Level 1 Quiz [Agents SDK]

Q-   What does the OpenAI Agents SDK primarily help with?

A- Building AI Agents

Q-   Which programming language is the SDK designed for?

A- Python

Q-   What is the main class to run agents?

A- Runner

Q-   What do you import to define a new agent?

A- from agents import Agent

Q-   The SDK allows integration with?

A- Multiple Tools and APIs

Q-   Which method is used to start agent execution?

A- Runner.run()

Q-   What is a context in the SDK?

A- A dependency injection object

Q-   Agents are generic on their?

A- Context type

Q-   Which decorator is used to define tools for agents?

A- @function_tool

Q-   What does AsyncOpenAI represent?

A- An asynchronous OpenAI client

Q-   To use environment variables, you commonly use?

A- dotenv

Q-   What does RunConfig help configure?

A- how the agent run executes

Q-   What is handoff in agents SDK?

A- transferring control between agents

Q-   Can you provide any Python object as context?

A- Yes

Q-   Which command installs the SDK?

A- pip install openai-agents

Q-   What is the purpose of set_tracing_disabled(True)?

A- Disable tracing logs

Q-   What is the default event loop for AsyncOpenAI?

A- asyncio

Q-   What file stores environment variables?

A- env

Q-   Can you create multiple agents in one run?

A- True

Q-   What parameter is passed to Runner.run()?

A- Context object

Q-   What is the benefit of @function_tool?

A-   converts a function into a callable tool for agent

Q-   Which library is often used with SDK for async HTTP?

A-   aiohttp

Q-   What does Agent class do?

A-   Defines an AI agent with tools and logic

Q-   Runner is responsible for?

A-   Running agents and managing context

Q-   Which of these is NOT part of SDK?

A-   FlaskAgent

Q-   Can agents call external APIs?

A-   Yes

Q-   Which package loads environment variables?

A-   python_dotenv

Q-   Is SDK limited to OpenAI models?

A-   No, you can use others with tools

Q-   What is agents.run module for?

A-   Configurations like RunConfig

Q-   Where do you define tools?

A-   Inside agent logic with decorators

Q-   Can agents maintain state across steps?

A-   Yes with Context

Q-   What Python version is recommended?

A-   3.10+

Q-   How to stop tracing?

A-   set_tracing_disabled(True)

Q-   Does SDK support sync and async?

A-   Yes both

Q-   Can you debug agents?

A-   Yes using tracing

Q-   Handoff allows?

A-   Delegation between agents

Q-   What is a tool in SDK?

A-   A function exposed for agent to call

Q-   What is installed via pip install?

A-   python packages

Q-   What is the role of OpenAIChatCompletionsModel?

A-   Model integration for chat completions

Q-   Which file often includes requirements?

A-   requirements.txt

Q-   Can agents be tested locally?

A-   Yes

Q-   What is an SDK?

A-   Software Development Kit

Q-   Where is context passed?

A- Runner.run(context=...)

Q- What is dotenv used for?

A- Load environment variables

Q- What is FastAPI's role with SDK?

A- Host the agent logic as an API

Q- What is a typical agent output?

A- JSON or text response

Q- Which folder should hold Python code?

A- src or root folder

Q- What type of decorator is @function_tool?

A- Python function decorator

Q- How do you manage secrets?

A- In .env + load with dotenv

Q- Does the SDK support multiple tools?

A- Yes, you can define many