```
"""
Building an Autonomous Agent in 5 minutes with:
- LLM: OpenAI, Anthropic, EleutherAI, Hugging Face: Transformers
- Tools: Search, Browser, ETC
- Long Term Mmeory: ChromaDB, Weaviate, Pinecone, ETC
"""

from swarms import Agent, OpenAIChat, tool
from examples.demos.agent_in_5.chroma_db import ChromaDB

# Initialize the memory
chroma = ChromaDB(
    metric="cosine",
    limit_tokens=1000,
    verbose=True,
    # docs_folder = "docs" # Add your docs folder here
)


"""
How to make a tool in Swarms:
- Use the @tool decorator
- Define the function with the required arguments
- Add a docstring with the description of the tool
"""
```

```
# Create a tool
@tool  # Use this decorator
def browser(query: str = None):  # Add types
    """
    Opens a web browser and performs a Google search with the given query.

    Args:
        query (str): The search query to be performed.

    Returns:
        str: A message indicating that the browser is being opened for the given query.
    """
    import webbrowser

    url = f"https://www.google.com/search?q={query}"
    webbrowser.open(url)
    return f"Opening browser for: {query}"


# Initialize the agent
agent = Agent(
    llm=OpenAIChat(),
    agent_name="AI Engineer",
    agent_description=(
        "Creates AI Models for special use cases using PyTorch"
    ),
    system_prompt=(
        "Create an AI model for earthquake prediction using PyTorch."
    ),
    max_loops=4,  # or "auto"
    autosave=True,
    dashboard=True,
    verbose=True,
    stopping_token="<DONE>",
    interactive=True,
    tools=[browser],
    long_term_memory=chroma,  # pass in your memory object
)

# Run the agent
out = agent.run(
    "Let's make an AI model for earthquake prediction in pytorch."
)
print(out)
```