

재.개정 이력

No	버전	작성일	변경내용	작성자
1	V1.0	2025년 8월 29일 금	신규 작성	장현호

0. 공통 규약

0.1. 공통 예외 처리

표준 에러 (RFC 7807 / Spring ProblemDetail)

```
{
  "type": "https://api.example.com/errors/validation",
  "title": "Validation error",
  "status": 400,
  "detail": "Field 'amount' must be >= 100",
  "instance": "/api/v1/wallet/topups",
  "errors": [
    { "field": "expectedTotal", "code": "Min", "message": "최소 100 이상이어야 합니다." }
  ]
}
```

0.1.1. 왜 ProblemDetail 을 쓰는가?

- Spring Boot 3.0부터 ProblemDetail 클래스가 JDK 17 + Jakarta EE 스펙에 맞춰 기본 제공됨
- RFC 7807: Problem Details for HTTP APIs를 구현한 표준 → FE/BE 간 인터페이스가 일정해짐
- 응답 필드(type, title, status, detail, instance)가 정형화돼서 프론트에서 파싱하기 편리
- properties 에 원하는 추가 필드(field, code, errors)를 붙일 수 있어 확장성도 충분

0.1.2. 공통 에러 카탈로그 정의

- type 값은 서비스 도메인별 고정 URI 형태
 - /errors/validation
 - /errors/idempotency-conflict
 - /errors/out-of-stock
 - /errors/insufficient-funds
- FE에서 type 값으로 바로 분기 가능 → UX 구현이 쉬워짐
- 필드 요약 정리
 - type + status = FE 분기/로직 처리 기준
 - title + detail = 사용자/로그 메시지
 - instance = 디버깅/트레이싱
 - properties = 서비스 맞춤형 확장

0.1.3. 응답 예시

검증 오류 (400)

```
{
  "type": "/errors/validation",
  "title": "Validation error",
  "status": 400,
  "detail": "입력값이 올바르지 않습니다.",
  "instance": "/api/v1/orders",
  "errors": [
    { "field": "expectedTotal", "code": "Min", "message": "최소 100 이상이어야 합니다." }
  ]
}
```

```
]
}
```

재고 부족(422)

```
{
  "type": "/errors/out-of-stock",
  "title": "Out of stock",
  "status": 422,
  "detail": "상품 2f1f7c3b-... 재고가 부족합니다.",
  "instance": "/api/v1/orders"
}
```

역등 충돌(409)

```
{
  "type": "/errors/idempotency-conflict",
  "title": "Idempotency conflict",
  "status": 409,
  "detail": "동일 키로 이미 다른 요청이 처리되었습니다.",
  "instance": "/api/v1/wallet/topups"
}
```

0.2. 멱등성(Idempotency) 처리 (보류)

클라이언트가 동일한 요청을 여러 번 보내더라도 서버의 최종 상태는 변하지 않아야 한다는 성질.

- 잔액 충전, 주문/결제, 선착순 쿠폰 발급은 “같은 요청이 여러 번 들어와도 결과는 한 번만 반영”

0.3. 커서 기반 페이지네이션(cursor-based pagination)

커서: DB의 특정 위치를 가리키는 포인터 같은 값

- 일반적인 `?page=1&size=20` 같은 오프셋 기반 페이지네이션 대신, “마지막으로 본 데이터의 위치”를 저장해두고, 그 이후 데이터를 가져오는 방식

0.3.1 동작 방식 예시

#1. 상품 목록 API

```
GET /api/v1/products?limit=20
```

#2. 첫 응답

```
{
  "data": {
    "items": [
      { "id": "prod_101", "name": "..." },
      { "id": "prod_102", "name": "..." } ✖
    ]
  },
  "meta": {
    "nextCursor": "prod_102"
  }
}
```

#3. 다음 호출

```
GET /api/v1/products?limit=20&cursor=prod_102
```

#3.1. SQL

```
WHERE id > 'prod_102' ORDER BY id LIMIT 20
```

1. Wallet (지갑)

1.1. GET /api/v1/wallet - 내 지갑/거래 목록

Query: limit (1~100, 기본 20), cursor(선택)

🔍 200 응답

```
{
  "data": {
    "balance": { "amount": 82000, "currency": "KRW" },
    "transactions": {
      "items": [
        {
          "id": "b6d2b6d4-7e9d-4b7f-8c1e-6f1f4c0a9a11",
          "type": "TOPUP",
          "amount": { "amount": 30000, "currency": "KRW" },
          "balanceAfter": { "amount": 112000, "currency": "KRW" },
          "refType": "TOPUP",
          "refId": "tp_20250829_0012",
          "createdAt": "2025-08-29T02:12:44Z"
        }
      ],
      "nextCursor": "eyJvZmZzZXQiOjF9"
    }
  }
}
```

1.2. POST /api/v1/wallet/topups - 지갑 충전 생성

Headers: Idempotency-Key 필수 ⚠(보류-먹등성 처리)

🔊 요청

```
{
  "amount": 30000,
  "method": "CARD",
  "provider": "TOSSPAYMENTS",
  "metadata": { "memo": "9월 구독" }
}
```

🔍 201 응답

```
{
  "data": {
    "id": "b6d2b6d4-7e9d-4b7f-8c1e-6f1f4c0a9a11",
    "type": "TOPUP",
    "amount": { "amount": 30000, "currency": "KRW" },
    "balanceAfter": { "amount": 112000, "currency": "KRW" },
    "refType": "TOPUP",
    "refId": "tp_20250829_0012",
    "createdAt": "2025-08-29T02:12:44Z"
  }
}
```

❌ 409 (먹등 충돌)

```
{
  "type": "https://api.example.com/errors/idempotency-conflict",
  "title": "Idempotency conflict",
  "status": 409,
  "detail": "Existing request with the same Idempotency-Key has different body."
}
```

2. Products (상품/카탈로그)

2.1. GET /api/v1/products - 상품 목록

Query: q, categoryId, sort, limit, cursor

🔍 200 응답

```
{
  "data": {
    "items": [
      {

```

```
    "id": "2f1f7c3b-0b2b-4af3-9c6d-37d4fef93ca1",
    "sku": "SKU-1001",
    "name": "블루투스 이어폰",
    "price": { "amount": 59000, "currency": "KRW" },
    "stock": 27,
    "popularityScore": 0.82,
    "status": "ACTIVE",
    "thumbnailUrl": "https://cdn/img/1001.png"
  }
],
},
"meta": { "nextCursor": "eyJvZmZzZXQiOiI0fQ" }
}
```

2.2. GET /api/v1/products/{productId} - 상품 상세

☑ 200 응답

```
{
  "data": {
    "id": "2f1f7c3b-0b2b-4af3-9c6d-37d4fef93ca1",
    "sku": "SKU-1001",
    "name": "블루투스 이어폰",
    "price": { "amount": 59000, "currency": "KRW" },
    "stock": 27,
    "status": "ACTIVE",
    "thumbnailUrl": "https://cdn/img/1001.png"
  }
}
```

2.3. GET /api/v1/products/popular?window3&limit=5 - 최근 N일 인기 Top-N

☑ 200 응답

```
{
  "data": {
    "items": [
      {
        "id": "2f1f7c3b-0b2b-4af3-9c6d-37d4fef93ca1",
        "name": "블루투스 이어폰",
        "price": { "amount": 59000, "currency": "KRW" }
      }
    ]
  }
}
```

3. Coupons (쿠폰)

3.1. GET /api/v1/coupons?onlyUsable=true - 보유 쿠폰 목록

☑ 200 응답

```
{
  "data": {
    "items": [
      {
        "id": "c36b1e5c-1fa8-4c26-9a2a-7153f7c8ba01",
        "code": "WELCOME10",
        "type": "PERCENT",
        "value": 10.0,
        "minAmount": 30000,
        "maxDiscount": 10000,
        "startsAt": "2025-08-01T00:00:00Z",
        "endsAt": "2025-09-30T23:59:59Z"
      }
    ]
  }
}
```

3.2. POST /api/v1/coupons/validate - 쿠폰 유효성 검증

🔊 요청

```
{ "code": "WELCOME10", "orderAmount": 59000 }
```

📄 200 응답

```
{
  "data": {
    "code": "WELCOME10",
    "isValid": true,
    "discountAmount": { "amount": 5900, "currency": "KRW" },
    "reason": null
  }
}
```

4. Orders & Payments (주문/결제)

4.1. POST /api/v1/orders - 주문 생성(재고 예약/쿠폰/총액 계산)

Headers: Idempotency-Key 필수 (△보류-역등성 처리)
규칙: 서버 계산 total ≠ expectedTotal → 409 Conflict

🔊 요청

```
{
  "items": [
    { "productId": "2f1f7c3b-0b2b-4af3-9c6d-37d4fef93ca1", "qty": 1 },
    { "productId": "8f2d1b4a-4f7c-42b9-9a47-1fb8c0a2b211", "qty": 2 }
  ],
  "couponCode": "WELCOME10",
  "expectedTotal": 152100
}
```

📄 201 응답

```
{
  "data": {
    "id": "ord_7f0a6a9e-f4f7-4baf-8e9b-923c9c7e8c1d",
    "orderNo": "20250829-000123",
    "status": "RESERVED",
    "items": [
      {
        "productId": "2f1f7c3b-0b2b-4af3-9c6d-37d4fef93ca1",
        "name": "블루투스 이어폰",
        "qty": 1,
        "unitPrice": { "amount": 59000, "currency": "KRW" },
        "lineTotal": { "amount": 59000, "currency": "KRW" }
      }
    ],
    "discount": { "amount": 5900, "currency": "KRW" },
    "total": { "amount": 152100, "currency": "KRW" },
    "paymentMethod": null,
    "createdAt": "2025-08-29T02:30:00Z",
    "updatedAt": "2025-08-29T02:30:00Z"
  }
}
```

❌ 409 (합계 불일치)

```
{
  "type": "https://api.example.com/errors/total-mismatch",
  "title": "Order total mismatch",
  "status": 409,
  "detail": "Client expectedTotal does not match server calculation."
}
```

❌ 422 (재고 부족)

```
{
  "type": "https://api.example.com/errors/out-of-stock",
  "title": "Out of stock",
  "status": 422,
  "detail": "Product 2f1f7c3b-... insufficient inventory."
}
```

4.2. GET /api/v1/orders/{orderId} - 주문 상세

📄 200 응답

```
{
  "data": {
    "id": "ord_7f0a6a9e-f4f7-4baf-8e9b-923c9c7e8c1d",
    "orderNo": "20250829-000123",
    "status": "RESERVED",
    "items": [
      {
        "productId": "2f1f7c3b-0b2b-4af3-9c6d-37d4fef93ca1",
        "name": "블루투스 이어폰",
        "qty": 1,
        "unitPrice": { "amount": 59000, "currency": "KRW" },
        "lineTotal": { "amount": 59000, "currency": "KRW" }
      }
    ],
    "discount": { "amount": 5900, "currency": "KRW" },
    "total": { "amount": 152100, "currency": "KRW" },
    "paymentMethod": null,
    "createdAt": "2025-08-29T02:30:00Z",
    "updatedAt": "2025-08-29T02:30:00Z"
  }
}
```

4.3. POST /api/v1/orders/{orderId}/pay - 결제 확정 (재고 확정 + 결제 승인)

Headers: Idempotency-Key 필수 ⚠(보류-역등성 처리)

🔊 요청

```
{ "method": "WALLET", "provider": null, "metadata": { "device": "web" } }
```

📄 200 응답

```
{
  "data": {
    "id": "ord_7f0a6a9e-f4f7-4baf-8e9b-923c9c7e8c1d",
    "orderNo": "20250829-000123",
    "status": "PAID",
    "items": [ /* 생략 */ ],
    "discount": { "amount": 5900, "currency": "KRW" },
    "total": { "amount": 152100, "currency": "KRW" },
    "paymentMethod": "WALLET",
    "createdAt": "2025-08-29T02:30:00Z",
    "updatedAt": "2025-08-29T02:31:12Z"
  }
}
```

❌ (지갑 잔액 부족)

```
{
  "type": "https://api.example.com/errors/insufficient-funds",
  "title": "Insufficient wallet balance",
  "status": 422,
  "detail": "Required 152100 KRW, available 82000 KRW."
}
```

5. Health

5.1. GET /api/v1/health

☑ 200 응답

```
{ "data": { "status": "ok", "uptimeSec": 12345 } }
```