원고 세 편의 다음 행동을 저장하는 연재 준비실을 만들었습니다

지피터스 24기 · 2026-10-13 공개 실습 자료

파일 구성과 실행

아래 소스를 파일명대로 저장하세요. demo 폴더에서 python3 data.py, python3 server.py를 실행하고 브라우저로 http://127.0.0.1:8769 를 엽니다. 다음 행동을 수정·저장한 뒤 새로고침해 확인합니다.

브라우저 저장 후 재조회 기록

공개 실습 소스 / 긴 줄은 지면에 맞춰 자동 줄바꿈됩니다. 코드 복사 시 줄바꿈을 확인하세요.


```
{
  "before": "브라우저 계산 결과를 결과 원고에 반영",
  "saved": "최종 원고의 숫자와 첨부를 확인",
  "read_after_reload": true,
  "method": "visible browser input/save/reload",
  "screenshots": [
    "service-before.png",
    "service-after-reload.png"
  ]
}
```

demo/data.py

공개 실습 소스 / 긴 줄은 지면에 맞춰 자동 줄바꿈됩니다. 코드 복사 시 줄바꿈을 확인하세요.

```
import json,sqlite3
from pathlib import Path
ROOT=Path(__file__).resolve().parents[1]
def connect(path):
 c=sqlite3.connect(path);c.row_factory=sqlite3.Row;c.execute('PRAGMA foreign_keys=ON');return c
if __name__=='__main__':
 path=ROOT/'demo/serial.sqlite';c=connect(path);c.executescript((ROOT/'demo/schema.sql').read_text())
 source=ROOT/'demo/seed.json';rows=json.loads(source.read_text())
 for i,r in enumerate(rows,1):
  c.execute('INSERT OR IGNORE INTO episode VALUES(?,?,?)',(r['id'],r['title'],r['question']))
  for label,name in [('baseline',f'baseline-{i:02}.md'),('revised',f'revised-{i:02}.md')]:
   c.execute('INSERT OR IGNORE INTO revision VALUES(?,?,?,?)',(r['id']+'-
'+label,r['id'],label,'../knowledge/source/'+name))
  c.execute('INSERT OR IGNORE INTO action VALUES(?,?,?,?,?)',(r['id'],r['status'],r['next_action'],'',''))
 c.commit();c.close()
 c=connect(path)
 print(json.dumps({table:c.execute(f'SELECT COUNT(*) FROM {table}').fetchone()[0] for table in
['episode','revision','action']},ensure_ascii=False))
```

demo/schema.sql

공개 실습 소스 / 긴 줄은 지면에 맞춰 자동 줄바꿈됩니다. 코드 복사 시 줄바꿈을 확인하세요.

```
PRAGMA foreign_keys=ON;
CREATE TABLE IF NOT EXISTS episode(id TEXT PRIMARY KEY,title TEXT NOT NULL,question TEXT NOT NULL);
CREATE TABLE IF NOT EXISTS revision(id TEXT PRIMARY KEY,episode_id TEXT NOT NULL REFERENCES episode(id),label TEXT NOT NULL,source TEXT NOT NULL);
CREATE TABLE IF NOT EXISTS action(episode_id TEXT PRIMARY KEY REFERENCES episode(id),state TEXT NOT NULL CHECK(state IN ('draft','review','approved','published')),next_action TEXT NOT NULL,evidence TEXT NOT NULL DEFAULT '',published_url TEXT NOT NULL DEFAULT '',CHECK(state!='published' OR published_url LIKE 'https://%'));
```

demo/seed.json

공개 실습 소스 / 긴 줄은 지면에 맞춰 자동 줄바꿈됩니다. 코드 복사 시 줄바꿈을 확인하세요.


```
[
  {
    "id": "E01",
    "title": "AI 여행 일정은 실제로 움직일 수 있을까?",
    "question": "이동·예산·운영시간을 검증할 수 있는가",
    "status": "draft",
    "kind": "preview",
    "evidence": [],
    "published_url": "",
    "next_action": "출발지와 여행 날짜 확인",
    "deadline": null
  },
  {
    "id": "E02",
    "title": "AI 더치페이 도구를 믿어도 될까?",
    "question": "분할 금액 합계가 원래 총액과 같은가",
    "status": "review",
    "kind": "preview",
    "evidence": [],
    "published_url": "",
    "next_action": "기대값에 맞춰 도구 구현 후 실행",
    "deadline": null
  },
  {
    "id": "E03",
    "title": "AI 실습 안내만으로 페이지를 만들 수 있을까?",
    "question": "초보자가 안내만 보고 이름 변경을 완료하는가",
    "status": "draft",
    "kind": "preview",
    "evidence": [],
    "published_url": "",
    "next_action": "사용 운영체제와 시험 참가자 확인",
    "deadline": null
  }
]
```

demo/server.py

공개 실습 소스 / 긴 줄은 지면에 맞춰 자동 줄바꿈됩니다. 코드 복사 시 줄바꿈을 확인하세요.

```
"""Local study prototype. Serves only this folder on loopback; no external send."""
import json
from http.server import ThreadingHTTPServer,SimpleHTTPRequestHandler
from pathlib import Path
from data import connect,ROOT
class Handler(SimpleHTTPRequestHandler):
 def __init__(self,*a,**kw):super().__init__(*a,directory=str(ROOT/'demo'),**kw)
 def reply(self,status,data):
  body=json.dumps(data,ensure_ascii=False).encode();self.send_response(status);self.send_header('Content-
Type','application/json; charset=utf-8');self.end_headers();self.wfile.write(body)
 def do_GET(self):
  if self.path=='/api/episodes':
   with connect(ROOT/'demo/serial.sqlite') as c:rows=[dict(r) for r in c.execute('SELECT e.*,a.state,a.next_action FROM
episode e JOIN action a ON a.episode_id=e.id ORDER BY e.id')]
   return self.reply(200,rows)
  if self.path not in ['/','/index.html','/split.js']:return self.reply(404,{'error':'파일을 찾을 수 없습니다.'})
  return super().do_GET()
 def do_POST(self):
  if self.path!='/api/action':return self.reply(404,{'error':'없는 경로입니다.'})
  if self.headers.get('Origin') not in [None,'http://127.0.0.1:8769']:return self.reply(403,{'error':'허용되지 않은 요청입니
다.'})
  try:
   size=int(self.headers.get('Content-Length','0'))
   if size<1 or size>4096:raise ValueError()
   d=json.loads(self.rfile.read(size));ident=d['id'];action=d['next_action']
   if not isinstance(ident,str) or not isinstance(action,str) or not action.strip() or len(action)>300:raise ValueError()
   with connect(ROOT/'demo/serial.sqlite') as c:
    result=c.execute('UPDATE action SET next_action=? WHERE episode_id=?',(action.strip(),ident))
    if result.rowcount!=1:return self.reply(404,{'error':'회차가 없습니다.'})
   return self.reply(200,{'saved':True})
  except (ValueError,KeyError,TypeError):return self.reply(400,{'error':'다음 행동을 1~300자로 입력해주세요.'})
if __name__=='__main__':ThreadingHTTPServer(('127.0.0.1',8769),Handler).serve_forever()
```

demo/index.html

공개 실습 소스 / 긴 줄은 지면에 맞춰 자동 줄바꿈됩니다. 코드 복사 시 줄바꿈을 확인하세요.

```
<!doctype html><html lang="ko"><head><meta charset="utf-8"><meta name="viewport" content="width=device-width,initial-scale=1"><title>연재 준비실 · 실습</title><style>
*{box-sizing:border-box}body{margin:0;padding:32px;background:#edf1f5;color:#192b42;font:17px/1.7 system-ui,sans-serif}main{background:white;border-radius:16px;padding:36px;max-width:960px;margin:auto}h1{font-size:32px;line-height:1.4;margin:0 0 16px}h2{font-size:24px;margin:40px 0 16px}h3{word-break:keep-all;font-size:19px;margin:0 0 8px}p{max-width:70ch}label{display:block;font-weight:600;margin:16px 0 6px}input{display:block;width:100%;max-width:600px;border:1px solid #52647a;border-radius:6px;padding:12px;font:inherit;color:inherit;background:white}button{background:#192b42;color:white;border:0;border-radius:6px;padding:12px 20px;font:inherit;margin-top:16px;cursor:pointer}button:hover{background:#334f70}button:disabled{opacity:.6;cursor:wait}input:focus-visible,button:focus-visible{outline:3px solid #b34c00;outline-offset:3px}::selection{background:#d3e3f4}input{caret-color:#192b42}article{border-top:1px solid #cdd8e4;padding:24px 0}small,.note{color:#52647a}.error{color:#a52622}output{display:block;margin-top:16px;font-variant-numeric:tabular-nums}output ol{max-height:260px;overflow:auto}footer{margin-top:36px;color:#52647a}a{color:#245484;text-underline-offset:4px}@media(max-width:600px){body{padding:12px}main{padding:24px 18px}h1{font-size:27px}button{width:100%}}
</style></head><body><main><h1>연재 준비실</h1><p>세 편의 원고에서 다음 행동을 이어갑니다. 아래 데이터는 공개 연재 실습용입니다. 저장한 다음 행동은 이 컴퓨터의 DB에 남습니다.</p><section aria-labelledby="episodes-title"><h2 id="episodes-title">이어서 할 일</h2><div id="episodes" aria-live="polite">원고를 불러오는 중입니다.</div></section><section aria-labelledby="split-title"><h2 id="split-title">2화 실험 · 원 단위 더치페이</h2><p>나머지 원은 앞 번호부터 1원씩 배분합니다. 실제 송금은 하지 않습니다.</p><form id="split-form" novalidate><label for="total">총액(원)</label><input id="total" inputmode="numeric" value="10000" aria-describedby="limit"><label for="people">인원 (명)</label><input id="people" inputmode="numeric" value="3"><p id="limit" class="note">총액 0~10억 원, 인원 1~1,000명. 정수만 입력할 수 있습니다.</p><button type="submit">금액 나누기</button><output id="split-result" aria-live="polite"></output></form></section><footer>지피터스 24기 실습 · 로컬 시제품 · 외부 사용자 검증 전</footer></main><script src="split.js"></script><script>
const box=document.getElementById('episodes');
async function load(){try{const res=await fetch('/api/episodes');if(!res.ok)throw Error();const rows=await res.json();box.replaceChildren();if(!rows.length){box.textContent='등록된 원고가 없습니다.';return;}for(const r of rows){const article=document.createElement('article'),title=document.createElement('h3'),question=document.createElement('p'),label=document.createElement('label'),input=document.createElement('input'),button=document.createElement('button'),status=document.createElement('p');title.textContent=r.id+' · '+r.title;question.textContent=r.question;label.textContent='다음 행동';input.id='action-'+r.id;label.htmlFor=input.id;input.value=r.next_action;input.maxLength=300;button.textContent='다음 행동 저장';button.type='button';status.setAttribute('role','status');input.addEventListener('input',()=>{status.className='note';status.textContent='수정됨 · 저장 필요';});button.onclick=async()=>{button.disabled=true;try{const resp=await fetch('/api/action',{method:'POST',headers:{'Content-Type':'application/json'},body:JSON.stringify({id:r.id,next_action:input.value})});const result=await resp.json();status.className=resp.ok?'':'error';status.textContent=resp.ok?'저장했습니다. 다시 열어도 유지됩니다.':result.error;}catch(e){status.className='error';status.textContent='저장하지 못했습니다. 로컬 서버가 실행 중인지 확인해주세요.';}finally{button.disabled=false;}};article.append(title,question,label,input,button,status);box.append(article);}}catch(e){box.textContent='원고를 불러오지 못했습니다. 로컬 서버를 실행한 뒤 새로고침해주세요.';}}
load();const form=document.getElementById('split-form'),out=document.getElementById('split-result');form.addEventListener('input',()=>out.replaceChildren());form.onsubmit=e=>{e.preventDefault();out.replaceChildren();out.className='';try{const amounts=splitAmount(document.getElementById('total').value,document.getElementById('people').value);const p=document.createElement('p');p.textContent='합계 '+amounts.reduce((a,b)=>a+b,0).toLocaleString('ko-KR')+'원 · '+amounts.length+'명';const ol=document.createElement('ol');for(const a of amounts){const
```

```
li=document.createElement('li');li.textContent=a.toLocaleString('ko-KR')+'원';ol.append(li);}out.append(p,ol);}catch(err)
{out.className='error';out.textContent=err.message;}};
</script></body></html>
```

demo/split.js

공개 실습 소스 / 긴 줄은 지면에 맞춰 자동 줄바꿈됩니다. 코드 복사 시 줄바꿈을 확인하세요.

```
(function(root){
function split(total, people){
 if(!/^\d+$/.test(String(total).trim()) || !/^\d+$/.test(String(people).trim())) throw Error('총액과 인원은 정수로 입력해주세
요.');
 const a=Number(total), n=Number(people);
 if(!Number.isSafeInteger(a)||a<0||a>1000000000)throw Error('총액은 0~1,000,000,000원으로 입력해주세요.');
 if(!Number.isSafeInteger(n)||n<1||n>1000)throw Error('인원은 1~1,000명으로 입력해주세요.');
 const q=Math.floor(a/n), r=a%n;
 return Array.from({length:n},(_,i)=>q+(i<r?1:0));
}
if(typeof module!=='undefined')module.exports=split;else root.splitAmount=split;
})(globalThis);
```