloudshell(ai-gcp-demos):~/mas$ python3 -m venv env
Fady@cloudshell(ai-gcp-demos):~/mas$ source env/bin/activate
(env) Fady@cloudshell(ai-gcp-demos):~/mas$ pip install google-adk
```

The screenshot shows the Cloud Shell Editor interface. The top navigation bar includes icons for file operations, search, and user profile. The left sidebar has sections for EXPLORER, OUTLINE, and TIMELINE, with the EXPLORER section currently active and showing a directory tree for a project named 'MAS'. The main area displays the 'Welcome' tab, which features a title 'Get Started with Code OSS for the Web' and a subtitle 'Customize your editor, learn the basics, and start coding'. Below this is a keyboard shortcut tip. The bottom panel is a terminal window titled 'cloudshell' showing pip dependency output:

```
y-resourceDetector-gcp-1.11.0a0 opentelemetry-sdk-1.37.0 opentelemetry-semantic-convention s-0.58b0 packaging-25.0 proto-plus-1.26.1 protobuf-6.33.1 pyarrow-22.0.0 pyasn1-0.6.1 pyasn1-modules-0.4.2 pycparser-2.23 pydantic-2.12.4 pydantic-core-2.41.5 pydantic-settings-2.1.2.0 pyjwt-2.10.1 pyparsing-3.2.5 python-dateutil-2.9.0.post0 python-dotenv-1.2.1 python-multipart-0.0.20 referencing-0.37.0 requests-2.32.5 rpds-py-0.29.0 rsa-4.9.1 shapely-2.1.2 s ix-1.17.0 sniffio-1.3.1 sqlalchemy-2.0.44 sqlalchemy-spanner-1.17.1 sqlparse-0.5.3 sse-starlette-3.0.3 starlette-0.48.0 tenacity-9.1.2 typing-extensions-4.15.0 typing-inspection-0.4.2 tzlocal-5.3.1 uritemplate-4.2.0 urllib3-2.5.0 uvicorn-0.38.0 watchdog-6.0.0 websockets -15.0.1 zipp-3.23.0  
(env) Fady@cloudshell(ai-gcp-demos):~/mas$ pip show google-adk
```

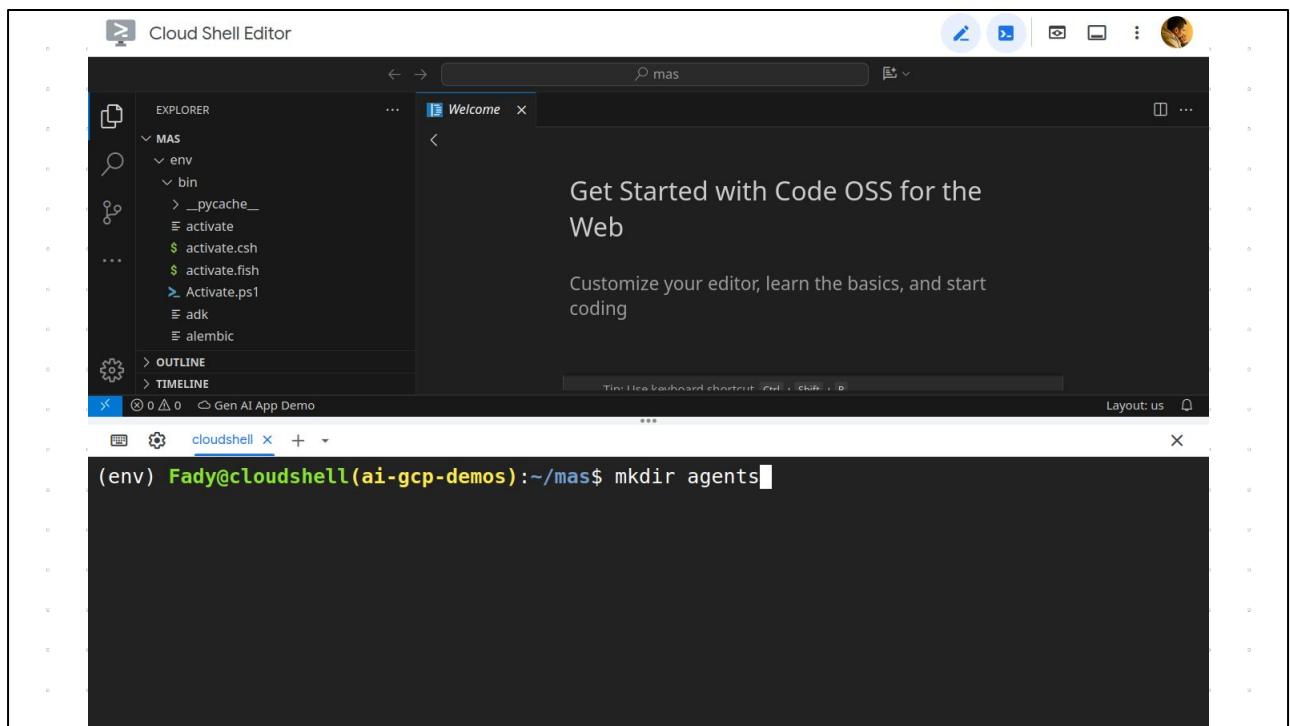
Cloud Shell Editor

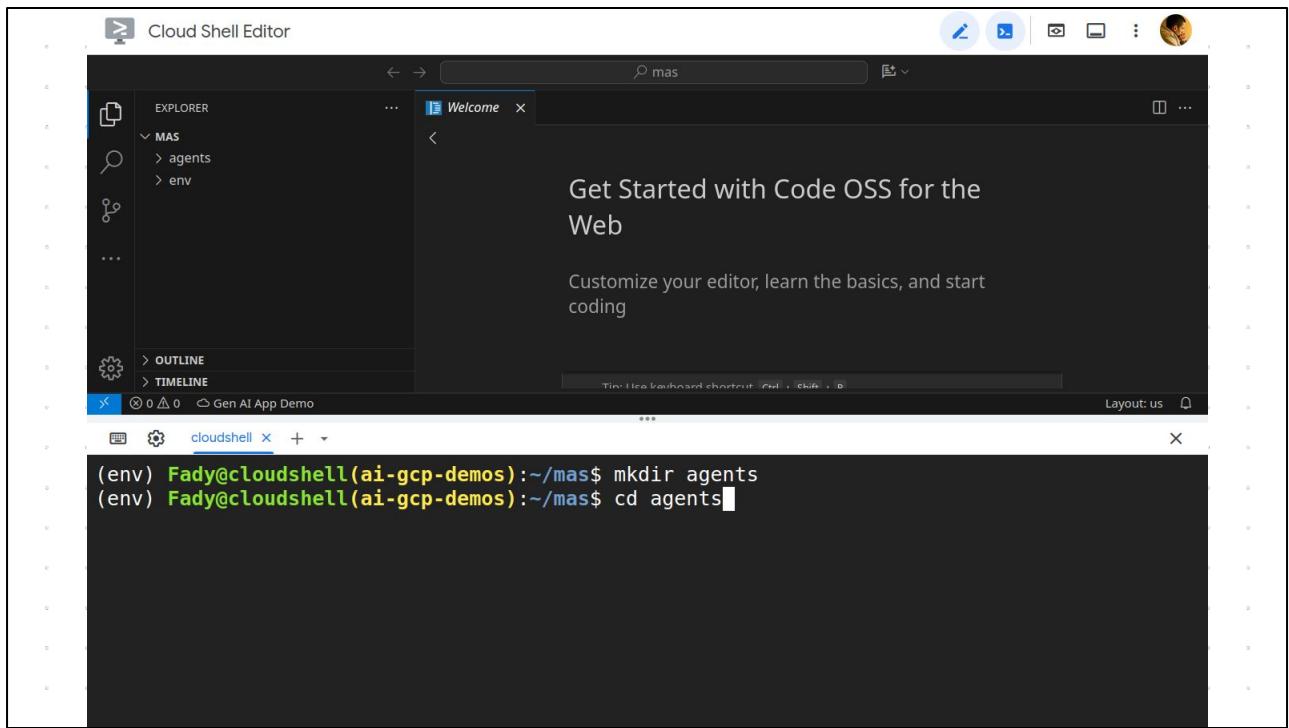
EXPLORER Welcome

(env) Fady@cloudshell(ai-gcp-demos):~/mas\$ pip show google-adk

```
Name: google-adk
Version: 1.19.0
Summary: Agent Development Kit
Home-page:
Author:
Author-email: Google LLC <googleapis-packages@google.com>
License:
Location: /home/fady_a_m_ibrahim/mas/env/lib/python3.12/site-packages
Requires: aiosqlite, aioio, authlib, click, fastapi, google-api-python-client, google-cloud-aiplatform, google-cloud-bigquery, google-cloud-bigquery-storage, google-cloud-bigtable, google-cloud-discoveryengine, google-cloud-secret-manager, google-cloud-spanner, google-cloud-speech, google-cloud-storage, google-genai, graphviz, jsonschema, mcp, opentelemetry-api, opentelemetry-exporter-gcp-logging, opentelemetry-exporter-gcp-monitoring, opentelemetry-exporter-gcp-trace, opentelemetry-exporter-otlp-proto-http, opentelemetry-resourcedetector-gcp, opentelemetry-sdk, pyarrow, pydantic, python-dateutil, python-dotenv, PyYAML, requests, sqlalchemy, sqlalchemy-spanner, starlette, tenacity, typing-extensions, tzlocal, uvicorn, watchdog, websockets
Required-by:
```

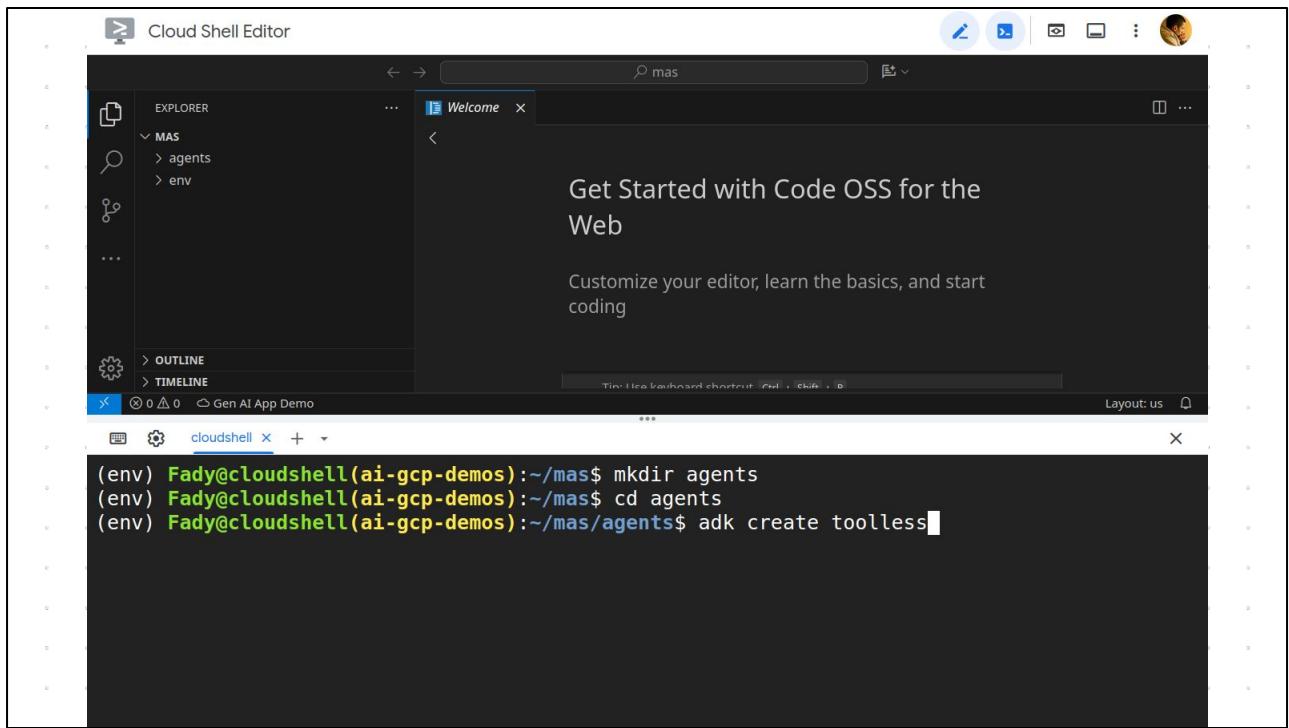
(env) Fady@cloudshell(ai-gcp-demos):~/mas\$

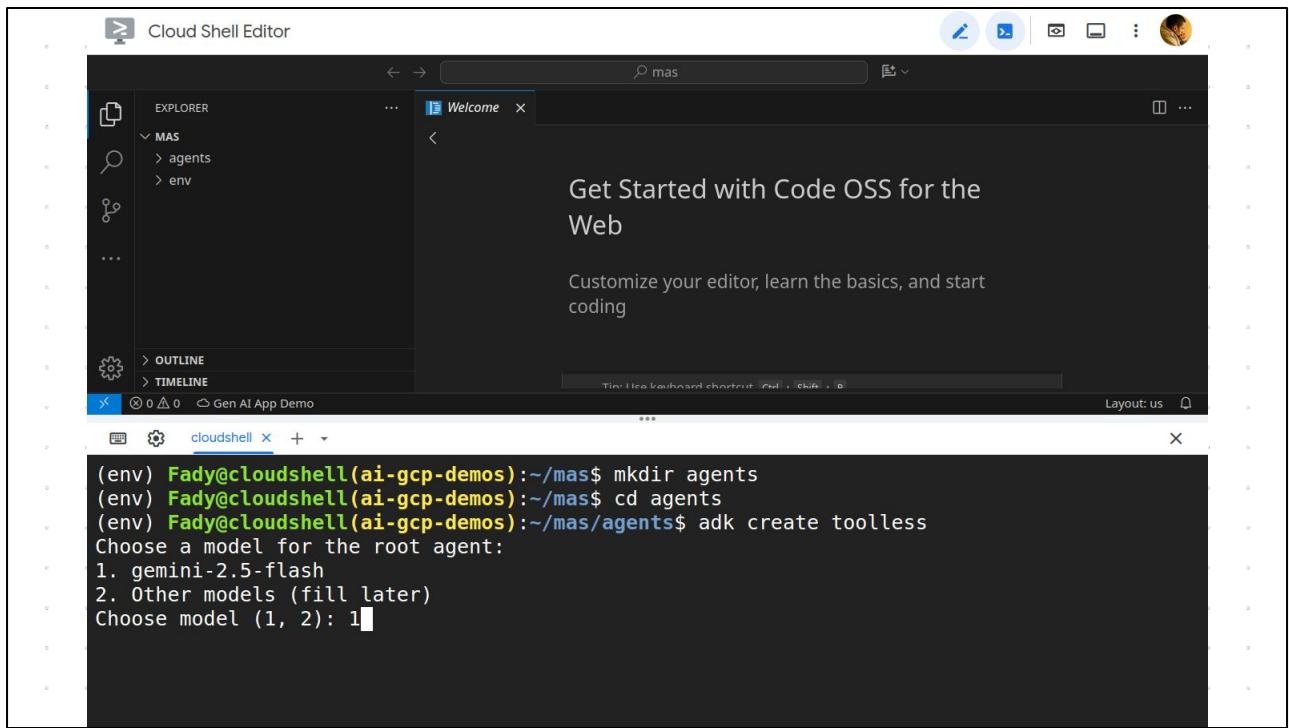






Create Agent





The screenshot shows the Cloud Shell Editor interface. The top bar has a 'Cloud Shell Editor' title and various icons. The left sidebar includes an 'EXPLORER' section with 'MAS' expanded, showing 'agents' and 'env'. Below it are 'OUTLINE' and 'TIMELINE' sections. The main area has a 'Welcome' tab open, displaying the message 'Get Started with Code OSS for the Web' and 'Customize your editor, learn the basics, and start coding'. A status bar at the bottom shows keyboard shortcuts like 'Ctrl + Shift + D' and 'Layout: us'. The bottom half of the screen is a terminal window titled 'cloudshell'. It shows the following command sequence:

```
(env) Fady@cloudshell(ai-gcp-demos):~/mas$ mkdir agents
(env) Fady@cloudshell(ai-gcp-demos):~/mas$ cd agents
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ adk create toolless
Choose a model for the root agent:
1. gemini-2.5-flash
2. Other models (fill later)
Choose model (1, 2): 1
1. Google AI
2. Vertex AI
Choose a backend (1, 2): 1
```

We can choose Vertex AI when working in production.
And we can choose Google AI when developing or creating a demo.

The screenshot shows the Cloud Shell Editor interface. At the top, there's a navigation bar with icons for file operations like Open, Save, and Print. Below the bar is a search field containing 'mas'. The main area has a dark theme with a sidebar on the left labeled 'EXPLORER' which includes sections for 'MAS' (with 'agents' and 'env' sub-items), 'OUTLINE', and 'TIMELINE'. A tab bar at the bottom shows 'cloudshell' is active. The central workspace displays a 'Welcome' page with the title 'Get Started with Code OSS for the Web' and a subtitle 'Customize your editor, learn the basics, and start coding'. Below this, a terminal window is open with the following content:

```
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ adk create toolless
Choose a model for the root agent:
1. gemini-2.5-flash
2. Other models (fill later)
Choose model (1, 2): 1
1. Google AI
2. Vertex AI
Choose a backend (1, 2): 1

Don't have API Key? Create one in AI Studio: https://aistudio.google.com/apikey

Enter Google API key: █
```

With Google AI, we need to provide an API key.
We can click on the link to open AI Studio to get the key.

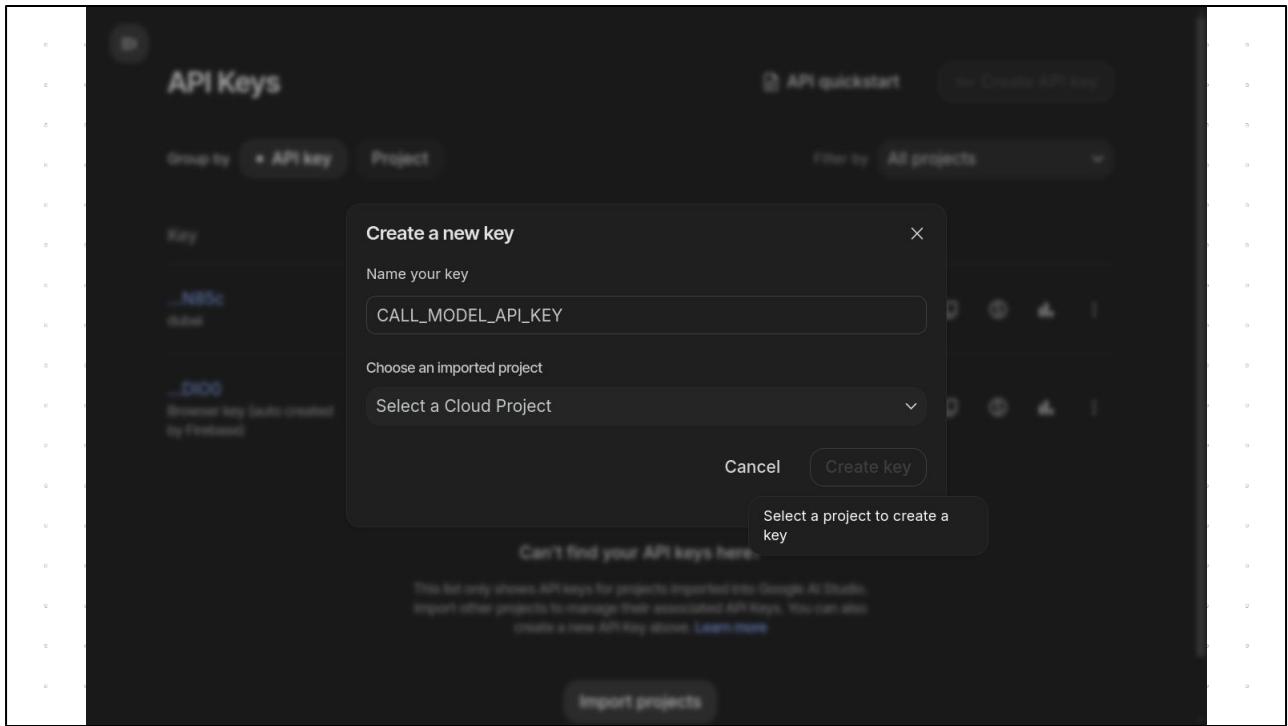
The screenshot shows the 'API Keys' page in Google AI Studio. At the top, there are navigation links for 'API quickstart' and 'Create API key'. Below that, there are filters for 'Group by' (set to 'API key') and 'Project' (set to 'All projects'), and a dropdown for 'Filter by' also set to 'All projects'. The main table lists two API keys:

Key	Project	Created on	Quota tier
...N85c dubai	AI-GCP-demos ai-gcp-demos	Nov 9, 2025	Upgrade Tier 1
...DIO0 Browser key (auto created by Firebase)	AI-GCP-demos ai-gcp-demos	Jun 30, 2023	Upgrade Tier 1

For each key, there are three icons: a copy icon, a settings icon, and a more options icon. A tooltip 'Copy API key' is shown over the first icon of the second row. Below the table, there's a message: 'Can't find your API keys here? This list only shows API keys for projects imported into Google AI Studio. Import other projects to manage their associated API Keys. You can also create a new API Key above. [Learn more](#)'.

[Import projects](#)

Either we can copy an existing API key (if one exists).



Or create a new one.

The screenshot shows the Cloud Shell Editor interface. The left sidebar has a dark theme with icons for file operations like copy, paste, search, and refresh. The main area has a title bar "Welcome" with a search bar containing "mas". Below the title bar is a section titled "Get Started with Code OSS for the Web" with the sub-instruction "Customize your editor, learn the basics, and start coding". A tip at the bottom says "Tip: Use keyboard shortcut Ctrl + Shift + P". The bottom half of the screen is a terminal window titled "cloudshell" with the command "(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents\$ edit toolless/.env" and its output: "Agent created in /home/fady_a_m_ibrahim/mas/agents/toolless: - .env - __init__.py - agent.py".

We will paste the key in the prompt and press Enter.
After that, the three files shown in the above screenshot are created.

The files are:

1. .env
2. __init__.py
3. agent.py

Cloud Shell Editor

EXPLORER ... Welcome .env x

agents > toolless > .env

1 GOOGLE_GENAI_USE_VERTEXAI=0
2 GOOGLE_API_KEY=<THE_API_KEY_VALUE_IS_HERE>

> OUTLINE
> TIMELINE

× 0 △ 0 ⌂ Gen AI App Demo

Ln 2, Col 43 Spaces: 4 UTF-8 LF () Properties Layout: us

cloudshell x + *

```
Agent created in /home/fady_a_m_ibrahim/mas/agents/toolless:  
- .env  
- __init__.py  
- agent.py  
  
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit toolless/.env  
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$
```

Cloud Shell Editor

EXPLORER

MAS

agents/toolless

 __init__.py

 .env

 agent.py

 env

OUTLINE

TIMELINE

Welcome .env __init__.py

agents > toolless > __init__.py > {} agent

1 from . import agent

2

Ln 1, Col 1 Spaces: 4 UTF-8 LF {} Python 3.12.3 ('env': venv) Layout: us

cloudshell

- .env
- __init__.py
- agent.py

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents\$ edit toolless/.env
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents\$ edit toolless/__init__.py
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents\$

Cloud Shell Editor

EXPLORER ...

agents > toolless > agent.py > ...

>Welcome .env __init__.py agent.py

agent.py

1 from google.adk.agents.llm_agent import Agent
2
3 root_agent = Agent(
4 model='gemini-2.5-flash',
5 name='root_agent',
6 description='A helpful assistant for user questions.',
7 instruction='Answer user questions to the best of your knowledge',
8)
9

Ln 9, Col 1 Spaces: 4 UTF-8 LF {} Python 3.12.3 ('env': venv) Layout: us

> OUTLINE > TIMELINE

cloudshell

- __init__.py
- agent.py

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents\$ edit toolless/.env
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents\$ edit toolless/__init__.py
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents\$ edit toolless/agent.py
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents\$

ADK Repo

<https://github.com/google/adk-python>

google / adk-python

Code Issues Pull requests Discussions Actions Projects Security ...

adk-python Public

Watch 136 Fork 2.5k Star 15.7k

main Go to file + <> Code

GWeale and copybara-github feat: expose servic... docs: Add AGENTS.md to provi... ci: bump action scripts versions Update the adk-web-dev-ui-fu... docs: Remove the list_unlabe... fix: Add example and fix for lo... feat: expose service URI flags feat: expose service URI flags

feat: expose service URI flags 14 hours ago 5 months ago 5 days ago 6 months ago yesterday last month 14 hours ago 14 hours ago

<https://github.com/google/adk-python/tree/main/src/google/adk>

About

An open-source, code-first Python toolkit for building, evaluating, and deploying sophisticated AI agents with flexibility and control.

google.github.io/adk-docs/

agent ai multi-agent
agents multi-agent-systems
ai-agents multi-agents ilm
ilm genai agentic
genai-chatbot agentic-ai
multi-agents-collaboration
aiagentframework agents-sdk

google / adk-python

Code Issues Pull requests Discussions Actions Projects Security ...

main ↗ main · adk-python / src / google / adk / ⌂ Go to file Add file ...

GWeale and copybara-github feat: expose service URI flags f283027 · 14 hours ago

Name	Last commit message	Last commit date
..		
a2a	fix: Let part converters also return multiple parts so they...	2 weeks ago
agents	feat(agents): Add warning for duplicate sub-agent names	yesterday
apps	docs: too many E(inv=2, role=user) plus reformatting	5 days ago
artifacts	fix: Remove app name from FileArtifactService directory ...	last week
auth	ADK changes	3 weeks ago
cli	feat: expose service URI flags	14 hours ago
code_executors	feat: Capture thinking output, forward raw payloads, an...	5 days ago

The screenshot shows a code editor interface with a dark theme. On the left is the main code editor window, and on the right is a sidebar titled "Symbols".

Main Editor (Llm_agent.py):

```
815     @_model_validator(mode='after')
816     def __model_validator_after__(self) -> LlmAgent:
817         return self
818
819     @field_validator('generate_content_config', mode='after')
820     @classmethod
821     > def validate_generate_content_config( ... )
822         return generate_content_config
823
824     @classmethod
825     @experimental
826     > def _resolve_tools( ... )
827         return resolved_tools
828
829     @override
830     @classmethod
831     @experimental
832     > def _parse_config( ... )
833         return kwargs
834
835
836     Agent: TypeAlias = LlmAgent
```

Symbols Sidebar:

Find definitions and references for functions and other symbols in this file by clicking a symbol below or in the code.

Filter symbols

- const logger
- const __SingleBeforeModelCallback
- const BeforeModelCallback
- const __SingleAfterModelCallback
- const AfterModelCallback
- const __SingleOnModelErrorCallback
- const OnModelErrorCallback
- const __SingleBeforeToolCallback
- const BeforeToolCallback
- const __SingleAfterToolCallback
- const AfterToolCallback
- const __SingleOnModelErrorCallback

main ↗ main / [adk-python](#) / src / google / adk / agents / [_init_.py](#)

Code Blame Raw ⌂ ⌄ ⌅ ⌆ ⌇

```
4 # you may not use this file except in compliance with the Licens
5 # You may obtain a copy of the License at
6 #
7 #     http://www.apache.org/licenses/LICENSE-2.0
8 #
9 # Unless required by applicable law or agreed to in writing, sof
10 # distributed under the License is distributed on an "AS IS" BAS
11 # WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express o
12 # See the License for the specific language governing permission
13 # limitations under the License.
14
15 import logging
16 import sys
17
18 from .base_agent import BaseAgent
19 from .invocation_context import InvocationContext
20 from .live_request_queue import LiveRequest
21 from .live_request_queue import LiveRequestQueue
22 from .llm_agent import Agent
23 from .llm_agent import LlmAgent
24 from .loop_agent import LoopAgent
25 from .parallel_agent import ParallelAgent
26 from .run_config import RunConfig
27 from .sequential_agent import SequentialAgent
28
```

Symbols X

Find definitions and references for functions and other symbols in this file by clicking a symbol below or in the code.

Filter symbols r

const _all_

Cloud Shell Editor

EXPLORER ...

MAS

agents / toolless

 agent.py

 init.py

 .env

agent.py

env

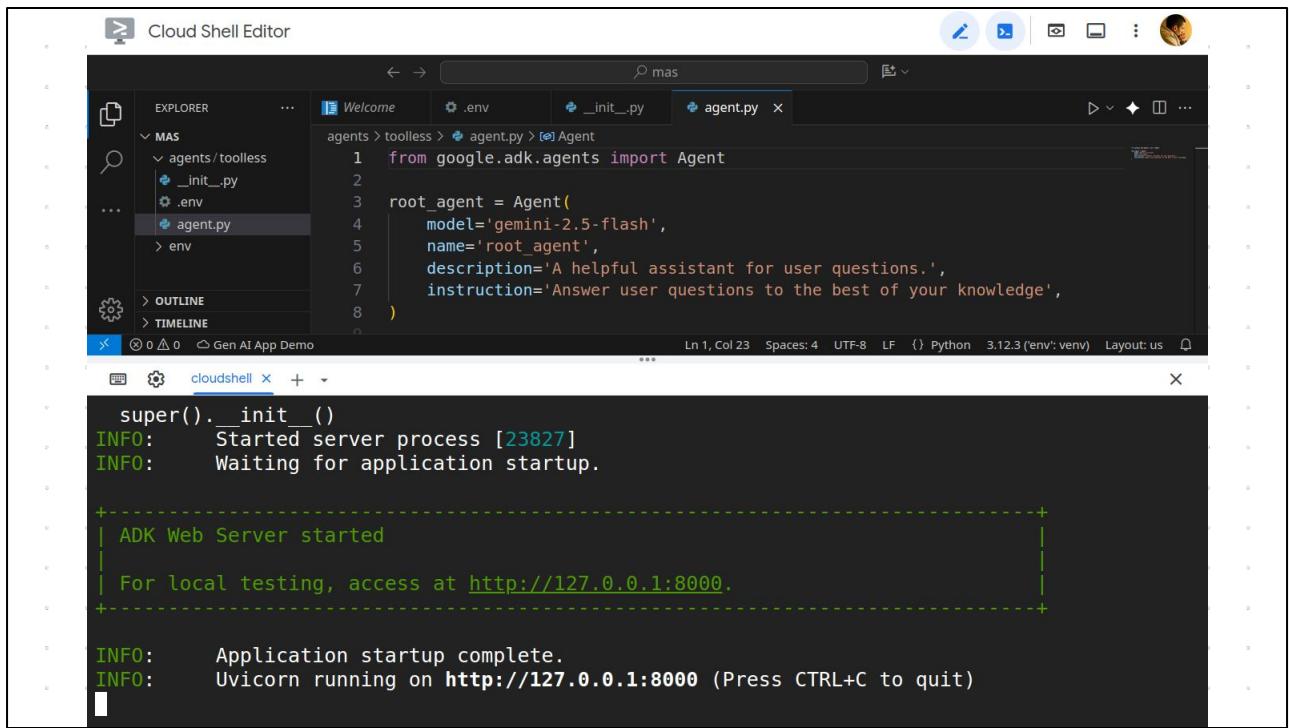
>Welcome .env __init__.py agent.py

agents > toolless > agent.py > Agent

```
1 from google.adk.agents import Agent
2
3 root_agent = Agent(
4     model='gemini-2.5-flash',
5     name='root_agent',
6     description='A helpful assistant for user questions.',
7     instruction='Answer user questions to the best of your knowledge',
8 )
9
```

Ln 1, Col 23 Spaces: 4 UTF-8 LF {} Python 3.12.3 ('env': venv) Layout: us

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents\$ adk web



The screenshot shows the Cloud Shell Editor interface. In the top navigation bar, there are tabs for "Welcome", ".env", "__init__.py", and "agent.py". The "agent.py" tab is active, displaying Python code for creating an AI agent:

```
1 from google.adk.agents import Agent
2
3 root_agent = Agent(
4     model='gemini-2.5-flash',
5     name='root_agent',
6     description='A helpful assistant for user questions.',
7     instruction='Answer user questions to the best of your knowledge',
8 )
```

Below the code editor, the status bar shows "Ln 1, Col 23" and "Python 3.12.3 (env: venv)".

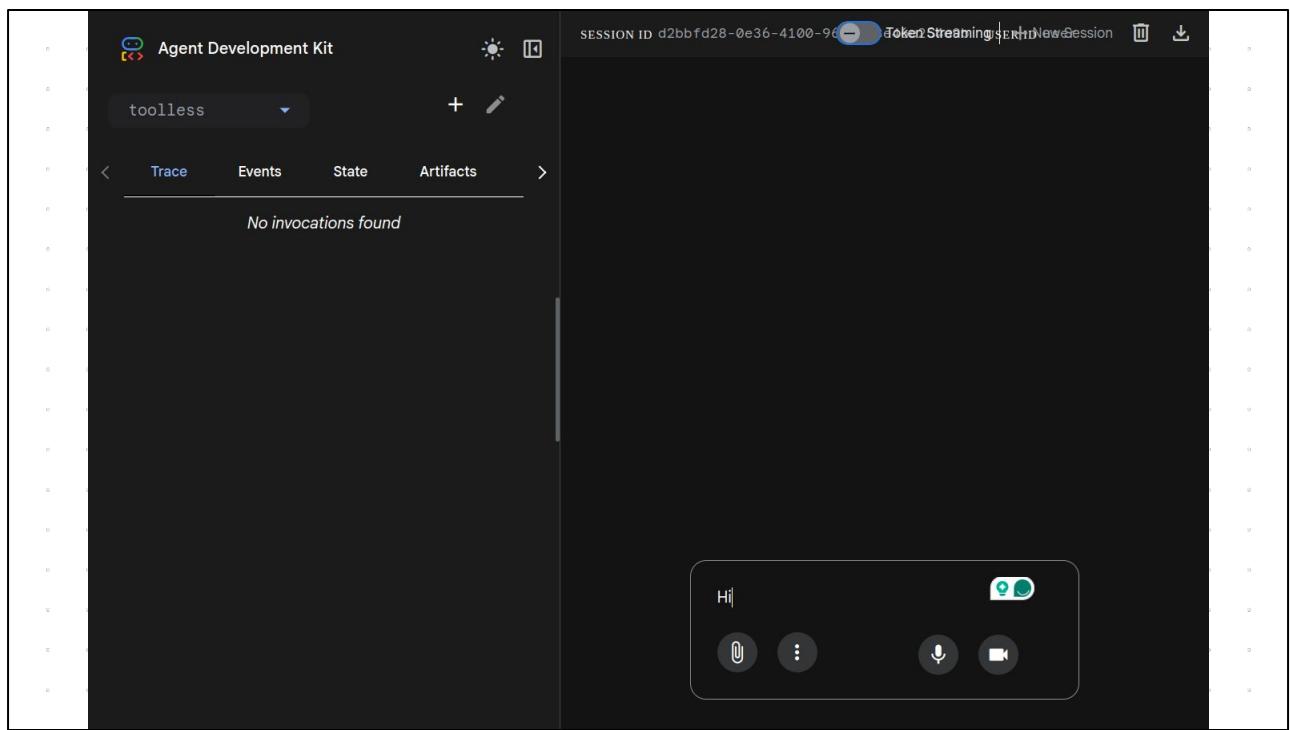
In the bottom panel, there is a terminal window titled "cloudshell" showing application startup logs:

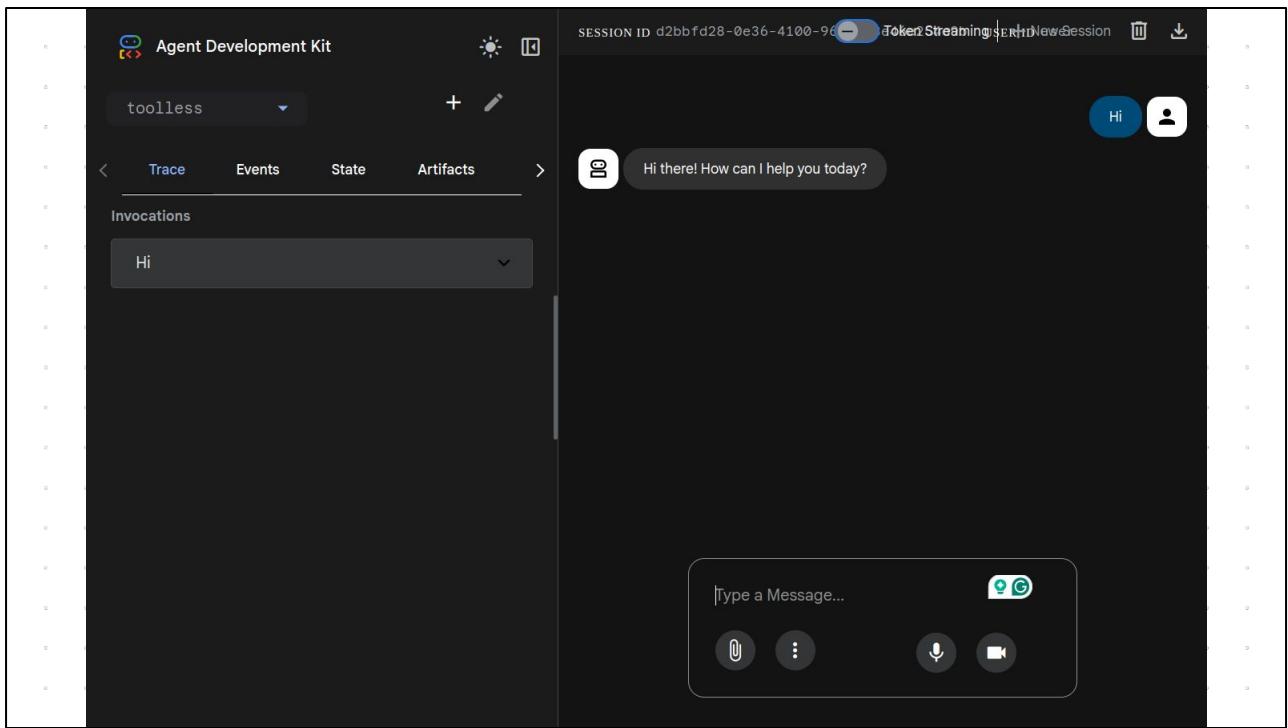
```
super().__init__()
INFO:     Started server process [23827]
INFO:     Waiting for application startup.

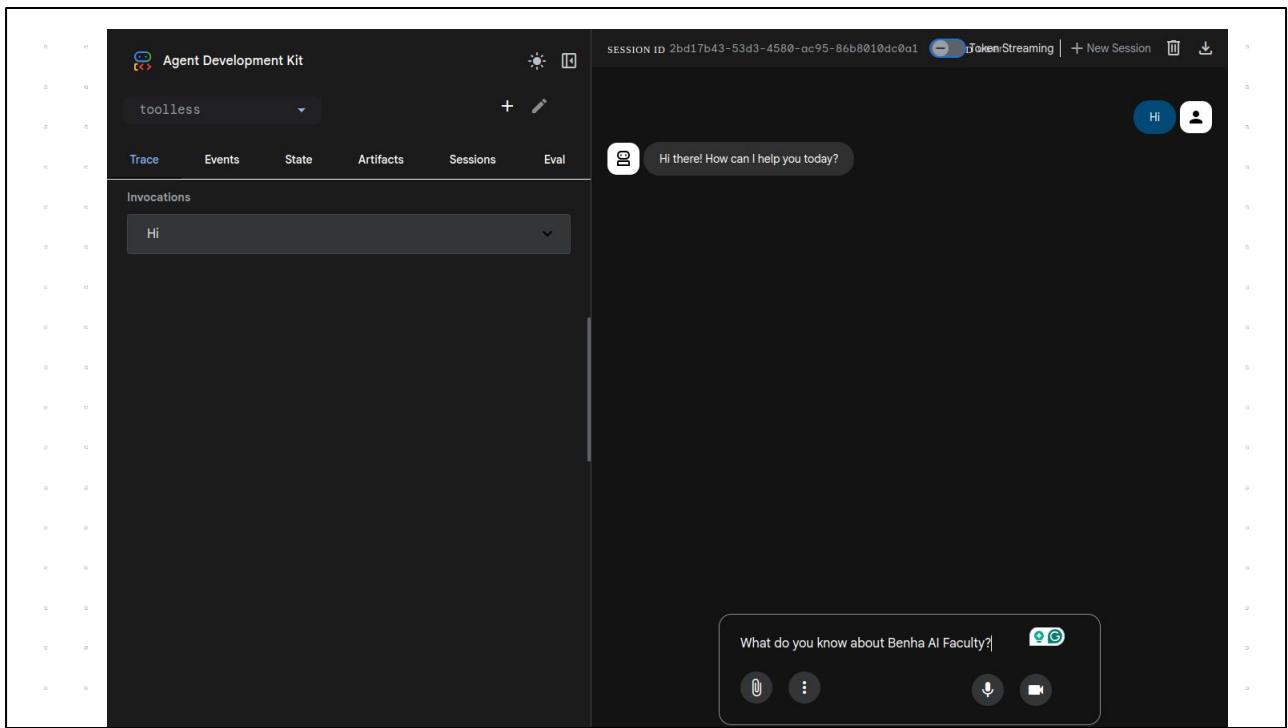
+-----+
| ADK Web Server started
| For local testing, access at http://127.0.0.1:8000.
+-----+

INFO:     Application startup complete.
INFO:     Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
```

We can click on the link and start testing our AI agent.







Agent Development Kit

SESSION ID 53d74a6a-6564-40d1-8ded-0544efc4a5b7 Token Streaming + New Session

toolless

+

Trace Events State Artifacts Sessions Eval

Invocations

What do you know about Benha AI Faculty?

Benha AI Faculty refers to the **Faculty of Artificial Intelligence** at Benha University in Egypt.

Here's what I know about it:

- **Part of Benha University:** It is one of the faculties within Benha University, a public university located in Benha, Qalyubia Governorate, Egypt.
- **Focus:** As the name suggests, its primary focus is on artificial intelligence, machine learning, data science, and related computational fields.
- **Programs:** Typically, such faculties offer undergraduate (Bachelor's) and potentially postgraduate (Master's, PhD) degrees in areas like:
 - Artificial Intelligence
 - Machine Learning
 - Data Science
 - Big Data Analytics
 - Robotics
 - Computer Vision
 - Natural Language Processing
- **Establishment:** Benha University has been actively expanding its technology and AI programs. The establishment of dedicated AI faculties is a trend in Egyptian universities to meet the growing demand for AI professionals.
- **Goals:** The faculty aims to:
 - Educate and train highly skilled professionals in AI
 - Conduct research and development in various AI sub-fields.
 - Contribute to the advancement of AI technology and its applications in Egypt and globally.
 - Foster innovation and provide solutions for industry and society using AI.

To get the most current and specific information (like exact degree programs, admission requirements, faculty members, and research areas), I would recommend visiting the official website of Benha University or the Faculty of Artificial Intelligence directly.

Type a Message...

...

...

...

Agent Development Kit

SESSION ID 53d74a6a-6564-40d1-8ded-0544efc4a5b7 Token Streaming + New Session

toolless

+

Trace Events State Artifacts Sessions Eval

Invocations

What do you know about Benha AI Faculty?

Benha AI Faculty refers to the **Faculty of Artificial Intelligence at Benha University** in Egypt.

Here's what I know about it:

- **Part of Benha University:** It is one of the faculties within Benha University, a public university located in Benha, Qalyubia Governorate, Egypt.
- **Focus:** As the name suggests, its primary focus is on artificial intelligence, machine learning, data science, and related computational fields.
- **Programs:** Typically, such faculties offer undergraduate (Bachelor's) and potentially postgraduate (Master's, PhD) degrees in areas like:
 - Artificial Intelligence
 - Machine Learning
 - Data Science
 - Big Data Analytics
 - Robotics
 - Computer Vision
 - Natural Language Processing
- **Establishment:** Benha University has been actively expanding its technology and AI programs. The establishment of dedicated AI faculties is a trend in Egyptian universities to meet the growing demand for AI professionals.
- **Goals:** The faculty aims to:
 - Educate and train highly skilled professionals in AI.
 - Conduct research and development in various AI sub-fields.
 - Contribute to the advancement of AI technology and its applications in Egypt and globally.
 - Foster innovation and provide solutions for industry and society using AI.

To get the most current and specific information (like exact degree programs, admission requirements, faculty members, and research areas), I would recommend visiting the official website of Benha University or the Faculty of Artificial Intelligence directly.

What are the results of today's matches in the Premier League?

Unmute Mute Camera

The screenshot shows the Agent Development Kit (ADK) interface. At the top, there's a header bar with the title "Agent Development Kit" and a session ID "SESSION ID 53d74a6a-6564-40d1-8ded-0544efc4a5b7". Below the header, a sidebar on the left lists "toolless" under "Tool", "Trace", "Events", "State", "Artifacts", "Sessions", and "Eval". The main area has two input fields: "What do you know about Benha AI Faculty?" and "What are the results of today's matches in the Premier League?". A large dark box on the right contains the AI's responses to these questions. The first response is about Benha AI Faculty, mentioning fields like Big Data Analytics, Robotics, Computer Vision, and Natural Language Processing, as well as the establishment of AI programs at Benha University. The second response is about today's Premier League matches, stating that no matches are scheduled for Friday, May 10th, 2024, and suggesting checking official websites or apps for updates.

SESSION ID 53d74a6a-6564-40d1-8ded-0544efc4a5b7 Taken Streaming | + New Session

toolless

+

Trace Events State Artifacts Sessions Eval

Invocations

What do you know about Benha AI Faculty?

What are the results of today's matches in the Premier League?

Big Data Analytics
Robotics
Computer Vision
Natural Language Processing

- Establishment: Benha University has been actively expanding its technology and AI programs. The establishment of dedicated AI faculties is a trend in Egyptian universities to meet the growing demand for AI professionals.
- Goals: The faculty aims to:
 - Educate and train highly skilled professionals in AI.
 - Conduct research and development in various AI sub-fields.
 - Contribute to the advancement of AI technology and its applications in Egypt and globally.
 - Foster innovation and provide solutions for industry and society using AI.

To get the most current and specific information (like exact degree programs, admission requirements, faculty members, and research areas), I would recommend visiting the official website of Benha University or the Faculty of Artificial Intelligence directly.

What are the results of today's matches in the Premier League?

As an AI, I don't have real-time access to live sports results that update by the minute. However, I can tell you that there are no Premier League matches scheduled for today, Friday, May 10th, 2024.

Premier League games typically take place on weekends, with occasional midweek fixtures. The next set of matches will be played over the weekend.

To get the most up-to-date information on results and upcoming fixtures, I recommend checking:

- The official Premier League website
- Major sports news websites (e.g., BBC Sport, Sky Sports, ESPN)
- Sports apps on your phone

Type a Message...

The screenshot shows the Cloud Shell Editor interface. In the top navigation bar, there are icons for file operations like Open, Save, and Print, along with a user profile icon. Below the bar, the title "Cloud Shell Editor" is displayed. The main area consists of two panes: the left pane is the "EXPLORER" view showing a directory structure for a project named "MAS" containing files like "agents/toolless", ".env", and "agent.py"; the right pane is the "CODE" view showing the content of "agent.py". The code defines an Agent object with specific parameters. Below the code editor is a status bar with information such as "Ln 1, Col 23", "Spaces: 4", "UTF-8", "LF", "Python 3.12.3 (env: venv)", and "Layout: us". The bottom pane is a terminal window titled "cloudshell" showing application logs. The logs indicate the application is waiting for shutdown, shutting down the ADK web server, and finally closing the runner and exiting. The terminal prompt at the end is "(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents\$".

```
INFO: Waiting for application shutdown.
+-----+
| ADK Web Server shutting down... |
+-----+
2025-11-26 10:30:20,538 - INFO - runners.py:1346 - Closing runner...
2025-11-26 10:30:20,539 - INFO - runners.py:1354 - Runner closed.
INFO: Application shutdown complete.
INFO: Finished server process [23827]
Aborted!
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$
```

We press Ctrl + c to cancel the adk web and return to the Linux prompt.



Create Agent
with Tool

Cloud Shell Editor

EXPLORER ... Welcome .env __init__.py agent.py

agents > toolless > agent.py > Agent

```
1 from google.adk.agents.llm_agent import Agent
2
3 root_agent = Agent(
4     model='gemini-2.5-flash',
5     name='root_agent',
6     description='A helpful assistant for user questions.',
7     instruction='Answer user questions to the best of your knowledge',
8 )
9
```

Ln 1, Col 34 Spaces: 4 UTF-8 LF {} Python 3.12.3 (env: venv) Layout: us

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents\$ adk create tool

Cloud Shell Editor

EXPLORER ... Welcome .env __init__.py agent.py

agents > toolless > agent.py > Agent

```
1 from google.adk.agents.llm_agent import Agent
2
3 root_agent = Agent(
4     model='gemini-2.5-flash',
5     name='root_agent',
6     description='A helpful assistant for user questions.',
7     instruction='Answer user questions to the best of your knowledge',
8 )
9
```

Ln 1, Col 34 Spaces: 4 UTF-8 LF {} Python 3.12.3 (env: venv) Layout: us

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents\$ adk create tool
Choose a model for the root agent:
1. gemini-2.5-flash
2. Other models (fill later)
Choose model (1, 2): 1

The screenshot shows the Cloud Shell Editor interface. The top bar includes tabs for 'Welcome', '.env', '_init_.py', and 'agent.py' (which is currently selected). The left sidebar has an 'EXPLORER' section showing a directory tree for 'MAS' containing 'agents/toolless', '_pycache_/_', '_init_.py', '.env', and 'agent.py'. Below this are 'OUTLINE' and 'TIMELINE' sections. The main area displays the 'agent.py' file content:

```
1 from google.adk.agents.llm_agent import Agent
2
3 root_agent = Agent(
4     model='gemini-2.5-flash',
5     name='root_agent',
6     description='A helpful assistant for user questions.',
7     instruction='Answer user questions to the best of your knowledge',
8 )
9
```

Below the code editor, status information includes 'Ln 1, Col 34', 'Spaces: 4', 'UTF-8 LF', 'Python 3.12.3 (env: venv)', and 'Layout: us'. The bottom part of the screen is a terminal window titled 'cloudshell' showing the command:

```
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ adk create tool
```

The terminal then prompts for a model choice:

```
Choose a model for the root agent:
1. gemini-2.5-flash
2. Other models (fill later)
```

It then asks for a model selection:

```
Choose model (1, 2): 1
```

It then asks for a backend choice:

```
1. Google AI
2. Vertex AI
```

It then asks for a backend selection:

```
Choose a backend (1, 2): 2
```

This time, we use Vertex AI, so we will need to specify the Cloud Project and the Cloud Region.

The screenshot shows the Cloud Shell Editor interface. On the left, the Explorer sidebar displays a project structure with a folder named 'MAS' containing subfolders 'agents' and 'toolless', and files like '_pycache_'. The main editor area shows a Python script named 'agent.py' with the following code:

```
1 from google.adk.agents.llm_agent import Agent
2
3 root_agent = Agent(
4     model='gemini-2.5-flash',
5     name='root_agent',
6     description='A helpful assistant for user questions.',
7     instruction='Answer user questions to the best of your knowledge',
8 )
```

Below the editor, the status bar indicates the file is Python 3.12.3 (env: venv) and the layout is US. The terminal tab at the bottom shows a session where the user runs the command `adk create tool`. The terminal output is as follows:

```
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ adk create tool
Choose a model for the root agent:
1. gemini-2.5-flash
2. Other models (fill later)
Choose model (1, 2): 1
1. Google AI
2. Vertex AI
Choose a backend (1, 2): 2

You need an existing Google Cloud account and project, check out this link for details:
https://google.github.io/adk-docs/get-started/quickstart/#gemini---google-cloud-vertex-ai

Enter Google Cloud project ID [ai-gcp-demos]:
```

By pressing Enter, we confirm the Cloud Project.

The screenshot shows the Cloud Shell Editor interface. The top bar includes tabs for 'Welcome', '.env', '_init_.py', and 'agent.py'. The code editor displays Python code for creating an Agent:

```
1 from google.adk.agents.llm_agent import Agent
2
3 root_agent = Agent(
4     model='gemini-2.5-flash',
5     name='root_agent',
6     description='A helpful assistant for user questions.',
7     instruction='Answer user questions to the best of your knowledge',
```

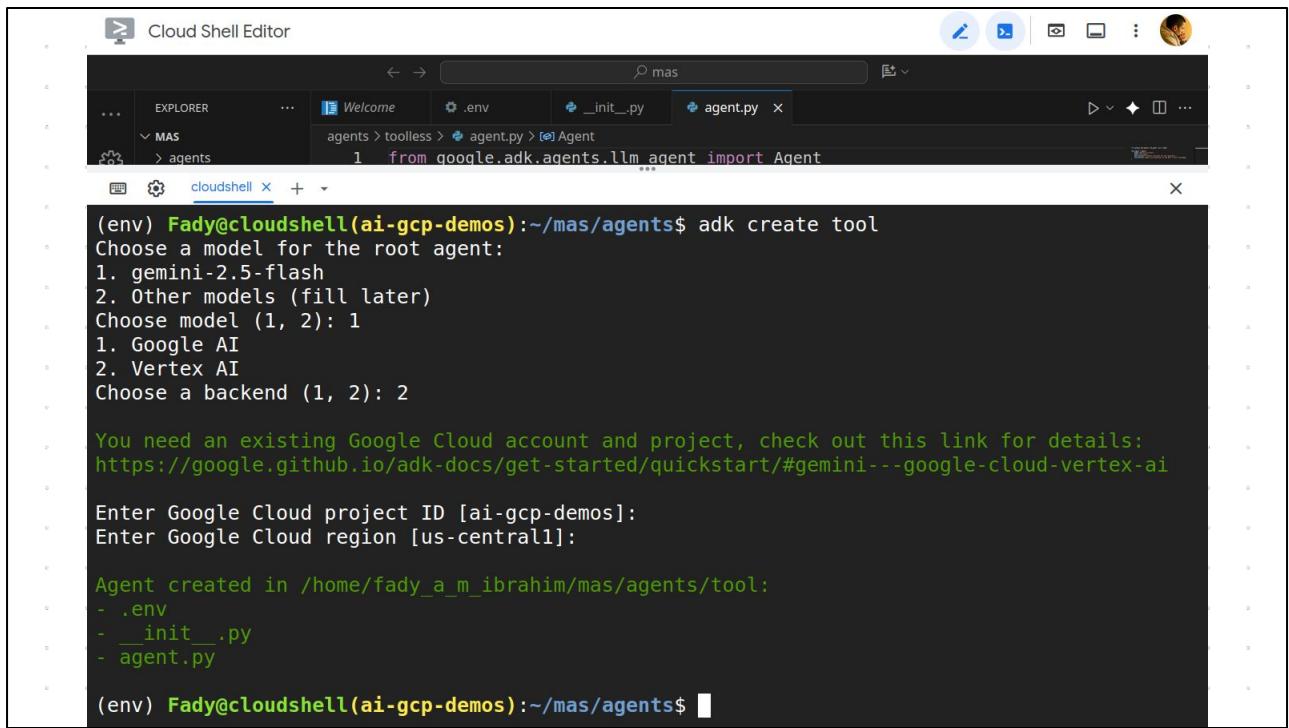
The bottom terminal window shows the command-line interaction:

```
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ adk create tool
Choose a model for the root agent:
1. gemini-2.5-flash
2. Other models (fill later)
Choose model (1, 2): 1
1. Google AI
2. Vertex AI
Choose a backend (1, 2): 2

You need an existing Google Cloud account and project, check out this link for details:
https://google.github.io/adk-docs/get-started/quickstart/#gemini---google-cloud-vertex-ai

Enter Google Cloud project ID [ai-gcp-demos]:
Enter Google Cloud region [us-central1]:
```

By pressing Enter, we confirm the Cloud Region.



The screenshot shows a Cloud Shell Editor interface with a terminal window open. The terminal window has a dark background and displays the following command-line session:

```
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ adk create tool
Choose a model for the root agent:
1. gemini-2.5-flash
2. Other models (fill later)
Choose model (1, 2): 1
1. Google AI
2. Vertex AI
Choose a backend (1, 2): 2

You need an existing Google Cloud account and project, check out this link for details:
https://google.github.io/adk-docs/get-started/quickstart/#gemini---google-cloud-vertex-ai

Enter Google Cloud project ID [ai-gcp-demos]:
Enter Google Cloud region [us-central1]:

Agent created in /home/fady_a_m_ibrahim/mas/agents/tool:
- .env
- __init__.py
- agent.py

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$
```

The three files are created like the previous AI agent.

Cloud Shell Editor

EXPLORER ... Welcome .env .../toolless __init__.py agent.py .env .../tool X

agents > tool > .env

1 GOOGLE_GENAI_USE_VERTEXAI=1
2 GOOGLE_CLOUD_PROJECT=ai-gcp-demos
3 GOOGLE_CLOUD_LOCATION=us-central1

Ln 1, Col 1 Spaces: 4 UTF-8 LF {} Properties Layout: us

cloudshell X +

```
You need an existing Google Cloud account and project, check out this link for details:  
https://google.github.io/adk-docs/get-started/quickstart/#gemini---google-cloud-vertex-ai  
Enter Google Cloud project ID [ai-gcp-demos]:  
Enter Google Cloud region [us-central1]:  
  
Agent created in /home/fady_a_m_ibrahim/mas/agents/tool:  
- .env  
- __init__.py  
- agent.py  
  
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit tool/.env  
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ 
```

Cloud Shell Editor

EXPLORER ...

MAS

agents

 tool

 ...
 __init__.py
 .env
 agent.py
 > toolless
 > env
> OUTLINE
> TIMELINE

agents > tool > __init__.py > {} agent

1 from . import agent
2

Ln 1, Col 1 Spaces: 4 UTF-8 LF {} Python 3.12.3 (env: venv) Layout: us

cloudshell x + x

```
https://google.github.io/adk-docs/get-started/quickstart/#gemini---google-cloud-vertex-ai

Enter Google Cloud project ID [ai-gcp-demos]:
Enter Google Cloud region [us-central1]:

Agent created in /home/fady_a_m_ibrahim/mas/agents/tool:
- .env
- __init__.py
- agent.py

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit tool/.env
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit tool/__init__.py
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ 
```

Cloud Shell Editor

EXPLORER ... _init_.py .../toolless agent.py .../toolless .env .../tool _init_.py .../tool agent.py .../tool

agents > tool > agent.py > ...

```
1 from google.adk.agents.llm_agent import Agent
2
3 root_agent = Agent(
4     model='gemini-2.5-flash',
5     name='root_agent',
6     description='A helpful assistant for user questions.',
7     instruction='Answer user questions to the best of your knowledge',
8 )
```

Ln 9, Col 1 Spaces: 4 UTF-8 LF {} Python 3.12.3 ('env': venv) Layout: us

cloudshell

```
Enter Google Cloud project ID [ai-gcp-demos]:
Enter Google Cloud region [us-central1]:
Agent created in /home/fady_a_m_ibrahim/mas/agents/tool:
- .env
- __init__.py
- agent.py

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit tool/.env
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit tool/__init__.py
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit tool/agent.py
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ 
```

main ↴ adk-python / src / google / adk / ↑ Top

errors	chore: Raise AlreadyExistsError when trying to create a r...	last month
evaluation	chore: Plumb memory service from LocalEvalService to E...	last week
events	chore: Clarify the usage of agent_state	3 weeks ago
examples	chore: Lazy load Vertex AI dependencies in ADK modules	3 weeks ago
features	feat: Add progressive SSE streaming feature	last week
flows	feat: Add enum constraint to agent_name for transfer_to...	yesterday
memory	feat: Add id and custom_metadata fields to MemoryEntry	last week
models	fix: Allow image parts in user messages for Anthropic Cl...	15 hours ago
planners	No public description	7 months ago
platform	refactor: Extract out platform specific code like threading	5 months ago
plugins	feat: update save_files_as_artifacts_plugin to never keep...	last week
sessions	fix: Update session last update time when appending ev...	5 days ago
telemetry	chore: Fix spelling in src	3 weeks ago
tools	feat(tools): Add debug logging to VertexAiSearchTool	yesterday

google / adk-python

Code Issues Pull requests Discussions Actions Projects Security ...

main ↗ adk-python / src / google / adk / tools / Go to file Add file ...

2 people and copybara-github feat(tools): Add debug logging to VertexAiSearchTool c6e7d6b · yesterday

Name	Last commit message	Last commit da...
..		
apihub_tool	chore: Fix spelling	3 weeks ago
application_integration_tool	ADK changes	3 months ago
bigrquery	feat: Propagate application_name set for the BigQuery T...	last week
bigtable	feat: add Bigtable tools	3 months ago
computer_use	chore: Remove unnecessary check tool type and tool att...	last month
google_api_tool	chore: Fix spelling in src	3 weeks ago
mcp_tool	fix: Update the retry_on_closed_resource decorator to re...	last week

adk-python / src / google / adk / tools
/ google_search_tool.py

Code Blame Raw ⌂ ⌄ ⌅ ⌆ ⌇ ⌈ ⌉

```
9 # unless required by applicable law or agreed to in writing, so
10 # distributed under the License is distributed on an "AS IS" BASIS
11 # WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
12 # implied. See the License for the specific language governing permission
13 # and limitations under the License.
14
15 from __future__ import annotations
16
17 from typing import TYPE_CHECKING
18
19 from google.genai import types
20 from typing_extensions import override
21
22 from ..utils.model_name_utils import is_gemini_1_model
23 from ..utils.model_name_utils import is_gemini_model
24 from .base_tool import BaseTool
25 from .tool_context import ToolContext
26
27 if TYPE_CHECKING:
28     from ..models import LlmRequest
29
30
31 > class GoogleSearchTool(BaseTool): ...
32         )
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
59
60
61
62
63
64
65
66
67
68
69
69
70
71
72
73
74
75
```

Symbols

Find definitions and references for functions and other symbols in this file by clicking a symbol below or in the code.

Filter symbols

class GoogleSearchTool

func __init__

func process_llm_request

const google_search

The screenshot shows a code editor interface with two main panes. The left pane displays the file `adk-python / src / google / adk / tools / __init__.py`. The right pane is a search interface titled "google_search" with the status "1/5".

Code Editor (Left):

```
24     from .apihub_tool.apihub_toolset import APIHubToolset
25     from .base_tool import BaseTool
26     from .discovery_engine_search_tool import DiscoveryEngineSearch
27     from .enterprise_search_tool import enterprise_web_search_tool
28     from .example_tool import ExampleTool
29     from .exit_loop_tool import exit_loop
30     from .function_tool import FunctionTool
31     from .get_user_choice_tool import get_user_choice_tool as get_
32     from .google_maps_grounding_tool import google_maps_grounding
33     from .google_search_tool import google_search
34     from .load_artifacts_tool import load_artifacts_tool as load_a
35     from .load_memory_tool import load_memory_tool as load_memory
36     from .long_running_tool import LongRunningFunctionTool
37     from .preload_memory_tool import preload_memory_tool as preloa
38     from .tool_context import ToolContext
39     from .transfer_to_agent_tool import transfer_to_agent
40     from .transfer_to_agent_tool import TransferToAgentTool
41     from .url_context_tool import url_context
42     from .vertex_ai_search_tool import VertexAiSearchTool
43
44     # If you are adding a new tool to this file, please make sure yo
45     # lazy mapping to avoid expensive imports. If the tool is not us
46     # party dependencies, please feel free to import it eagerly at t
47     # file.
```

Symbols Search (Right):

Find definitions and references for functions and other symbols in this file by clicking a symbol below or in the code.

Filter symbols

- const _LAZY_MAPPING
- const _all_
- func __getattr__
- func __dir__

Cloud Shell Editor

EXPLORER ... _init_.py .../toolless agent.py .../toolless .env .../tool _init_.py .../tool agent.py .../tool

agents > tool > agent.py > root_agent

```
1 from google.adk.agents.llm_agent import Agent
2 from google.adk.tools import google_search
3
4 root_agent = Agent([
5     model='gemini-2.5-flash',
6     name='root_agent',
7     description='A helpful assistant for user questions.',
8     instruction='Answer user questions to the best of your knowledge',
9     tools=[google_search],
10 ])
```

Ln 9, Col 27 Spaces: 4 UTF-8 LF Python 3.12.3 ('env': venv) Layout: us

cloudshell

```
Enter Google Cloud region [us-central1]:
```

```
Agent created in /home/fady_a_m_ibrahim/mas/agents/tool:
- .env
- __init__.py
- agent.py

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit tool/.env
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit tool/_init_.py
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit tool/agent.py
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ 
```

Cloud Shell Editor

EXPLORER ... _init_.py .../toolless agent.py .../toolless .env .../tool _init_.py .../tool agent.py .../tool

agents > tool > agent.py > root_agent

```
1 from google.adk.agents.llm_agent import Agent
2 from google.adk.tools import google_search
3
4 root_agent = Agent(
5     model='gemini-2.5-flash',
6     name='root_agent',
7     description='A helpful assistant for user questions.',
8     instruction='Answer user questions to the best of your knowledge',
9     tools=[google_search],
10 )
```

Ln 9, Col 27 Spaces: 4 UTF-8 LF Python 3.12.3 ('env': venv) Layout: us

cloudshell

```
Enter Google Cloud region [us-central1]:  
Agent created in /home/fady_a_m_ibrahim/mas/agents/tool:  
- .env  
- __init__.py  
- agent.py  
  
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit tool/.env  
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit tool/__init__.py  
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit tool/agent.py  
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ adk web
```

The screenshot shows a Cloud Shell Editor interface. In the top navigation bar, there is a logo, the text "Cloud Shell Editor", and several icons for file operations like copy, paste, and search. Below the navigation bar is a search bar with the text "mas". The left sidebar has sections for "EXPLORER", "OUTLINE", and "TIMELINE". The "EXPLORER" section shows a file tree with a folder named "MAS" containing subfolders "agents", "tool", ".env", and "agent.py". The "agent.py" file is selected and shown in the main editor area. The code in "agent.py" is as follows:

```
from google.adk.agents.llm_agent import Agent
from google.adk.tools import google_search

root_agent = Agent(
    model='gemini-2.5-flash',
    name='root_agent',
    description='A helpful assistant for user questions.',
    instruction='Answer user questions to the best of your knowledge',
    tools=[google_search],
)
```

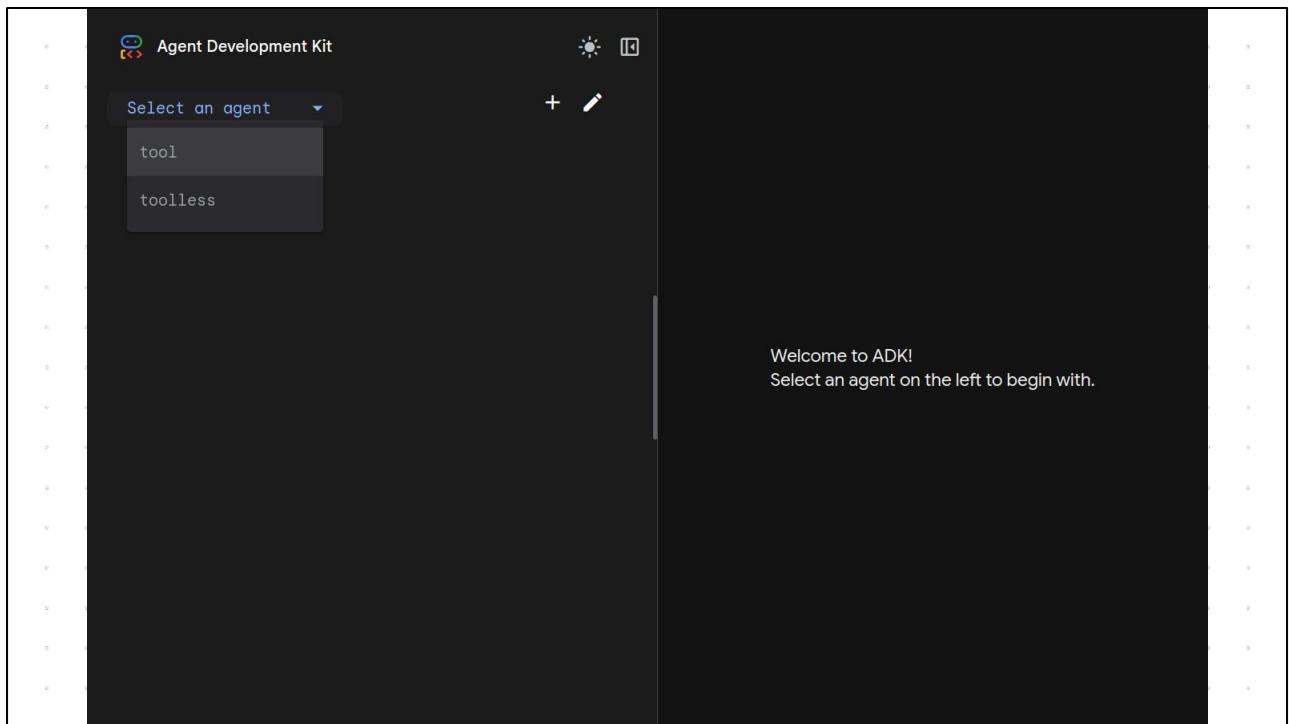
Below the code editor is a status bar with information: Ln 9, Col 27, Spaces: 4, UTF-8, LF, Python 3.12.3 ('env': venv), Layout: us. The bottom half of the screen is a terminal window titled "cloudshell" with the following output:

```
INFO: Waiting for application startup.

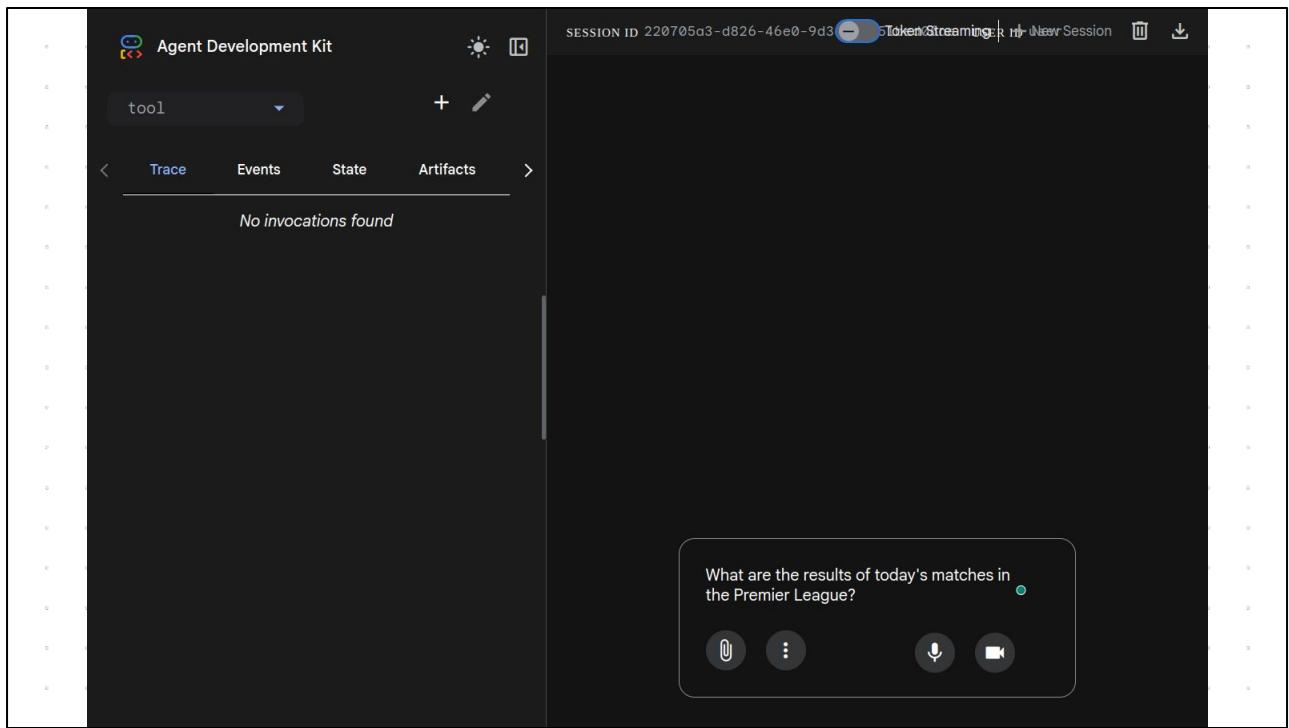
+-----+
| ADK Web Server started
|
| For local testing, access at http://127.0.0.1:8000.
+-----+

INFO: Application startup complete.
INFO: Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
```

We can click on the link and start testing our AI agent.



Because we now have two AI agents in the “agents” folder, we need to choose which one to use from the combobox.



Agent Development Kit

SESSION ID 220705a3-d826-46e0-9d33-5655f3333333 Token Streaming New Session

tool +

Trace Events State Artifacts >

Invocations

What are the results of today's matches in the Premier League?

There were no Premier League matches played on Wednesday, November 26, 2025. The Premier League schedule for November 2025 includes matchweeks that span weekends and early parts of the month, with an international break occurring between November 10-21. Fixtures were scheduled for other dates in November, such as the 22nd, 23rd, 24th, 29th, and 30th, but not for the 26th.

Premier League matches on November 26 2025 were tr

Type a Message... 

Unmute  Microphone 

Cloud Shell Editor

EXPLORER ... _init_.py .../toolless agent.py .../toolless .env .../tool _init_.py .../tool agent.py .../tool

mas

agents > tool > agent.py > root_agent

```
1 from google.adk.agents.llm_agent import Agent
2 from google.adk.tools import google_search
3
4 root_agent = Agent(
5     model='gemini-2.5-flash',
6     name='root_agent',
7     description='A helpful assistant for user questions.',
8     instruction='Answer user questions to the best of your knowledge',
9     tools=[google_search],
10 )
11
```

Ln 9, Col 27 Spaces: 4 UTF-8 LF {} Python 3.12.3 (env: venv) Layout: us

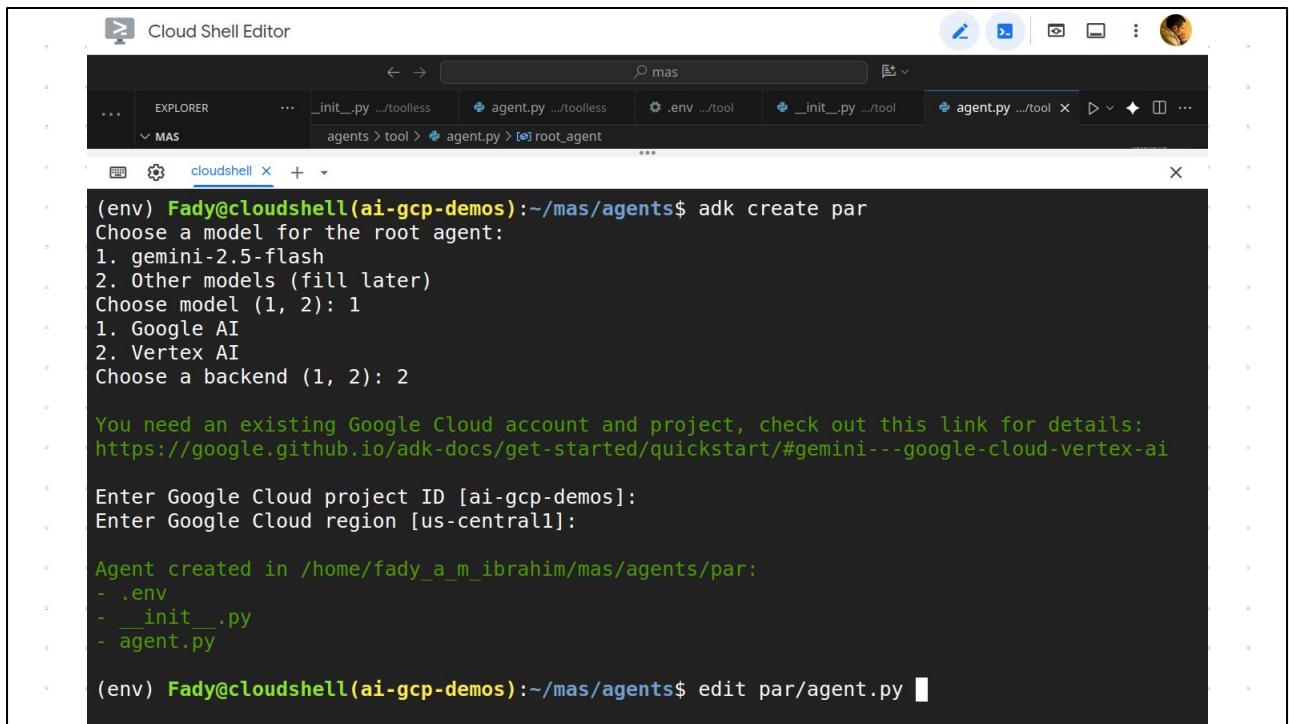
cloudshell

```
| ADK Web Server shutting down...
+-----+
2025-11-26 12:55:07,976 - INFO - runners.py:1346 - Closing runner...
2025-11-26 12:55:07,976 - INFO - runners.py:1354 - Runner closed.
INFO: Application shutdown complete.
INFO: Finished server process [11298]

Aborted!
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$
```



Create Parallel
Agent



The screenshot shows a Cloud Shell Editor interface with a terminal window open. The terminal window has a dark background and displays the following command-line session:

```
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ adk create par
Choose a model for the root agent:
1. gemini-2.5-flash
2. Other models (fill later)
Choose model (1, 2): 1
1. Google AI
2. Vertex AI
Choose a backend (1, 2): 2

You need an existing Google Cloud account and project, check out this link for details:
https://google.github.io/adk-docs/get-started/quickstart/#gemini---google-cloud-vertex-ai

Enter Google Cloud project ID [ai-gcp-demos]:
Enter Google Cloud region [us-central1]:

Agent created in /home/fady_a_m_ibrahim/mas/agents/par:
- .env
- __init__.py
- agent.py

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit par/agent.py
```

We created the third AI agent in the same way as the previous one.

Cloud Shell Editor

EXPLORER ... agent.py .../toolless .env .../tool __init__.py .../tool agent.py .../par X D v ♦ ...

agents > par > agent.py > ...

```
1  from google.adk.agents.llm_agent import Agent
2
3  root_agent = Agent(
4      model='gemini-2.5-flash',
5      name='root_agent',
6      description='A helpful assistant for user questions.',
7      instruction='Answer user questions to the best of your knowledge',
8  )
9
```

Ln 9, Col 1 Spaces: 4 UTF-8 LF {} Python 3.12.3 (env: venv) Layout: us

cloudshell X

```
Enter Google Cloud project ID [ai-gcp-demos]:
Enter Google Cloud region [us-central]:
Agent created in /home/fady_a_m_ibrahim/mas/agents/par:
- .env
- __init__.py
- agent.py

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit par/agent.py
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$
```

adk-python / src / google / adk / agents
/ parallel_agent.py

Code Blame Raw ⌂ ⌄ ⌅ ⌆ ⌇ ⌈ ⌉

```
115     async def _merge_agent_run()
134         async def process_an_agent(events_for_one_agent):
135             sentinel_count += 1
136             else:
137                 yield event
138                 # Signal to agent that it should generate next event.
139                 resume_signal.set()
140
141
162     class ParallelAgent(BaseAgent):
163         """A shell agent that runs its sub-agents in parallel in an is
164
165         This approach is beneficial for scenarios requiring multiple p
166         attempts on a single task, such as:
167
168         - Running different algorithms simultaneously.
169         - Generating multiple responses for review by a subsequent eva
170         """
172         config_type: ClassVar[type[BaseAgentConfig]] = ParallelAgentCo
173         """The config type for this agent."""
174
175         @override
176         async def _run_async_impl(
```

ParallelAgent 3/6 ⌂ ⌄ ⌅ ⌆ ⌇ ⌈ ⌉

Find definitions and references for functions and other symbols in this file by clicking a symbol below or in the code.

Filter symbols ⌂ r

- func _create_branch_ctx_for_s...
- func _merge_agent_run_pre_3...
- func propagate_exceptions
- func process_an_agent
- func _merge_agent_run
- func process_an_agent
- class ParallelAgent
 - const config_type
 - func _run_async_impl
 - func _run_live_impl

Cloud Shell Editor

EXPLORER ... agent.py .../toolless .env .../tool __init__.py .../tool agent.py .../par X D v ♦ ...

agents > par > agent.py > [e]root_agent

```
1 from google.adk.agents.parallel_agent import ParallelAgent
2
3 root_agent = ParallelAgent(
4     #model='gemini-2.5-flash',
5     name='root_agent',
6     description='A helpful assistant for user questions.',
7     #instruction='Answer user questions to the best of your knowledge',
8 )
9
```

Ln 6, Col 59 Spaces: 4 UTF-8 LF {} Python 3.12.3 (env: venv) Layout: us

cloudshell X +

```
Enter Google Cloud project ID [ai-gcp-demos]:
Enter Google Cloud region [us-central1]:
Agent created in /home/fady_a_m_ibrahim/mas/agents/par:
- .env
- __init__.py
- agent.py

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit par/agent.py
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$
```

Cloud Shell Editor

EXPLORER ... agent.py .../toolless .env .../tool __init__.py .../tool agent.py .../par X agent.py .../par X D v ⚡ ...

agents > par > agent.py > [e]root_agent

```
1 from google.adk.agents.parallel_agent import ParallelAgent
2
3 root_agent = ParallelAgent(
4     #model='gemini-2.5-flash',
5     name='root_agent',
6     description='A helpful assistant for user questions.',
7     #instruction='Answer user questions to the best of your knowledge',
8 )
9
```

Ln 6, Col 59 Spaces: 4 UTF-8 LF {} Python 3.12.3 (env: venv) Layout: us

cloudshell X + X

```
Enter Google Cloud project ID [ai-gcp-demos]:
Enter Google Cloud region [us-central1]:

Agent created in /home/fady_a_m_ibrahim/mas/agents/par:
- .env
- __init__.py
- agent.py

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ edit par/agent.py
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ mkdir par/sub█
```

Cloud Shell Editor

EXPLORER ... agent.py .../toolless .env .../tool __init__.py .../tool agent.py .../par X agent.py .../par X D v ...

agents > par > agent.py > root_agent

```
1 from google.adk.agents.parallel_agent import ParallelAgent
2
3 root_agent = ParallelAgent(
4     #model='gemini-2.5-flash',
5     name='root_agent',
6     description='A helpful assistant for user questions.',
7     #instruction='Answer user questions to the best of your knowledge',
8 )
9
```

Ln 6, Col 59 Spaces: 4 UTF-8 LF {} Python 3.12.3 (env: venv) Layout: us

(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents\$ mv tool par/sub/

The screenshot shows the Cloud Shell Editor interface. The top half is a code editor with a dark theme, displaying Python code for a parallel agent. The code imports `ParallelAgent` from `google.adk.agents.parallel_agent`, creates a new instance named `root_agent` with specific parameters, and defines its description and instruction. Below the code editor is a terminal window titled "cloudshell" showing a command-line session. The user is in a directory `~/mas/agents` and runs the command `mv tool par/sub/`. The terminal has tabs for "0", "0", "Gen AI App Demo", and "cloudshell". The status bar at the bottom of the terminal indicates the environment is "env: venv".

Cloud Shell Editor

EXPLORER ...

MAS

agents

 par

 sub / tool

 > _pycache_

 agent.py

 init.py

 .env

 > .env

 > OUTLINE

 > TIMELINE

agent.py .../toolless

.env .../tool

init.py .../tool

agent.py .../par

agent.py .../par X

mas

agents > par > agent.py > [e]root_agent

```
1 from google.adk.agents.parallel_agent import ParallelAgent
2
3 root_agent = ParallelAgent(
4     #model='gemini-2.5-flash',
5     name='root_agent',
6     description='A helpful assistant for user questions.',
7     #instruction='Answer user questions to the best of your knowledge',
8 )
9
```

Ln 6, Col 59 Spaces: 4 UTF-8 LF {} Python 3.12.3 (env: venv) Layout: us

cloudshell X +

```
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ mv tool par/sub/
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ mv toolless/ par/sub/
```

Cloud Shell Editor

EXPLORER ...

MAS
 agents/par
 sub
 tool
 > toolless
 > _init_.py
 .env
 agent.py
 > env

agent.py .../toolless .env .../tool __init__.py .../tool agent.py .../par X agent.py .../par X ...

agents > par > agent.py > [e]root_agent

```
1 from google.adk.agents.parallel_agent import ParallelAgent
2
3 root_agent = ParallelAgent(
4     #model='gemini-2.5-flash',
5     name='root_agent',
6     description='A helpful assistant for user questions.',
7     #instruction='Answer user questions to the best of your knowledge',
8 )
9
```

Ln 6, Col 59 Spaces: 4 UTF-8 LF {} Python 3.12.3 (env: venv) Layout: us

> OUTLINE > TIMELINE

X 0 ▲ 0 ⌂ Gen AI App Demo

cloudshell X + -

```
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ mv tool par/sub/
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ mv toolless/ par/sub/
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$
```

Cloud Shell Editor

EXPLORER ...

MAS
 agents/par
 sub
 tool
 > toolless
 > _init_.py
 > .env
 > agent.py
 > env

agent.py .../toolless .env .../tool __init__.py .../tool agent.py .../par agent.py .../par X

```
agents > par > agent.py > {} tool_agent
1   from google.adk.agents.parallel_agent import ParallelAgent
2   from .sub.tool.agent import root_agent as tool_agent
3
4   root_agent = ParallelAgent(
5       #model='gemini-2.5-flash',
6       name='root_agent',
7       description='A helpful assistant for user questions.',
8       #instruction='Answer user questions to the best of your knowledge',
9   )
10
```

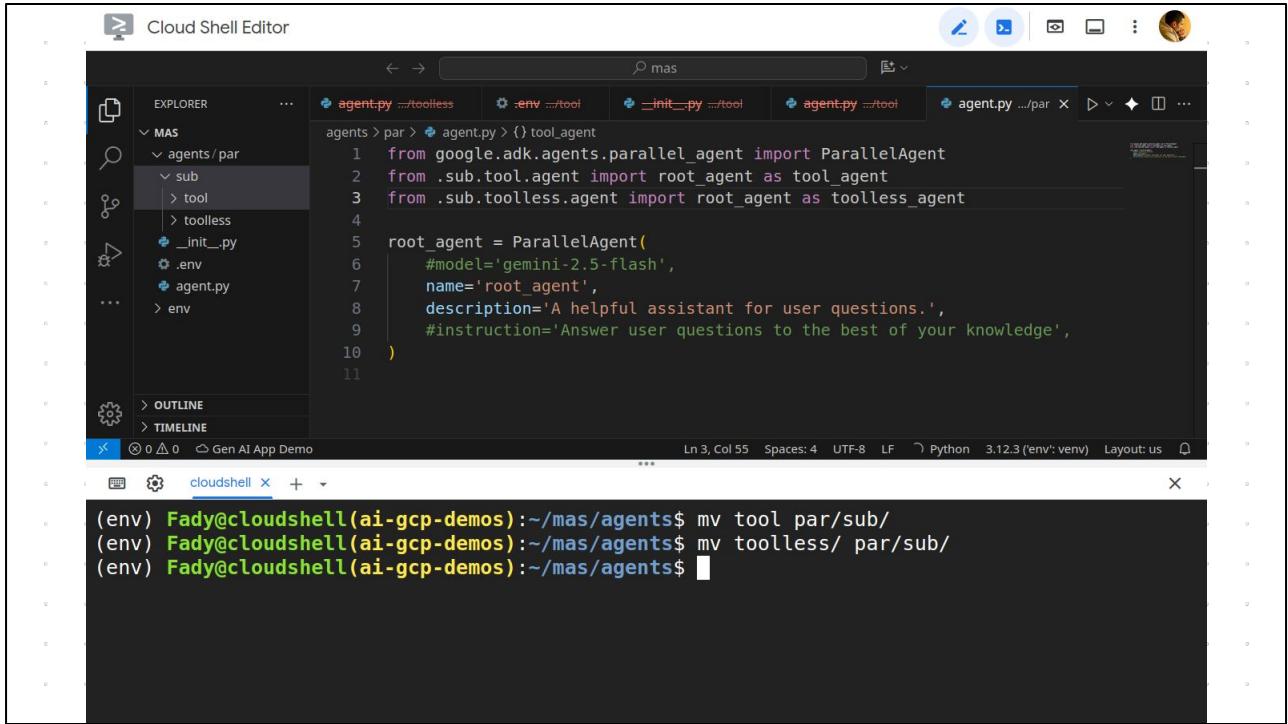
Ln 2, Col 53 Spaces: 4 UTF-8 LF Python 3.12.3 (env: venv) Layout: us

> OUTLINE > TIMELINE

X 0 ▲ 0 ⌂ Gen AI App Demo

cloudshell X +

```
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ mv tool par/sub/
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ mv toolless/ par/sub/
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$
```



Cloud Shell Editor

EXPLORER ...

MAS

agents / par / sub / tool / toolless

init.py .env agent.py env

agent.py .../toolless .env .../tool __init__.py .../tool agent.py .../par

agents > par > agent.py > root_agent

```
1 from google.adk.agents.parallel_agent import ParallelAgent
2 from .sub.tool.agent import root_agent as tool_agent
3 from .sub.toolless.agent import root_agent as toolless_agent
4
5 root_agent = ParallelAgent(
6     name='root_agent',
7     description='A helpful assistant for user questions.',
8     sub_agents=[tool_agent, toolless_agent],
9 )
10
```

Ln 8, Col 45 Spaces: 4 UTF-8 LF Python 3.12.3 (env: venv) Layout: us

> OUTLINE > TIMELINE

cloudshell

```
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ mv tool par/sub/
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ mv toolless/ par/sub/
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$
```

Cloud Shell Editor

EXPLORER ...

MAS

agents / par / sub / tool / toolless

init.py .env agent.py env

agent.py .../toolless .env .../tool __init__.py .../tool agent.py .../par

agents > par > agent.py > root_agent

```
1 from google.adk.agents.parallel_agent import ParallelAgent
2 from .sub.tool.agent import root_agent as tool_agent
3 from .sub.toolless.agent import root_agent as toolless_agent
4
5 root_agent = ParallelAgent(
6     name='root_agent',
7     description='A helpful assistant for user questions.',
8     sub_agents=[tool_agent, toolless_agent],
9 )
10
```

LN 8, COL 45 SPACES: 4 UTF-8 LF Python 3.12.3 (env: venv) Layout: us

> OUTLINE > TIMELINE

X 0 ▲ 0 ⚡ Gen AI App Demo

cloudshell X +

```
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ mv tool par/sub/
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ mv toolless/ par/sub/
(env) Fady@cloudshell(ai-gcp-demos):~/mas/agents$ adk web
```

Cloud Shell Editor

EXPLORER ...

MAS
 agents/par
 sub
 > tool
 > toolless
 > _init_.py
 > .env
 > agent.py
 > env

agent.py .../toolless .env .../tool __init__.py .../tool agent.py .../par agent.py .../par X

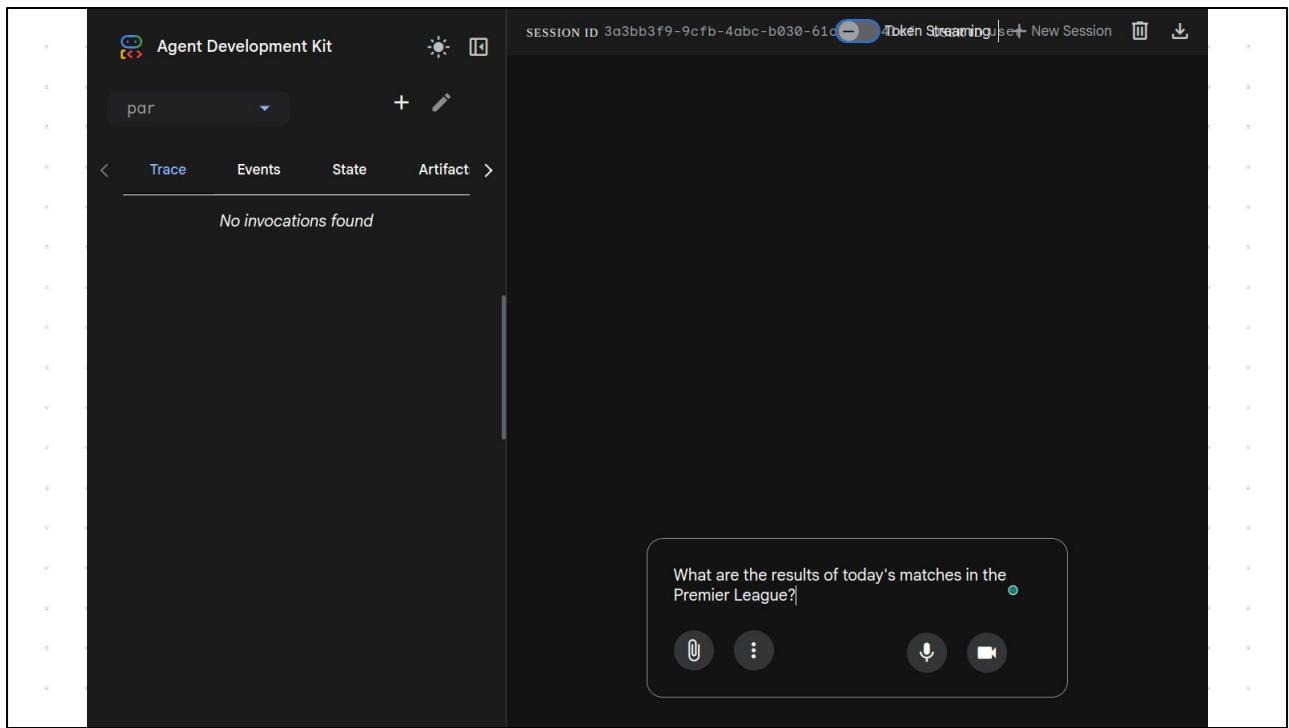
agents > par > agent.py > [e]root_agent

```
1  from google.adk.agents.parallel_agent import ParallelAgent
2  from .sub.tool.agent import root_agent as tool_agent
3  from .sub.toolless.agent import root_agent as toolless_agent
4
5  root_agent = ParallelAgent(
6      name='root_agent',
7      description='A helpful assistant for user questions.',
8      sub_agents=[tool_agent, toolless_agent],
9  )
10
```

LN 8, Col 45 Spaces: 4 UTF-8 LF Python 3.12.3 ('env': venv) Layout: us

cloudshell X + -

```
| ADK Web Server started
| For local testing, access at http://127.0.0.1:8000.
+-----+
INFO:     Application startup complete.
INFO:     Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
```



Agent Development Kit

par

SESSION ID 0567a41b-4f0b-406e-8e91 Token Streaming New Session

What are the results of today's matches in the Premier League?

There were no Premier League matches scheduled for today, November 26, 2025.

The Premier League's Matchweek 12 concluded on November 24, 2025, and Matchweek 13 is set to begin on November 29, 2025. The month of November 2025 included matchweeks 10 through 13, with an international break occurring between November 10-21.

Premier League fixtures today November 26 2025

I apologize, but as an AI, I do not have access to real-time, live sports data. Therefore, I cannot provide you with the results of today's Premier League matches.

To get the most up-to-date information, I recommend checking dedicated sports news websites or apps. Some reliable sources include:

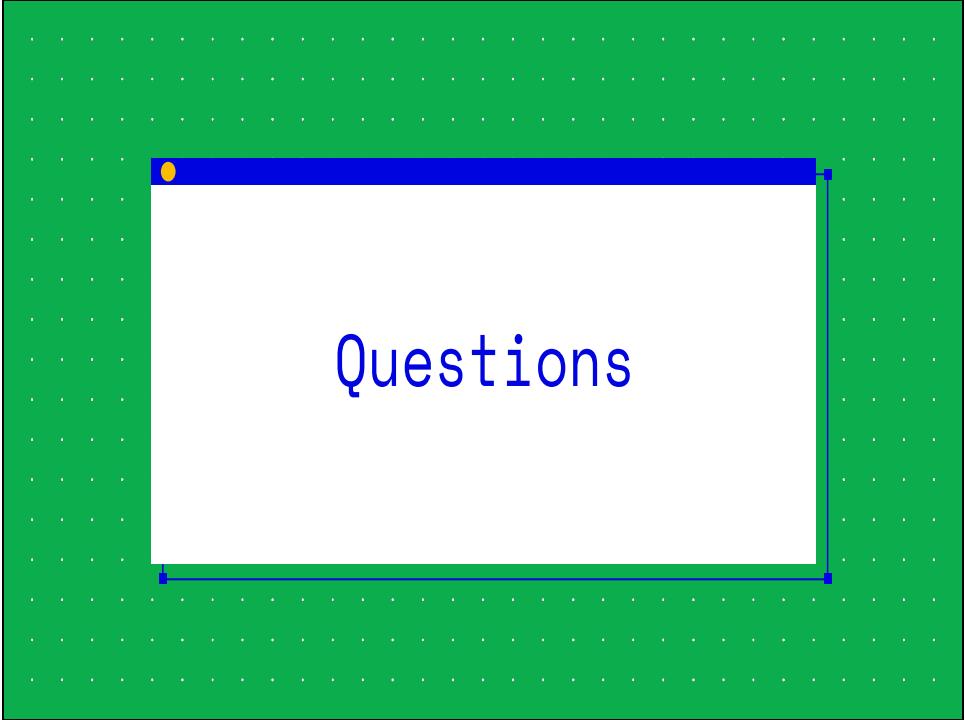
- Premier League Official Website
- BBC Sport
- Sky Sports
- ESPN
- Google Sports (just search for "Premier League scores")

Type a Message...

...

...

...



Questions

Links

<https://github.com/fcai-b/agents>