

- Examine the document cleaning code. What kinds of characters are removed when the text is cleaned?

`unicode.unidecode(text)` removes non-ASCII characters, `re.sub(r"\s+", " ", text)` removes spaces and tabs and `.split()` removes trailing whitespaces

- What size chunks is the content split into?

From `text_splitter = RecursiveCharacterTextSplitter(chunk_size=10000, chunk_overlap=1000)`, each chunk has a size of 10000 characters

- What amount of overlap is there between chunks?

From `text_splitter = RecursiveCharacterTextSplitter(chunk_size=10000, chunk_overlap=1000)`, the overlap is 1000

- How many documents are loaded at a time into the vector database?

From `add_documents(vectorstore, chunks, 300)`, it's 300 at a time.

- Show the document URLs that are returned for the following queries:
 - Tell me about Firebase ->
https://codelabs.cs.pdx.edu/labs/C09.2g_firebase/index.html?index=..%2F..cs430
 - Tell me about Terraform →
https://codelabs.cs.pdx.edu/labs/C07.1a_terraform_gb/index.html?index=..%2F..cs430
 - Tell me about Cloud Run ->
https://codelabs.cs.pdx.edu/labs/C06.2g_cloudrun_webproxy/index.html?index=..%2F..cs430
- What are the advantages and disadvantages of Firebase?

It provides authentication, and save development time as an advantage but one disadvantage is that it leads to vendor lock-in

- How do you use Terraform?

This is taken from the direct response the LLM provide to the query "Tell me about Terraform"

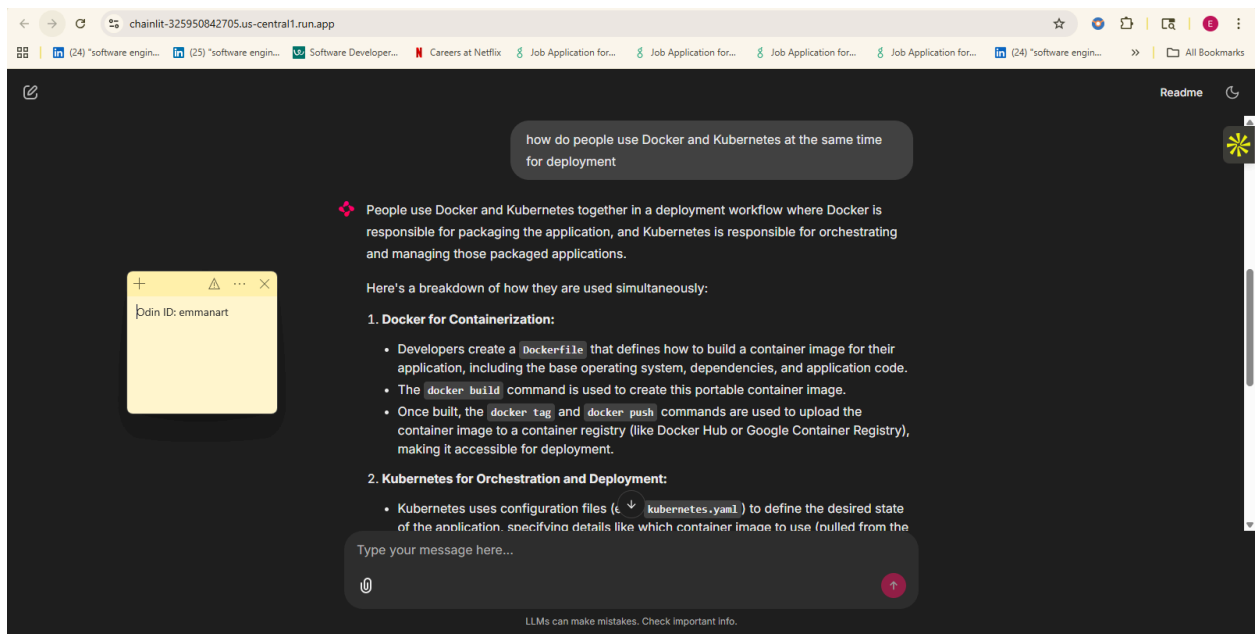
Consistent Workflow:* The typical workflow involves:

1. **Defining Configuration:** Creating declarative configuration files (using a structured data format similar to JSON and YAML) that specify the cloud resources to be created. These files are often committed to a Git repository for version control.
2. **Planning:** Running a `terraform plan` command to simulate a deployment. This step shows the potential effects of the configuration, including which resources will be added, changed, or destroyed, without actually making any changes.
3. **Applying:** Executing a `terraform apply` command to deploy the specified resources onto the cloud platform.
4. **Initializing:** Using `terraform init` to load necessary software modules (providers) that help Terraform interact with the cloud provider.
5. **Destroying:** Using `terraform destroy` to take down the deployed infrastructure.

- Why would you use Cloud Run instead of App Engine?

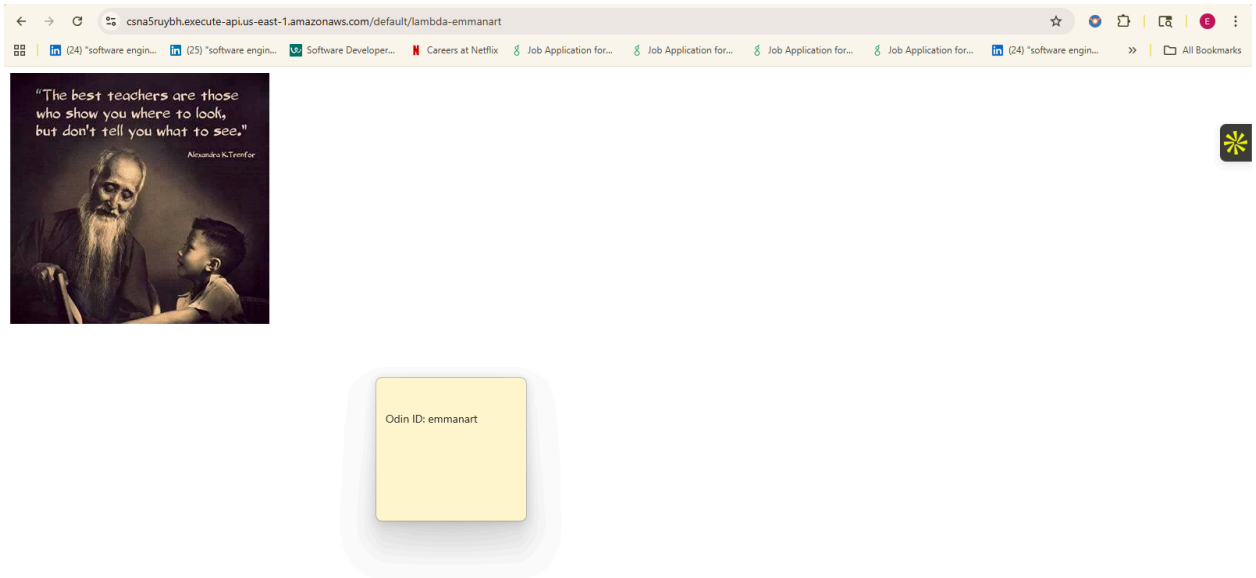
From the LLM response, Unlike App Engine, which supports a restricted set of environments, Cloud Run offers developers the flexibility to run any environment they can package into a container. This is highly beneficial for those familiar with building containers for their applications.

- Take a screenshot that includes the URL and the results of a successful query

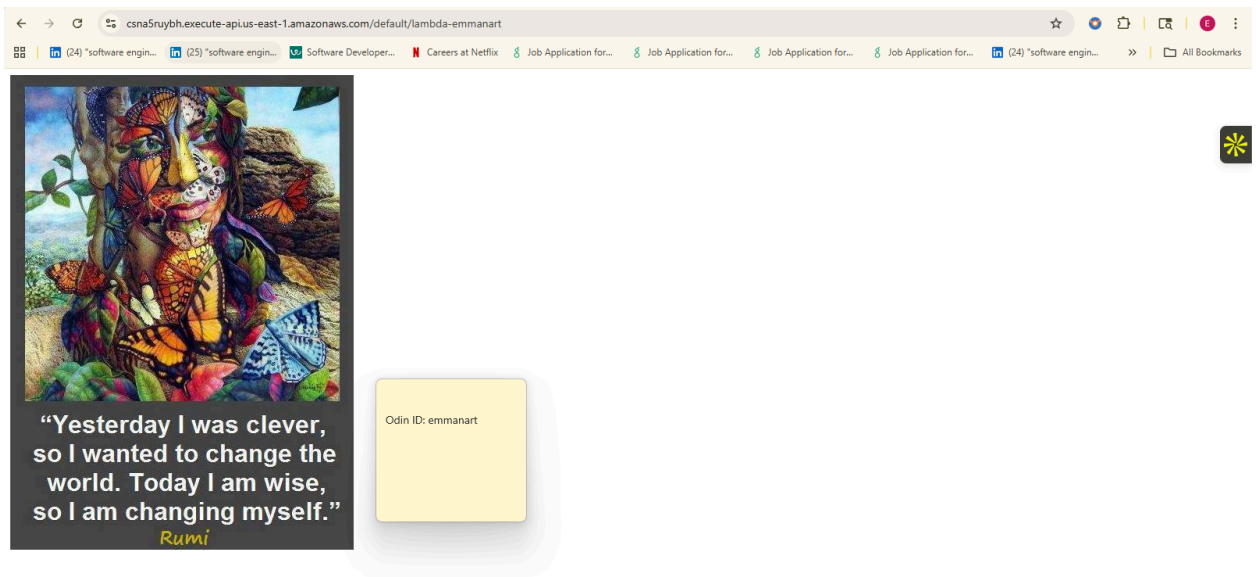


Lab 8.3a

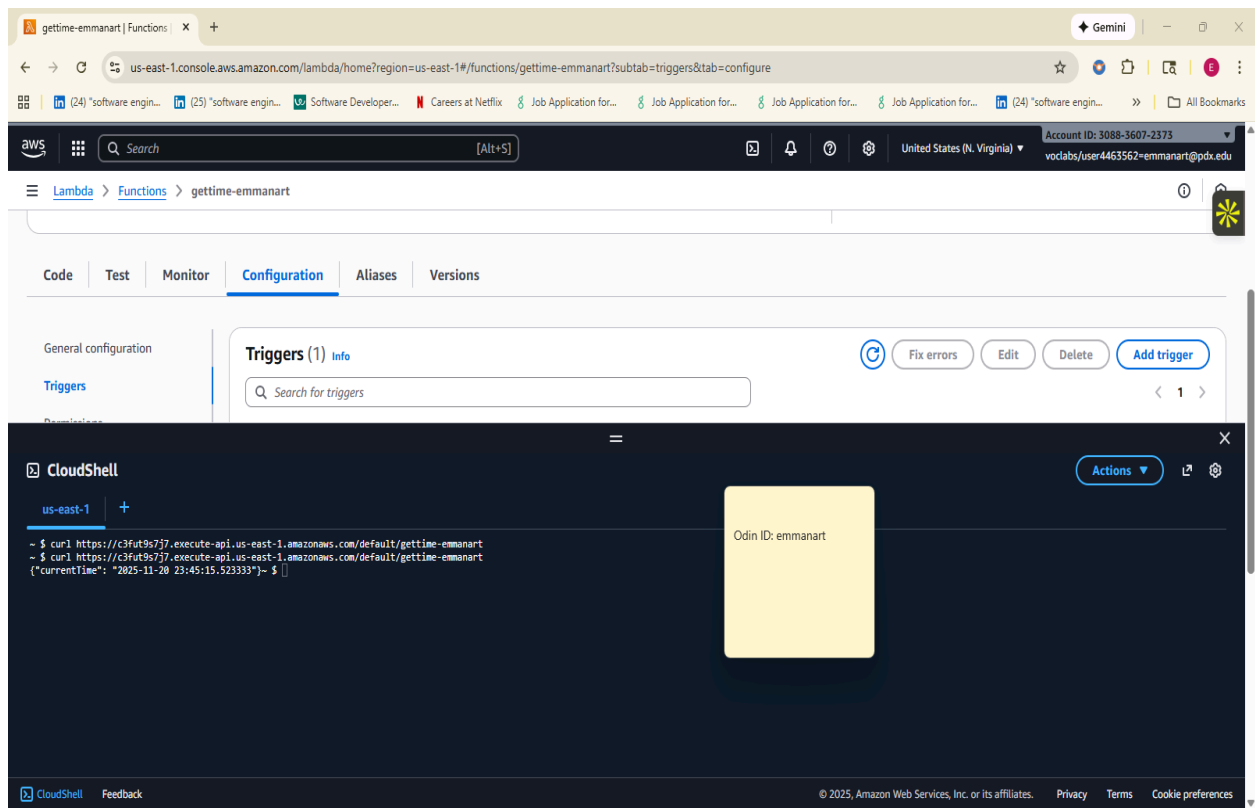
- Take a screenshot of the resulting page including the URL bar



- Click "Reload" in the browser and take another screenshot showing the image has changed:

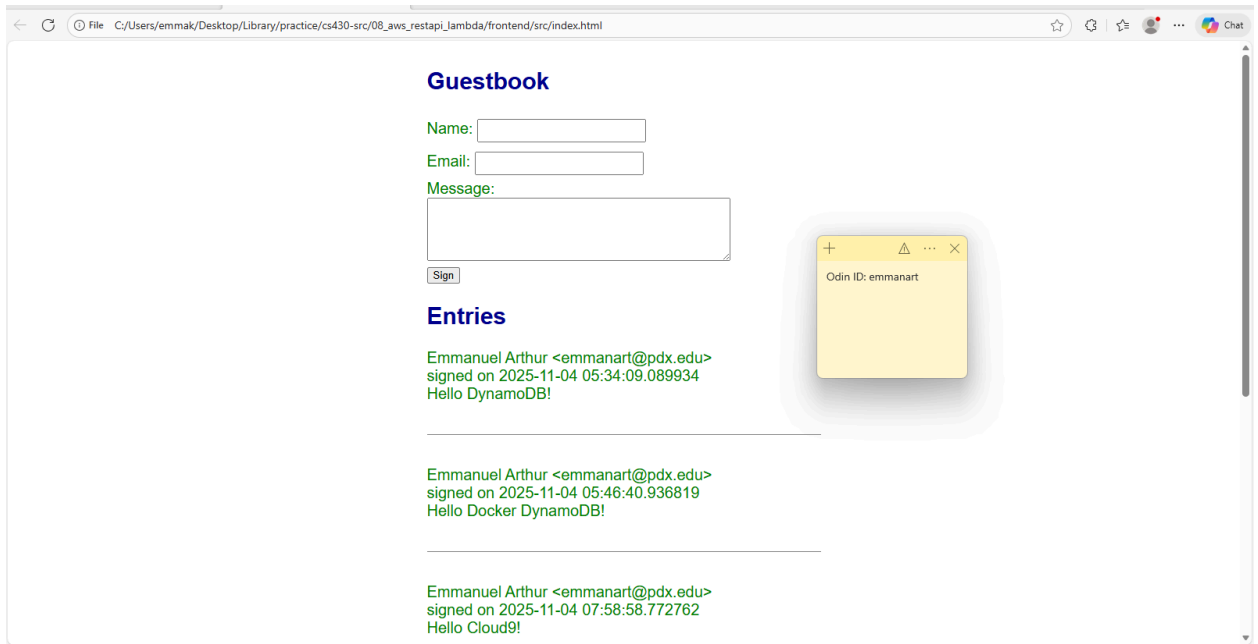


- Use `curl` on your Linux VM to access the API endpoint and show the results. Take a screenshot for your lab notebook.

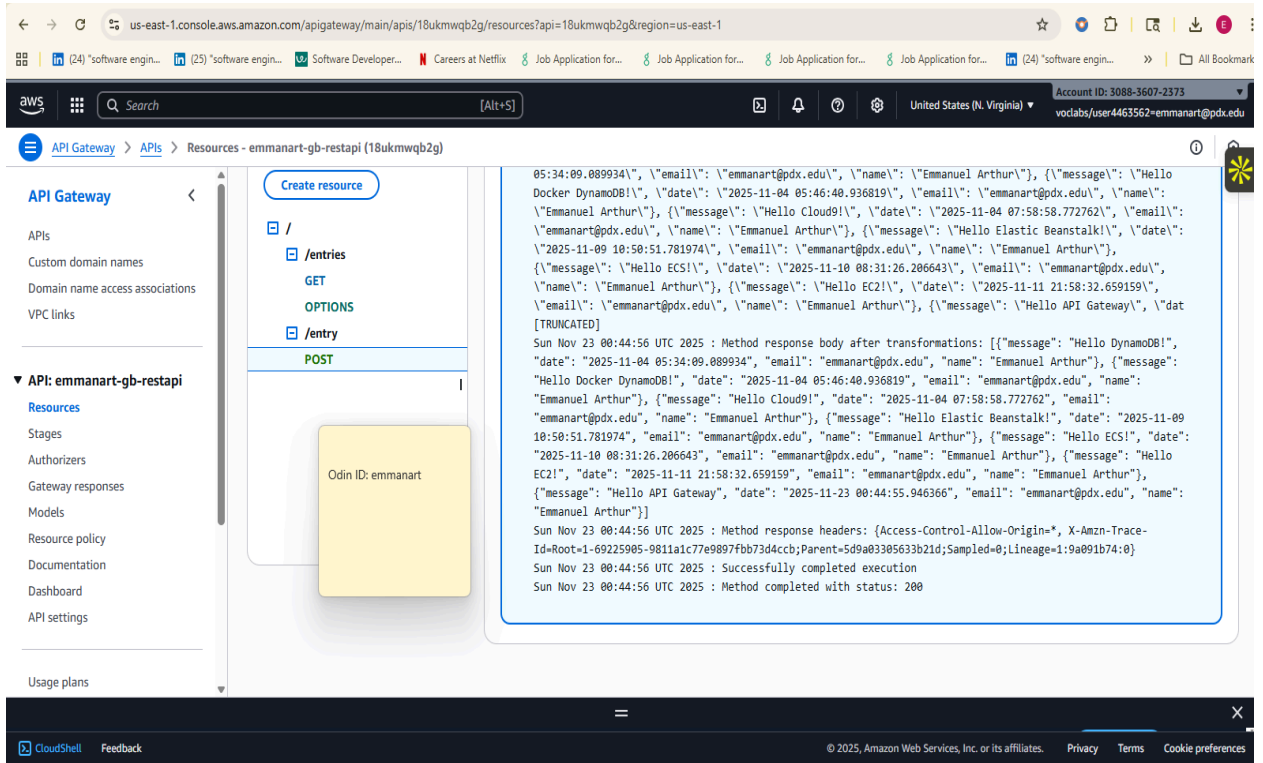


Lab 8.4a

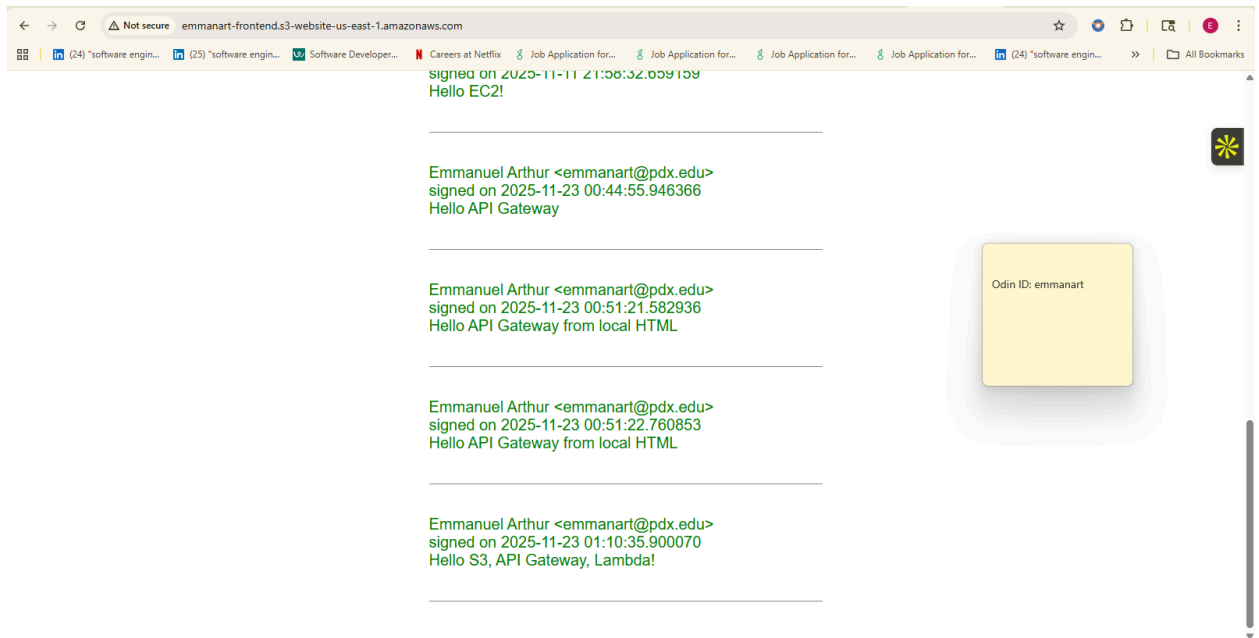
- Take a screenshot that shows that you can view the entries in the backend database.



- Take a screenshot showing that the submission worked.



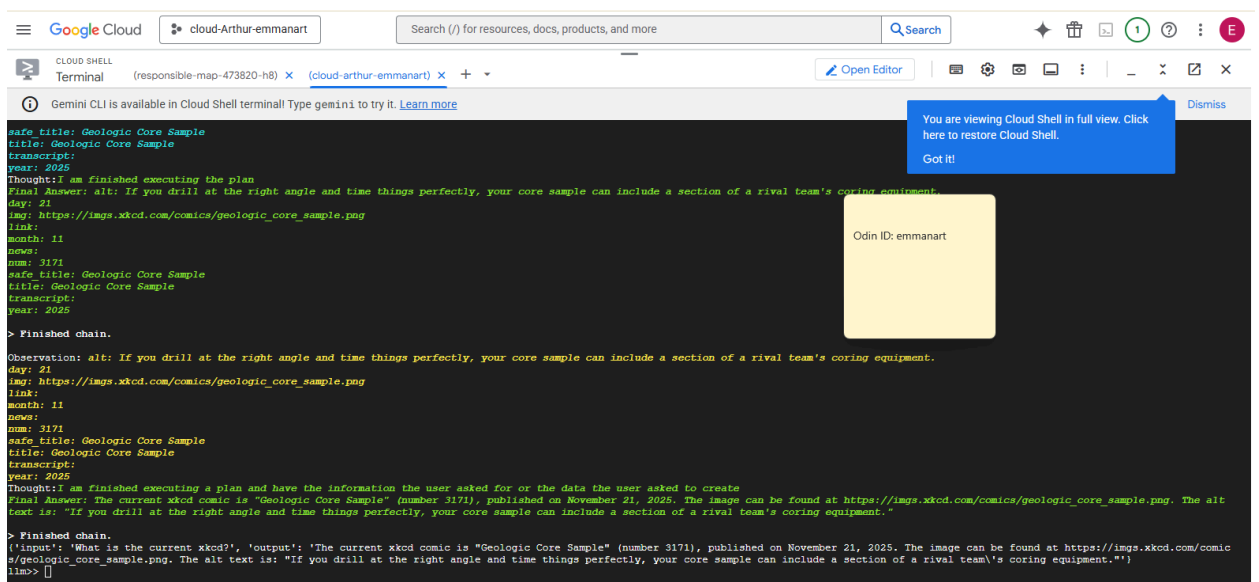
- Take a screenshot as before that shows your entry and the static website hosting URL.



Lab 8.2 g

Take a screenshot showing the results of execution

- What is the current xkcd?



- What is the image link of the current xkcd?

```

> Finished chain.
('input': 'What is the current xkcd?', 'output': 'The current xkcd comic is "Geologic Core Sample" (number 3171), published on November 21, 2025. The image can be found at https://imgs.xkcd.com/comics/geologic_core_sample.png. The alt text is: "If you drill at the right angle and time things perfectly, your core sample can include a section of a rival team\'s coring equipment."'')
llm>> What is the image link of the current xkcd?

> Entering new AgentExecutor chain...
Action: api_planner
Action Input: What is the image link of the current xkcd?
Observation: 1. GET /info.0.json to fetch the current comic's metadata, which will include the image link.
Thought: I'm ready to execute the API call.
Action: api_controller
Action Input: 1. GET /info.0.json to fetch the current comic's metadata, which will include the image link.

> Entering new AgentExecutor chain...
Action: requests_get
Action Input: {"url": "http://xkcd.com/info.0.json", "params": {}, "output_instructions": "Extract the 'num' (id), 'title' (name), 'img' (image URL), and 'alt' (alt text) from the response."}
Observation: id: 3171
name: Geologic Core Sample
image URL: https://imgs.xkcd.com/comics/geologic_core_sample.png
alt text: If you drill at the right angle and time things perfectly, your core sample can include a section of a rival team's coring equipment.
Thought: I am finished executing the plan (or, I cannot finish executing the plan without knowing some other information.)
Final Answer: id: 3171
name: Geologic Core Sample
image URL: https://imgs.xkcd.com/comics/geologic_core_sample.png
alt text: If you drill at the right angle and time things perfectly, your core sample can include a section of a rival team's coring equipment.

> Finished chain.
Observation: id: 3171
name: Geologic Core Sample
image URL: https://imgs.xkcd.com/comics/geologic_core_sample.png
alt text: If you drill at the right angle and time things perfectly, your core sample can include a section of a rival team's coring equipment.
Thought: I am finished executing a plan and have the information the user asked for or the data the user asked to create
Final Answer: The image link of the current xkcd comic is: https://imgs.xkcd.com/comics/geologic_core_sample.png

> Finished chain.
('input': 'What is the image link of the current xkcd?', 'output': 'The image link of the current xkcd comic is: https://imgs.xkcd.com/comics/geologic_core_sample.png')
llm>>

```

- What was xkcd 327 about?

```

Mom: Oh, yes. Little Bobby Tables, we call him.
Phone: Well, we've lost this year's student records. I hope you're happy.
Mom: And I hope you've learned to sanitize your database inputs.
((title-text: Her daughter is named Help I'm trapped in a driver's license factory.))
**Link:** (empty)
**News:** (empty)

> Finished chain.
Observation: Here is the information about the comic:
* **ID (num):** 327
* **Title:** Exploits of a Mom
* **Safe Title:** Exploits of a Mom
* **Day:** 10
* **Month:** 10
* **Year:** 2007
* **Image URL:** https://imgs.xkcd.com/comics/exploits_of_a_mom.png
* **Alt Text:** Her daughter is named Help I'm trapped in a driver's license factory.
* **Transcript:** [[A woman is talking on the phone, holding a cup]]
Phone: Hi, this is your son's school. We're having some computer trouble.
Mom: Oh dear-did he break something?
Phone: In a way.
Phone: Did you really name your son "Robert"? DROP TABLE Students;--"?
Mom: Oh, yes. Little Bobby Tables, we call him.
Phone: Well, we've lost this year's student records. I hope you're happy.
Mom: And I hope you've learned to sanitize your database inputs.
((title-text: Her daughter is named Help I'm trapped in a driver's license factory.))
**Link:** (empty)
**News:** (empty)
Thought: I am finished executing a plan and have the information the user asked for or the data the user asked to create
Final Answer: The xkcd comic 327 is titled "Exploits of a Mom". It was published on October 10, 2007. The comic is about a mom who named her son "Robert"; DROP TABLE Students;--" and the school losing their student records because they didn't sanitize their database inputs. The alt text for the comic is: "Her daughter is named Help I'm trapped in a driver's license factory."

> Finished chain.
('input': 'What was xkcd 327 about?', 'output': 'The xkcd comic 327 is titled "Exploits of a Mom". It was published on October 10, 2007. The comic is about a mom who named her son "Robert"; DROP TABLE Students;--" and the school losing their student records because they didn\'t sanitize their database inputs. The alt text for the comic is: "Her daughter is named Help I\'m trapped in a driver\'s license factory."'')
llm>>

```

- For these questions sent to the agent to interact with the database they produced the error below:
Find the admin user in the database and output the password hash.

What password hash algorithm and number of iterations are being used to store hashes?

```
0, 'safety_ratings': [], 'finish_reason': 'STOP', 'model_name': 'gemini-2.5-flash', 'safety_ratings': [], 'model_provider': 'google_genai', id='lc_run--48b00c20-49e7-42ee-9a2b-87ba55194d69-0', tool_calls=[{'name': 'query', 'args': {'query': 'SELECT DISTINCT hash_algorithm, iterations FROM users'}, 'id': 'f4f3bc04-d4c0-48f4-9b40-df699cb2c921', 'type': 'tool_call'}, usage_metadata={'input_tokens': 113, 'output_tokens': 189, 'total_tokens': 302, 'input_token_details': {'cache_read': 0}, 'output_token_details': {'reasoning': 168}})]))\n\n[values] {'messages': [HumanMessage(content='What password hash algorithm and number of iterations are being used to store hashes?', additional_kwargs={}, response_metadata={}, id='71cbe639-51f6-4453-9612-f2d12d392f22'), AIMessage(content='', additional_kwargs={'function_call': {'name': 'query', 'arguments': {'query': 'SELECT DISTINCT hash_algorithm, iterations FROM users'}}}, 'gemini_function_call_thoughts': {'thoughts': [{'id': 'f4f3bc04-d4c0-48f4-9b40-df699cb2c921', 'query': 'SELECT DISTINCT hash_algorithm, iterations FROM users', 'id': 'f4f3bc04-d4c0-48f4-9b40-df699cb2c921', 'type': 'tool_call'}, usage_metadata={'input_tokens': 113, 'output_tokens': 189, 'total_tokens': 302, 'input_token_details': {'cache_read': 0}, 'output_token_details': {'reasoning': 168}})]))\n\nError calling tool 'query'\n\nTraceback (most recent call last):\n  /home/emmanart/cs430-src/07_agents/.venv/lib/python3.12/site-packages/fastmcp/tools/tool_manager.py:160 in call_tool\n    \n  /home/emmanart/cs430-src/07_agents/.venv/lib/python3.12/site-packages/fastmcp/tools/tool.py:330 in run\n    \n  /home/emmanart/cs430-src/07_agents/.venv/lib/python3.12/site-packages/pydantic/type_adapter.py:441 in validate_python\n    \n  /home/emmanart/cs430-src/07_agents/02_sqlite/vulnerable_sqlite_mcp_server.py:13 in query\n    \n    10 | database = *.db\n    11 | con = sqlite3.connect(database)\n    12 | cur = con.cursor()\n    13 | results = cur.execute(query)\n    14 | con.commit()\n    15 | output_string = '\\n'.join(['', '.join(map(str, row)) for row in re\n    16 | return output_string\n\nOperationalError: no such column: hash_algorithm\nError: Error calling tool 'query': no such column: hash_algorithm\nllm> ]
```

- List the passhash of **foo** or **1=1--**

Same error as image above

- Drop the users table

Successfully dropped

- Take a screenshot of the deployed service that includes your OdinID

The screenshot shows the Google Cloud console interface. At the top, the browser address bar displays `console.cloud.google.com/run?referrer=search&cloudshell=true&project=cloud-arthur-emmanart`. The main header includes the Google Cloud logo, the project name `cloud-arthur-emmanart`, and a search bar. Below the header, the 'Services' section is active, displaying a table of services. The table has columns for Name, Deployment type, Req/sec, Region, Authentication, Ingress, Last deployed, Deployed by, and Recommendation. The services listed are `gcp-gb`, `hw4`, `python-blur-function`, and `svc-02-sqlite`. A yellow tooltip shows 'Odin ID: emmanart'. A blue tooltip offers to share the recommendation details URL. At the bottom, a terminal window is open, showing the command `https://cloud.google.com/run/docs` and the output `(✓) Your application is now live here: https://svc-02-sqlite-ydpsmikaqz-9g-a.run.app`.

Name	Deployment type	Req/sec	Region	Authentication	Ingress	Last deployed	Deployed by	Recommendation
<code>gcp-gb</code>	Container	0	us-west1	Allow unauthenticated	All	Nov 9, 2025	emmanart@pdx.edu	—
<code>hw4</code>	Container	0	us-west1	Allow unauthenticated	All	11 days ago	emmanart@pdx.edu	Security
<code>python-blur-function</code>	Function	0	us-west1	Require authentication	All	Nov 9, 2025	emmanart@pdx.edu	—
<code>svc-02-sqlite</code>	Container	0	us-east1	Allow unauthenticated	All	2 minutes ago	emmanart@pdx.edu	—

Odin ID: emmanart

Share this recommendation with others
Copy the recommendation details URL to the clipboard.

Terminal
cloudshell
Gemini CLI is available in Cloud Shell terminal! Type `gemini` to try it. [Learn more](#)

```
https://cloud.google.com/run/docs
( ✓ ) Your application is now live here:
https://svc-02-sqlite-ydpsmikaqz-9g-a.run.app
emmanart@cloudshell:~$
```

- Take a screenshot of the request including the HTTP method and User-Agent used by the MCP client to access the MCP server

Google Cloud cloud-Arthur-emmanart Search (/) for resources, docs, products, and more Search

Service details Edit & deploy new revision Connect to repo Test Learn Refresh

svc-02-sqlite Region: us-east1 URL: <https://svc-02-sqlite-325950842705.us-east1.run.app> Scaling: Auto (Min: 0, Max: 20)

Observability Revisions Source Triggers Networking Security YAML

Metrics Logs SLOs Errors

Odin ID: emmanart

Logs Severity: Default Filter Search all fields and values

Severity	Timestamp	Summary
INFO	2025-11-23 11:52:31.172 PST	169.254.169.126:16470 - "POST /mcp HTTP/1.1" 200 OK

Explain this log entry Open in Logs Explorer

```
{
  insertId: "692365ff0002a0696d991e1b"
  labels: (1)
    logName: "projects/cloud-arthur-emmanart/logs/run.googleapis.com%2Fstdout"
    payload: "textPayload"
    receiveLocation: "us-east1"
    receiveTimestamp: "2025-11-23T19:52:31.497486896Z"
  resource: (2)
    severity: "DEFAULT"
    textPayload: "INFO: 169.254.169.126:16470 - \"POST /mcp HTTP/1.1\" 200 OK"
    timestamp: "2025-11-23T19:52:31.172137Z"
    traceSampled: false
}
```

2025-11-23 11:55:56.543 PST Truncated response body. Usually implies that the request timed out or the application exited b...

No newer entries found matching current filter.

svc-02-sqlite Region: us-east1 URL: <https://svc-02-sqlite-325950842705.us-east1.run.app> Scaling: Auto (Min: 0, Max: 20)

Observability Revisions Source Triggers Networking Security YAML

Metrics Logs SLOs Errors

Odin ID: emmanart

Logs Severity: Default Filter Search all fields and values

Severity	Timestamp	Summary
INFO	2025-11-23 11:50:48.574 PST	169.254.169.126:13210 - "POST /mcp HTTP/1.1" 200 OK
INFO	2025-11-23 11:50:48.589 PST	169.254.169.126:13210 - "POST /mcp HTTP/1.1" 200 OK
INFO	2025-11-23 11:50:54.037 PST	169.254.169.126:37240 - "GET /mcp HTTP/1.1" 200 OK
INFO	2025-11-23 11:50:54.041 PST	169.254.169.126:37240 - "GET /mcp HTTP/1.1" 200 OK
INFO	2025-11-23 11:50:59.051 PST	169.254.169.126:37250 - "POST /mcp HTTP/1.1" 202 Accepted
INFO	2025-11-23 11:50:59.055 PST	169.254.169.126:37250 - "POST /mcp HTTP/1.1" 202 Accepted
INFO	2025-11-23 11:50:59.142 PST	169.254.169.126:37266 - "POST /mcp HTTP/1.1" 200 OK
INFO	2025-11-23 11:50:59.148 PST	169.254.169.126:37266 - "POST /mcp HTTP/1.1" 200 OK
INFO	2025-11-23 11:52:07.637 PST	169.254.169.126:39782 - "POST /mcp HTTP/1.1" 200 OK
INFO	2025-11-23 11:52:07.648 PST	169.254.169.126:39782 - "POST /mcp HTTP/1.1" 200 OK
INFO	2025-11-23 11:52:20.498 PST	169.254.169.126:40456 - "POST /mcp HTTP/1.1" 200 OK
INFO	2025-11-23 11:52:20.503 PST	169.254.169.126:40456 - "POST /mcp HTTP/1.1" 200 OK
INFO	2025-11-23 11:52:31.167 PST	169.254.169.126:16470 - "POST /mcp HTTP/1.1" 200 OK
INFO	2025-11-23 11:52:31.172 PST	169.254.169.126:16470 - "POST /mcp HTTP/1.1" 200 OK
INFO	2025-11-23 11:55:56.543 PST	Truncated response body. Usually implies that the request timed out or the application exited b...

Lab 8.4g

- Include the hostname for the API gateway in your lab notebook.

gbapigw-45qmprv5.uc.gateway.dev

- Take a screenshot of its output

```

SyntaxError: invalid syntax
>>> for key in resp_json[0]:
...     print("Name: ", resp_json[0]['name'])
File "<stdin>", line 2
    print("Name: ", resp_json[0]['name'])
    ^
SyntaxError: invalid syntax
>>> for key in resp_json[0]:
...     print("Name: ", resp_json[0]['name'])
File "<stdin>", line 2
    print("Name: ", resp_json[0]['name'])
    ^
SyntaxError: invalid syntax
>>> for key in resp_json[0]:
...     print("Name: ", resp_json[0]['name'], "\n", "Email: ", resp_json[0]['email'], '\n', "Date: ", resp_json[0]['date'], '\n', "Message: ", resp_json[0]['message'], '\n' )
...
Name: Emmanuel Arthur
Email: emmanart@pdx.edu
Date: 2025-11-04 07:13:08.151893+00:00
Message: Hello Docker Datastore!

Name: Emmanuel Arthur
Email: emmanart@pdx.edu
Date: 2025-11-04 07:13:08.151893+00:00
Message: Hello Docker Datastore!

Name: Emmanuel Arthur
Email: emmanart@pdx.edu
Date: 2025-11-04 07:13:08.151893+00:00
Message: Hello Docker Datastore!

Name: Emmanuel Arthur
Email: emmanart@pdx.edu
Date: 2025-11-04 07:13:08.151893+00:00
Message: Hello Docker Datastore!

>>>

```

Odin ID: emmanart

- Take a screenshot of the output for your lab notebook
- Take a screenshot showing the preflight request to the API that allows API access, as well as the subsequent fetch request have been successful. Note that this request may not appear in Firefox browsers.

Guestbook

Name:

Email:

Message:

Entries

Emmanuel Arthur <emmanart@pdx.edu>
signed on 2025-11-04 07:13:08.151893+00:00
Hello Docker Datastore!

Odin ID: emmanart

Emmanuel Arthur <emmanart@pdx.edu>
signed on 2025-11-09 11:01:51.940701+00:00
Hello App Engine!

Emmanuel Arthur <emmanart@pdx.edu>

Name	Status	Type	Initiator	Size	Time
index.html	200	document	Other	0.9 kB	6 ms
style.css	200	stylesheet	index.html#6	0.6 kB	18 ms
guestbook.js	200	script	index.html#7	2.4 kB	22 ms
entries	200	fetch	guestbook.js#8	0.4 kB	183 ms
entries	200	preflight	Preflight	0.0 kB	87 ms

5 requests | 4.2 kB transferred | 5.0 kB resources | Finish: 318 ms | DOMContentLoaded: 49 ms | Load: 50 ms

How can I help you?

Why is this network request taking so long?

Are there any security headers present?

- Take a screenshot of the Guestbook including the URL.

