2025.05.27.(수)

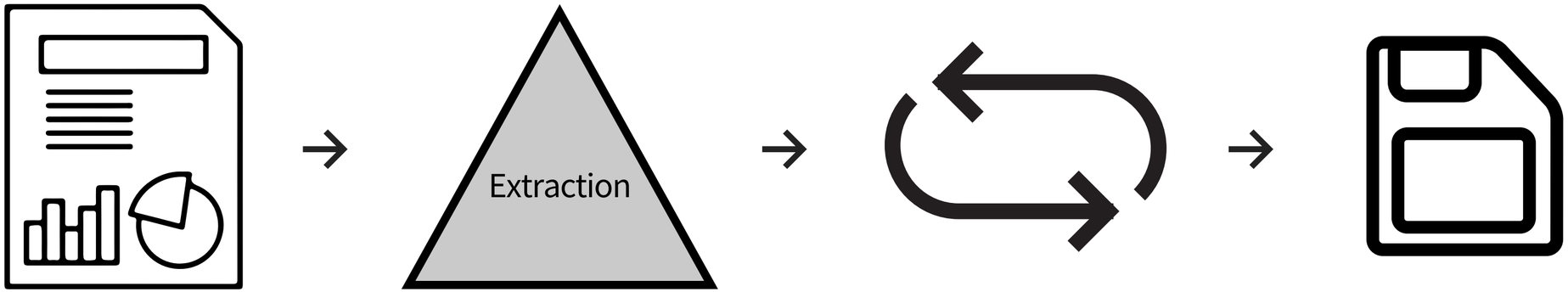
NS3 Code Generator

: LLM 기반의 네트워크 시뮬레이션 코드 자동 생성

202101109 박수화
202102722 최희원

목차

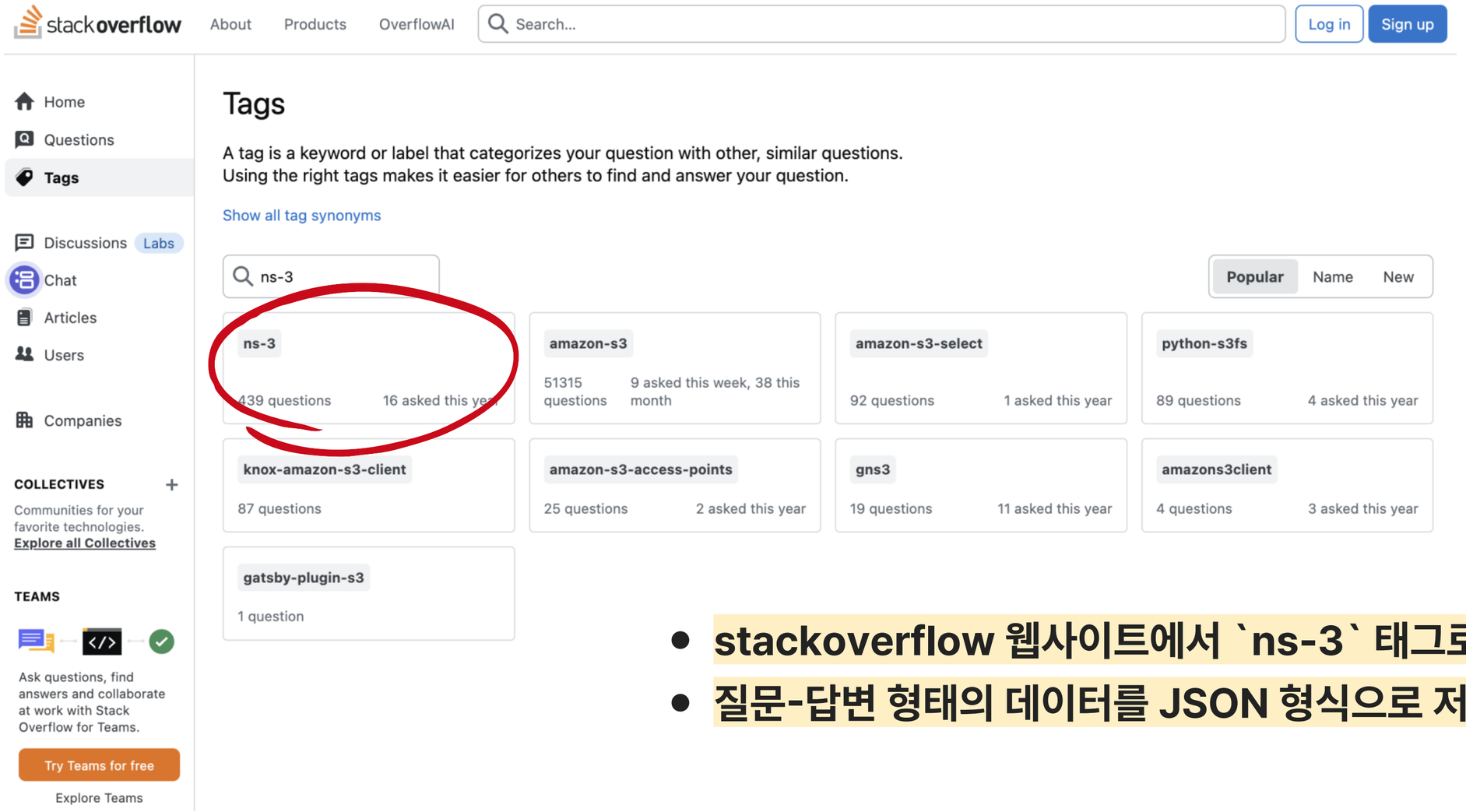
RAG DB 구축 흐름



수집 → 추출 → 변환 → 저장

Data Crawling (stackoverflow)

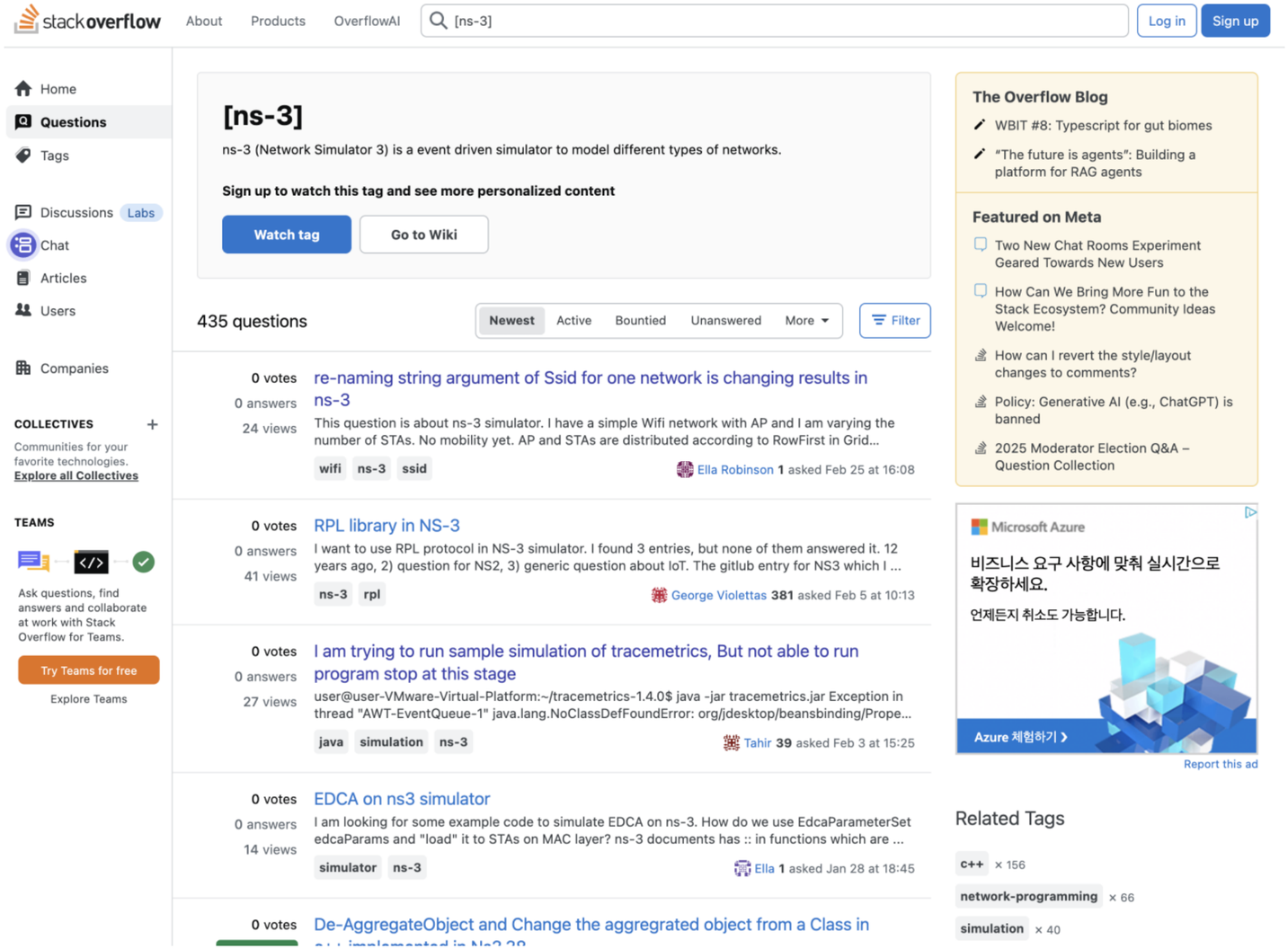
NS-3 태그 기반 Stack Overflow Q&A 데이터 수집



- stackoverflow 웹사이트에서 `ns-3` 태그로 분류된 게시물 수집
- 질문-답변 형태의 데이터를 JSON 형식으로 저장

Data Crawling (stackoverflow)

NS-3 태그 기반 Stack Overflow Q&A 데이터 수집



Data Crawling (stackoverflow)

Crawling 방법

- stackoverflow 웹사이트에서 `ns-3` 태그로 분류된 게시물 수집
- 질문-답변 형태의 데이터를 JSON 형식으로 저장
- 질문 페이지에는 여러 개의 답변이 달릴 수 있기 때문에, 질문자가 직접 채택한 답변이 있으면, 답변을 우선적으로 수집

  → 질문자가 답변을 채택하지 않았을 때는, Stack Overflow에서 사용자들이 투표하여 가장 많은 득표(추천)를 얻은 답변을 선정

  → 답변이 없는 경우 제외

Data Crawling (stackoverflow)

Crawling 방법

stackoverflow.py

```python
import time
import json
import cloudscraper
from bs4 import BeautifulSoup
import re

# Configuration
BASE_URL    = 'https://stackoverflow.com'
TAG         = 'ns-3'
DELAY       = 1                 # Delay between requests (seconds)
MAX_PAGES   = 30                # Number of pages to crawl (15 posts per page)
OUTPUT_FILE = 'ns3_qa.json'

scraper = cloudscraper.create_scraper()
HEADERS = {
    'accept': 'text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp
    'accept-language': 'ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7',
    'cache-control': 'no-cache',
    'pragma': 'no-cache',
    'sec-ch-ua': '"Chromium";v="136", "Google Chrome";v="136", "Not.A/Brand";v="99"',
    'sec-ch-ua-mobile': '?0',
    'sec-ch-ua-platform': '"macOS"',
    'sec-fetch-dest': 'document',
    'sec-fetch-mode': 'navigate',
    'sec-fetch-site': 'none',
    'sec-fetch-user': '?1',
    'upgrade-insecure-requests': '1',
    'user-agent': 'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHT
    'cookie': 'prov=17412cb4-95bb-42a0-b540-b9eb92e3817f; __cflb=02DiuFA7zZL3enAQJD3AX8Zzv
}

# Helpers
def slugify(title: str) -> str:
    """Convert title to URL slug."""
    slug = title.lower()
    slug = re.sub(r"[^a-z0-9\s-]", "", slug)
    slug = re.sub(r"[\s-]+", "-", slug).strip('-')
    return slug


def get_question_links(page: int) -> list:
    """Fetch question URLs from the tagged listing page."""
    url = f"{BASE_URL}/questions/tagged/{TAG}?tab=newest&page={page}"
    resp = scraper.get(url, headers=HEADERS)
    resp.raise_for_status()
    soup = BeautifulSoup(resp.text, 'html.parser')

    links = []
    for summary in soup.select('div.s-post-summary'):
        a = summary.select_one('h3.s-post-summary--content-title a.s-link')
        if a and a.get('href'):
            links.append(BASE_URL + a['href'])
    return links
```

Data Crawling (stackoverflow)

Crawling 방법

stackoverflow.py

```python
def parse_question(url: str) -> dict:
    """Parse question page for title, body, and best answer."""
    resp = scraper.get(url, headers=HEADERS)
    resp.raise_for_status()
    soup = BeautifulSoup(resp.text, 'html.parser')

    # Title
    title_tag = soup.select_one('h1 a.question-hyperlink')
    title = title_tag.get_text(strip=True) if title_tag else ''

    # Question body
    q_body = soup.select_one('div.s-prose.js-post-body')
    question = q_body.get_text("\n", strip=True) if q_body else ''

    # Answer: accepted first, else highest-score
    answer = ''
    acc = soup.select_one('div.answer.js-accepted-answer')
    if acc:
        body = acc.select_one('div.s-prose.js-post-body')
        answer = body.get_text("\n", strip=True) if body else ''
    else:
        candidates = []
        for a_div in soup.select('div.answer'):
            body = a_div.select_one('div.s-prose.js-post-body')
            text = body.get_text("\n", strip=True) if body else ''
            score_tag = a_div.select_one('div.js-vote-count')
            try:
                score = int(score_tag.get_text(strip=True))
            except:
                score = 0
            candidates.append((score, text))
        if candidates:
            answer = max(candidates, key=lambda x: x[0])[1]

    return {'url': url, 'title': title, 'question': question, 'answer': answer}
```

```python
# Main
def main():
    results = []
    for page in range(1, MAX_PAGES + 1):
        print(f"[Page {page}] fetching listings...")
        try:
            links = get_question_links(page)
        except Exception as e:
            print(f"Failed to fetch page {page}: {e}")
            break
        if not links:
            break

        for link in links:
            print(f"  -> parsing {link}")
            try:
                qa = parse_question(link)
                # Skip unanswered questions
                if not qa['answer'].strip():
                    print(f"    skipped {link} (no answer)")
                    continue
                results.append(qa)
            except Exception as e:
                print(f"    parse error for {link}: {e}")
            time.sleep(DELAY)

    # Save to JSON
    with open(OUTPUT_FILE, 'w', encoding='utf-8') as f:
        json.dump(results, f, ensure_ascii=False, indent=2)
    print(f"Saved {len(results)} Q&A entries to {OUTPUT_FILE}")

if __name__ == '__main__':
    main()
```

Data Crawling (stackoverflow)

Crawling 결과

ns3_qa.json


```
[
  {
    "url": "https://stackoverflow.com/questions/79179369/de-aggregateobject-and-change-the-aggregrated-object-from-a-class-in-c-impleme",
    "title": "De-AggregateObject and Change the aggregrated object from a Class in c++ implemented in Ns3.38",
    "question": "I implemented Vehicle Ad-hoc Network in NS3.38 and used default Node class in NS3. also I created Cluster Object that has some
    "answer": "I think your goal in this question is clustering nodes and getting each cluster from node object. also each cluster has multi noc
  },
  {
    "url": "https://stackoverflow.com/questions/79146271/importing-opencv-and-tensorflow-lite-into-ns-3-38",
    "title": "Importing OpenCV and TensorFlow Lite into ns-3.38",
    "question": "I am currently working on a project that involves using\nOpenCV\nand\nTensorFlow Lite\nwithin the\nns-3.38\nsimulation environm
    "answer": "You're missing include headers in scratch/EDEN/node.cc.\n#include \"ns3/network-module.h\""
  },
  {
    "url": "https://stackoverflow.com/questions/79090343/missing-system-dependencies-for-ns3",
    "title": "missing system dependencies for ns3",
    "question": "I have been setting up ns-3-allinone on my WSL Python virtual environment using the in-built 'bake' tool. However during the se
    "answer": "Are you running bake from within your virtual environment?\nIf no, then try doing that.\nIf yes, then it could be a limitation of
  },
  {
    "url": "https://stackoverflow.com/questions/78645306/ns-3-multiple-wifi-clients-and-one-server-using-udp",
    "title": "NS-3 Multiple WiFi clients and one server using UDP",
    "question": "I am trying to built a ns-3 simulation with the following characteristics . I want multiple client nodes that will periodically
    "answer": "I don't know if its right but it seems to work I did it with UdpServer\nNS_LOG_INFO(\"Create UdpServer application on AP node.\")
  },
  {
    "url": "https://stackoverflow.com/questions/78485567/how-to-make-full-buffer-traffic-in-ns-3-lte-module",
    "title": "How to make full-buffer traffic in NS-3 LTE-module",
    "question": "I've already read documentation for LTE-module and EPC packets, but I don't understand how can I create full-buffer traffic. In
    "answer": "you can use                                                    Time interPacketInterval = MilliSeconds(1);  //1ms for
  },
```

감사합니다 :)