

FastAPI 실습 문제집

준비사항

- FastAPI 설치: `pip install fastapi uvicorn`
- Thunder Client 확장 프로그램 설치
- 서버 실행: `uvicorn main:app --reload`

Pydantic 모델 작성 팁

- `from pydantic import BaseModel`로 импорт
- 클래스로 정의하고 BaseModel 상속
- 타입 힌트 사용 (str, int, float, bool 등)
- 기본값 설정: `field_name: str = "기본값"`
- 선택적 필드: `Optional[타입]` 사용 시 `from typing import Optional` 필요

문제 1 (하급) - 사용자 정보 조회

주소 매개변수 방식

요구사항

- GET 메서드로 `/user/{user_id}` 엔드포인트 작성
- `user_id`는 정수형 매개변수
- 응답으로 사용자 ID와 이름을 반환
- 사용자 이름은 "사용자{user_id}" 형태로 생성

테스트 예시

GET http://localhost:8000/user/123

문제 2 (하급) - 상품 등록

본문 요청 방식

요구사항

- POST 메서드로 `/product` 엔드포인트 작성
- 상품 정보를 받는 Pydantic 모델 생성
 - 상품명 (필수, 문자열)

- 가격 (필수, 정수)
- 설명 (선택, 문자열, 기본값: "설명 없음")
- 등록 성공 메시지와 받은 상품 정보 반환

테스트 예시

json

POST http://localhost:8000/product
Content-Type: application/json

```
{  
  "name": "노트북",  
  "price": 1000000  
}
```

문제 3 (하급) - 게시글 상세 조회

주소 매개변수 방식

요구사항

- GET 메서드로 `/post/{post_id}/category/{category}` 엔드포인트 작성
- `post_id`는 정수형, `category`는 문자열 매개변수
- 응답으로 게시글 ID, 카테고리, 제목 반환
- 제목은 "[category] 게시글 {post_id}" 형태로 생성

테스트 예시

GET http://localhost:8000/post/5/category/notice

문제 4 (중급) - 회원 가입

본문 요청 방식

요구사항

- POST 메서드로 `/register` 엔드포인트 작성
- 회원 정보를 받는 Pydantic 모델 생성
 - 이메일 (필수, 문자열)
 - 비밀번호 (필수, 문자열)

- 이름 (필수, 문자열)
- 나이 (선택, 정수, 기본값: 0)
- 활성화 상태 (선택, 불린, 기본값: True)
- 이메일에 "@"가 포함되어 있으면 가입 성공, 없으면 "잘못된 이메일" 에러 반환
- 성공 시 회원 정보와 "가입 완료" 메시지 반환

테스트 예시

```
json

POST http://localhost:8000/register
Content-Type: application/json

{
  "email": "test@example.com",
  "password": "123456",
  "name": "홍길동"
}
```

문제 5 (중급) - 주문 상태 업데이트

주소 매개변수 + 본문 요청 방식

요구사항

- PUT 메서드로 `/order/{order_id}/status` 엔드포인트 작성
- `order_id`는 정수형 매개변수
- 주문 상태 정보를 받는 Pydantic 모델 생성
 - 상태 (필수, 문자열)
 - 메모 (선택, 문자열, 기본값: "")
- 상태가 "complete", "pending", "cancelled" 중 하나가 아니면 "잘못된 상태" 에러 반환
- 성공 시 주문 ID, 새로운 상태, 메모 반환

테스트 예시

```
json
```

```
PUT http://localhost:8000/order/100/status
```

```
Content-Type: application/json
```

```
{  
  "status": "complete",  
  "memo": "배송 완료"  
}
```

문제 6 (상급) - 사용자별 리뷰 작성

주소 매개변수 + 본문 요청 방식

요구사항

- POST 메서드로 `/user/{user_id}/review/{product_id}` 엔드포인트 작성
- `user_id`, `product_id`는 정수형 매개변수
- 리뷰 정보를 받는 Pydantic 모델 생성
 - 평점 (필수, 정수)
 - 내용 (필수, 문자열)
 - 추천 여부 (선택, 불린, 기본값: False)
- 평점이 1~5 범위를 벗어나면 "평점은 1-5 사이여야 합니다" 에러 반환
- 내용이 10자 미만이면 "리뷰는 10자 이상 작성해주세요" 에러 반환
- 성공 시 사용자 ID, 상품 ID, 리뷰 정보 모두 반환

테스트 예시

json

```
POST http://localhost:8000/user/1/review/50
```

```
Content-Type: application/json
```

```
{  
  "rating": 5,  
  "content": "정말 좋은 상품이었습니다. 다음에도 구매하고 싶어요!",  
  "recommend": true  
}
```

Thunder Client 사용법

1. VS Code에서 Thunder Client 확장 프로그램 실행

2. "New Request" 버튼 클릭
3. HTTP 메서드 선택 (GET, POST, PUT)
4. URL 입력
5. POST/PUT의 경우 "Body" 탭에서 JSON 데이터 입력
6. "Send" 버튼으로 요청 전송