

Implementation Assignment 2: Build an Agentic Loop from Scratch

CS 394E — Contemporary Topics in LLMs · KAUST · Fall 2026

Released	Tuesday 2026-10-06
Due	Tuesday 2026-10-13, 23:59 AST, on Blackboard
Starter file	agent_loop.ipynb (on the course website, Assignments page)
Weight	5% of the final grade (graded out of 10 points, plus up to 2 bonus points)

What this assignment is about

Agent frameworks such as LangChain, LangGraph and AutoGen all contain the same small loop: call the LLM, run the tools it asks for, send the results back, and repeat until it gives a final answer. In this assignment you write that loop yourself, in about 20 lines of Python, against a hosted LLM and a toy environment.

The idea to take away is one sentence:

The LLM proposes what should happen next; the runtime makes it happen.

This builds on week 4, where tool use and the agent loop were introduced through the *ReAct* reading. It is also the plumbing that weeks 7 and 8 (multi-agent frameworks, agent infrastructure and memory) build on.

What you will do

The notebook walks you through four parts. Only edit the cells marked TODO, the answer cells, and (if you use Ibex) the PROVIDER line.

Part	What	You write
0	Setup: API key and provider, LLM client, toy environment, tools	a <code>.env</code> file with your key (and your choice of provider)
1	Call the LLM once and inspect the tool call it returns	Question 1
2	Implement the tool runtime (<code>execute_tool_call</code>), then do one agent step by hand	~5 + ~4 lines

Part	What	You write
3	Turn that step into a loop (run_agent), run it, and check the environment	~10 lines, Question 2 (optional bonus: ~6 lines and a question)

Setup

- **Python 3.10 or newer** (3.12 is what we tested with) on your own laptop. No GPU is needed.
- **An LLM to talk to.** Two options; you choose one with a single line at the top of the notebook (PROVIDER = "mistral", the default, or "ibex"), and nothing else changes:
 - **Mistral API** ("mistral"): create a key on the free “Experiment” plan at <https://console.mistral.ai/>. You will reuse it in assignment 3 (RAG). The free plan allows roughly one request per second; the provided call_llm already paces itself and retries on transient errors.
 - **The course’s LLM server on Ibex** ("ibex"): DeepSeek V4 Flash, with a personal key handed out by the instructors. It is **best effort**: it only works from the KAUST network or the VPN, and it can be slow or unavailable when many people use it at once. Don’t count on it the evening before the deadline; if it is down, switch to Mistral.
- Put your key in a file named .env next to the notebook: one line, MISTRAL_API_KEY=your-key-here or IBEX_LLM_API_KEY=your-key-here.
- Use your **own** key, don’t share it, and never paste it into the notebook: you are submitting the notebook.

Rules

- **Do not use an agent framework** (LangChain agents, LangGraph, AutoGen, CrewAI, the OpenAI Agents SDK, ...). The point is to write the loop yourself.
- Do not change the provided cells, the tools or the tool schemas.

What to submit

Submit on Blackboard the notebook agent_loop.ipynb **with outputs**, after running it top to bottom (Kernel > Restart & Run All). That single file is the whole submission; no separate report is needed.

Grading (10 points + 2 bonus)

Component	Points
Correct tool runtime, execute_tool_call (Part 2)	2
Correct agent loop: the step by hand and run_agent (Parts 2–3)	4
Question 1	2
Question 2	2
Total	10
Bonus (optional): safe_execute_tool_call and the bonus question	+2

The exact wording of the LLM's final answer is **not** graded, since it varies from run to run. We grade whether your runtime executes the tools, sends the results back to the model, and keeps looping until the model gives a final answer. The bonus is optional: implementing `safe_execute_tool_call` and answering the bonus question is worth up to 2 extra points.

Expected effort

About **30–60 minutes**, plus the time to set up the API key. If you are stuck for much longer than that, ask on Campuswire.

Late policy

10% penalty per 24 hours (or fraction thereof) up to 72 hours late, after which a zero is recorded without an approved extension.

Use of AI assistants

Using AI tools for this assignment is **not forbidden, but discouraged**: the point is to write the loop yourself. If you do use them, **disclosure is mandatory**: in the last cell of the notebook, name the tools you used and what you used them for. You remain responsible for everything you submit.