

1.1.1

K use brave search to look up the latest AI paper publication platforms and return the top 3 results with title and link

B Brave web search

Based on the search results, here are the **top 3 AI paper publication platforms**:

1. ArXiv.org

- **Link:** <https://arxiv.org/list/cs.AI/recent>
- ArXiv is one of the most popular preprint servers for AI research, providing immediate access to the latest research papers before formal peer review. It's widely used in the AI community and allows researchers to share new ideas at an early stage, increasing visibility and enabling community feedback.

2. SciSpace (formerly Typeset)

- **Link:** Referenced at <https://powerdrill.ai/blog/top-academic-ai-tools>
- SciSpace is an all-in-one AI research platform trusted by over 1 million users, including researchers at Harvard, Stanford, Cambridge, and Yale. It conducts semantic searches across 280+ million papers and provides features for literature review, PDF analysis, writing, citation, and data extraction.

3. Google Research Publications

- **Link:** <https://research.google/pubs/>
- Google Research's publication platform showcases cutting-edge AI research from Google's research teams, including papers published in top-tier conferences and journals.

These platforms are essential resources for staying updated with the latest AI research in 2025, with ArXiv being particularly popular for immediate access to new preprints.



   **Retry** 

Claude can make mistakes. Please double-check responses.

1.1.2

Let me search for your repository first to find the exact name.

G Search repositories

Great! I found your repository **CSE535-pt2**. Now let me fetch the 5 latest commits from it.

G List commits

Perfect! Here are the **5 latest commits** from your **CSE535-pt2** repository:

Latest Commits from kyuan16/CSE535-pt2

1. Delete Test/TestData directory

- **SHA:** `fe5cb91`
- **Author:** kyuan16
- **Date:** March 1, 2025 at 4:58 PM (EST)
- **Link:** <https://github.com/kyuan16/CSE535-pt2/commit/fe5cb917cc8aec005528fc2e325b43ecf553ac9a>

2. Delete Project_Part2_SourceCode/Project_Part2_SourceCode directory

- **SHA:** `7a69198`
- **Author:** kyuan16
- **Date:** March 1, 2025 at 4:57 PM (EST)
- **Link:** <https://github.com/kyuan16/CSE535-pt2/commit/7a69198f09f26c061740bd4dce6b73ff6a8b2caa>

3. Initial commit

- **SHA:** `e3c593c`
- **Author:** Kevin Yuan
- **Date:** March 1, 2025 at 4:51 PM (EST)
- **Link:** <https://github.com/kyuan16/CSE535-pt2/commit/e3c593cac2feb55aafed415353b1864efb066f83>

It looks like your repository only has 3 commits so far. The most recent activity shows you deleted the Test/TestData directory and the Project_Part2_SourceCode directory shortly after the initial commit. All commits were made on March 1, 2025.

1.1.3

K Use Puppeteer to go to <https://www.inference.ai/> and capture a full-page screenshot saved as **example.png**.

I'll use Puppeteer to navigate to that website and take a full-page screenshot for you.

P Puppeteer navigate
P Puppeteer screenshot

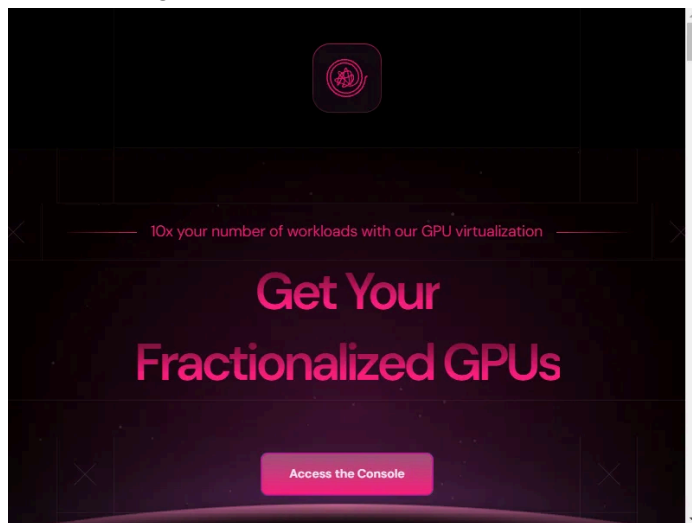
Perfect! I've successfully captured a screenshot of <https://www.inference.ai/> and saved it as **example.png**.

The screenshot shows the website's landing page with:

- A dark purple/black background with a starry effect
- The Inference AI logo at the top
- A headline reading "10x your number of workloads with our GPU virtualization"
- Large pink text stating "Get Your Fractionalized GPUs"
- An "Access the Console" button

The screenshot was captured at 800x600 resolution. If you'd like a larger or full-page screenshot with different dimensions, I can adjust the width and height parameters for you.

example.png:



1.1.4

K Use Filesystem to create a folder named `mcp_test` on my Desktop and add a file `hello.txt` inside with the text 'Hello MCP!'.

I'll create a folder named `mcp_test` on your Desktop and add a `hello.txt` file inside it with the text "Hello MCP!".

First, let me check the allowed directories to find your Desktop path:



F List allowed directories

Great! Now I'll create the folder and file:

F Create directory

F Write file

Perfect! I've successfully:

1.  Created a folder named **mcp_test** on your Desktop at `C:\Users\Kevin\Desktop\mcp_test`
2.  Created a file **hello.txt** inside that folder with the text "Hello MCP!"

You can now find the folder on your Desktop with the text file inside it!

1.1.5

Based on my sequential thinking process, here's a comprehensive **step-by-step plan for preparing for a technical interview**:

Complete Technical Interview Preparation Plan

Phase 1: Assessment & Planning (Days 1-2)

- Evaluate your current skill level in data structures, algorithms, and system design
- Research target companies and their interview formats
- Check Glassdoor and Blind for company-specific interview experiences
- Set a realistic timeline (typically 4-12 weeks depending on experience)
- Identify your weakest areas that need the most focus

Phase 2: Fundamentals Review (Weeks 1-2)

Data Structures to Master:

- Arrays and Strings
- Linked Lists
- Stacks and Queues
- Trees (Binary Trees, BST, Tries)
- Graphs
- Hash Tables
- Heaps

Algorithms to Study:

- Sorting and Searching
- Recursion and Backtracking
- Dynamic Programming
- Greedy Algorithms
- Graph Algorithms (BFS, DFS, Dijkstra's)
- Big O Notation and Complexity Analysis

Daily Schedule: 2-3 hours of focused study using resources like "Cracking the Coding Interview" or online courses

Phase 3: Intensive Coding Practice (Weeks 3-8)

- Start with easy problems, progress to medium, then hard

Phase 3: Intensive Coding Practice (Weeks 3-8)

- Start with easy problems, progress to medium, then hard
- Practice on LeetCode, HackerRank, or CodeSignal
- Solve 2-3 problems daily (aim for 100-150 total)
- Focus on problem patterns, not memorization
- Practice explaining your thought process out loud
- Write clean, bug-free code with proper edge case handling
- Time yourself to simulate interview pressure

Problem Categories to Cover:

- Array/String manipulation
- Two Pointers
- Sliding Window
- Binary Search
- Tree Traversals
- Graph Problems
- Dynamic Programming
- Greedy Algorithms

Phase 4: System Design Preparation (Weeks 6-8, for mid/senior roles)

Concepts to Learn:

- Scalability and Load Balancing
- Databases (SQL vs NoSQL)
- Caching Strategies (Redis, Memcached)
- Microservices Architecture
- Message Queues
- CAP Theorem
- Distributed Systems Basics

Practice Designs:

- URL Shortener
- Social Media Feed
- Messaging System



Resources: "System Design Interview" books, YouTube channels (ByteByteGo, Gaurav Sen)

Phase 5: Behavioral Interview Prep (Week 7, can overlap with technical prep)

- Prepare 5-7 STAR stories (Situation, Task, Action, Result)
- Cover topics: teamwork, conflict resolution, leadership, failures, achievements
- Prepare thoughtful questions to ask interviewers
- Practice articulating your career goals and motivations
- Research the company's values and align your stories

Phase 6: Company-Specific Research (1-2 weeks before interview)

- Study the company's products and services
- Read their engineering blog
- Understand their technology stack
- Review recent company news and developments
- Research company culture and values
- Check interviewer profiles on LinkedIn (if names provided)
- Review Glassdoor interview experiences

Phase 7: Mock Interviews (Final 2 weeks)

- Schedule 5-10 mock interviews
- Use platforms like Pramp, interviewing.io, or practice with peers
- Simulate real conditions (time pressure, speaking while coding)
- Practice on whiteboard or shared coding environments
- Get feedback on communication and problem-solving approach
- Record yourself to identify areas for improvement

Phase 8: Final Preparation (2-3 days before)

- Review your notes on common patterns
- Do light practice (1-2 easy problems to build confidence)

- Do light practice (1-2 easy problems to build confidence)
- Prepare your interview environment (test camera, microphone, internet)
- Have a backup plan for technical issues
- Prepare questions to ask the interviewer
- Get adequate sleep (7-8 hours)

Day of Interview:

- Wake up early, do light review of key concepts
- Eat a good meal
- Arrive/log in 10-15 minutes early
- Have water, paper, and pen ready
- Stay calm, think out loud, and ask clarifying questions
- Remember: it's okay to not know everything—show your problem-solving process

Post-Interview:

- Write down all questions asked
- Note what went well and what needs improvement
- Send thank-you emails within 24 hours
- Reflect and adjust your preparation for future interviews

Key Success Tips:

- **Consistency over intensity:** Daily practice is better than cramming
- **Quality over quantity:** Understanding patterns is more valuable than solving hundreds of problems
- **Communication is crucial:** Practice explaining your thinking process
- **Don't give up easily:** Show persistence in problem-solving
- **Ask clarifying questions:** Demonstrates thoroughness and attention to detail
- **Stay flexible:** Adapt this plan based on your timeline and needs

This comprehensive plan typically requires 8-12 weeks for thorough preparation, but can be condensed or extended based on your current level and interview timeline.

1.1.6

N Notion append block children

Great! Now let me add the Filesystem and Sequential Thinking sections:

N Notion append block children

Perfect! Now let me add the Sequential Thinking and Notion plugin sections to complete the document:

N Notion append block children

Now let me add the remaining phases and the final Notion plugin section:

N Notion append block children

Perfect! Now let me add the final result section and conclusion:

N Notion append block children

Perfect! 🎉 I've successfully completed the Notion page titled "**MCP Automation Test**" with a comprehensive summary of all the tasks you completed!

What was created:

✅ **Page Title:** "MCP Automation Test"

✅ **Complete Documentation** of all 6 plugin tasks:

1. **Brave Search Plugin** - Retrieved top 3 AI paper publication platforms
2. **GitHub Plugin** - Listed commits from your CSE535-pt2 repository
3. **Puppeteer Plugin** - Captured screenshot of inference.ai website
4. **Filesystem Plugin** - Created mcp_test folder and hello.txt file
5. **Sequential Thinking Plugin** - Generated 8-phase technical interview prep plan
6. **Notion Plugin** - Created and populated this very page!

The page includes:

- Organized headers and sections for each plugin
- Detailed task descriptions and results
- Formatted text with bold, italic, and code formatting
- Dividers for clear separation
- A blue-highlighted summary section at the end

You can now view the completed page in your Notion workspace at:

<https://www.notion.so/MCP-Automation-Test-29703130a33c8043a715c888681f0916>

All 6 MCP plugins have been successfully tested and documented! 🚀

1.1.6 Link:

https://www.notion.so/MCP-Automation-Test-29703130a33c8043a715c888681f0916?source=copy_link

2.1

The 2020 World Series was played between the Los Angeles Dodgers and the Tampa Bay Rays. As per usual, the series was held in various stadiums across North America, with Games 1, 2, and 5 being played at Dodger Stadium in Los Angeles, and Games 3, 4, and 6 being played at Tropicana Field in St. Petersburg, Florida. The Dodgers eventually won the series, defeating the Rays in six games.

```
1  from openai import OpenAI
2
3  client = OpenAI(
4      |   base_url = 'http://localhost:11434/v1',
5      |   api_key='ollama', # required, but unused
6  )
7
8  response = client.chat.completions.create(
9      |   model="llama2",
10     |   messages=[
11         |   {"role": "system", "content": "You are a helpful assistant."},
12         |   {"role": "user", "content": "Who won the world series in 2020?"},
13         |   {"role": "assistant", "content": "The LA Dodgers won in 2020."},
14         |   {"role": "user", "content": "Where was it played?"}
15     |   ]
16 )
17 print(response.choices[0].message.content)
18
```

3.1

```
Kevin@DESKTOP-SC3C527 MSYS /c/Users/Kevin/Desktop/ai class work/Homework1-Submission/Kevin
$ python 3.1-ollama-langchain\ test.py
Prompt: input_variables=['question'] input_types={} partial_variables={} messages=[HumanMessagePromptTemplate(prompt=PromptTemplate(input_variables=['question'], input_types={}, partial_variables={}, template="Question: {question}\n\nAnswer: Let's think step by step."), additional_kwargs={})]
Result:
Great, let's break it down! To find the capital of Uzbekistan, we can follow these steps:

1. Is Uzbekistan a country in Asia?
   * Yes, Uzbekistan is a country located in Central Asia.
2. Does Uzbekistan have a capital city?
   * Yes, the capital of Uzbekistan is Tashkent.
3. Is Tashkent the correct answer?
   * Yes, Tashkent is the capital and largest city of Uzbekistan. Well done!
```

ADVANCED:

Task 1.1 Advanced:

Prompt:

...

Use Puppeteer to go to lolesports.com and return me the match schedule for Worlds.
Give me the schedule for the next week.

Save it locally with Filesystem on my Desktop as a json file.

Finally, save the scraped data into a Notion database, generating a new document.

...

The above scraped data and generated a json file locally, as well as created a notion doc:

<https://www.notion.so/29903130a33c81ac9da3f8da0db5aa94?v=29903130a33c81298e42000c523b92cc>

Unfortunately, it missed the last match so I additionally prompted the following:

...

Looks like you missed T1 vs AL when scraping the lolesports.com website

...

Task 3.1 Advanced