

Introduction

카카오페이에는 멤버십 적립 서비스가 있습니다.

- 사용자는 원하는 멤버십을 등록할 수 있습니다.
(멤버십은 해피포인트, 신세계포인트, CJ ONE 3가지만 존재합니다.)
- 포인트 적립비율은 결제금액의 1%로 고정됩니다.
- 멤버십 사용에 대한 기능은 만들지 않습니다.

이번 과제에서는 UI를 제외한 RETS API를 구현하는 것이 목표입니다.

위 목표와 아래의 요구사항을 만족하는 API 서비스를 자유롭게 정의하여 구현해주시면 됩니다.

기능 요구 사항

멤버십 연결하기, 나의 멤버십 조회, 멤버십 연결끊기, 포인트 적립 API 를 구현합니다.

요청한 사용자 식별값은 문자열 형태이며 "X-USER-ID" 라는 HTTP Header 로 전달됩니다. 이 값은 포인트 적립할때 바코드 대신 사용됩니다.
(SpringSecurity는 사용하지 않습니다.)

Content-type 응답 형태는 application/json(JSON) 형식을 사용합니다.

각 기능 및 제약사항에 대한 단위테스트를 반드시 작성합니다.

요구사항에 따른 상세 기술 구현 사항

1 멤버십 전체 조회

다음의 요건을 만족하는 나의 멤버십 조회 API 를 작성해주세요.

- 모든 멤버십을 조회합니다.
- 사용자 식별값을 입력값으로 받습니다.
- 나의 멤버십 조회 응답은 다음 내용을 포함합니다.
 - 멤버십 ID, 멤버십 이름, 포인트, 멤버십상태(활성, 비활성), 가입 일시

2 멤버십 등록하기

다음의 요건을 만족하는 멤버십 등록하기 API 작성해주세요.

- 사용자 식별값, 멤버십 ID, 멤버십 이름, 포인트를 입력값으로 받습니다.
- 나의 멤버십 응답은 다음 내용을 포함합니다.
 - 멤버십 ID, 멤버십 이름

3 멤버십 삭제(비활성화)하기

다음의 요건을 만족하는 멤버십 삭제(비활성화)하기 API 를 작성해주세요.

- 멤버십을 비활성화 상태로 변경합니다.
- 사용자 식별값, 멤버십 ID를 입력값으로 받습니다.

4 멤버십 상세조회

다음의 요건을 만족하는 멤버십 상세 조회 API 를 작성해주세요.

- 사용자 식별값, 멤버십 ID를 입력값으로 받습니다.
- 나의 멤버십 응답은 다음 내용을 포함합니다.
 - 멤버십 ID, 멤버십 이름, 사용자 식별값, 포인트, 가입일시,멤버십 상태(활성, 비활성)을 응답합니다.

5 포인트 적립

다음의 요건을 만족하는 멤버십 포인트 적립 API 를 작성해주세요.

- 사용자 식별값, 멤버십 ID, 사용 금액을 입력값으로 받습니다.
- 포인트 적립은 결제 금액의 1%가 적립됩니다.

기술 요구 사항

- 개발 언어 Java 8 or Kotlin
- Framework - Spring Boot
- Persistence Framework - JPA
- 데이터베이스 mysql / redis / H2 활용 가능
 - mysql (host: mysql-server, port: 3306, user: root, password: password)
 - redis (host: redis-server, port: 6379)
- 에러응답, 에러코드는 자유롭게 정의해주세요.
- 개발 편의를 위해 2가지 라이브러리를 사용할수 있습니다.
 - 1. commons.lang3
 - 2. guava

평가항목

과제는 다음 내용을 고려하여 평가하게 됩니다.

- 프로젝트 구성 방법 및 관련된 시스템 아키텍처 설계 방법이 적절한가?
- 요구사항을 잘 이해하고 구현하였는가?
- RESTAPI 의 응답 포맷이 일정한가?
- DB 테이블 설계 및 쿼리는 효율적으로 작성 되었는가?
- 예외처리를 적절하게 하였는가?(예외 타입별로 HTTP STATUS 코드의 처리가 적절한지)
- 작성한 테스트 코드는 적절한 범위의 테스트를 수행하고 있는가? (예. 유닛/통합 테스트 등)

API 명세서

요청공통

헤더

항목	값(예)	설명
Content-Type	application/json	JSON 형태 사용
X-USER-ID	test1	사용자 식별값

응답 공통

HTTP 응답코드

응답코드	설명
200	성공
400	잘못된 요청
404	찾을 수 없음
500	내부 서버 오류

헤더

항목	값	설명
Content-Type	application/json	JSON 형식 응답

내용

이름	타입	필수	설명
success	boolean	O	성공 여부
response	object	O	성공 응답 내용
error	object	O	에러 응답 내용

응답 예

정상처리 및 오류처리 모두 success 필드를 포함합니다.

- 정상처리라면 true, 오류처리라면 false 값을 출력합니다. 정상처리는 response 필드를 포함하고 error 필드는 null 입니다. 오류처리는 error 필드를 포함하고 response 필드는 null 입니다. error 필드는 status, message 필드를 포함합니다.
- status : HTTP Response status code 값과 동일한 값을 출력해야 합니다.
- message : 오류 메시지가 출력 됩니다.

정상 처리

```
{
  "success":true,
  "response":[
    {
      "seq":1,
      "membershipId":"spc",
      "userId":"test1",
      "membershipName":"happypoint",
      "startDate":"2021-06-20T14:48:29.831",
      "membershipStatus":"Y",
      "point":120
    },
    {
      "seq":2,
      "membershipId":"shinsegae",
      "userId":"test1",
      "membershipName":"shinsegaepoint",
      "startDate":"2021-06-20T14:48:30.011",
      "membershipStatus":"Y",
      "point":3500
    },
    {
      "seq":3,
      "membershipId":"cj",
      "userId":"test1",
      "membershipName":"cjone",
      "startDate":"2021-06-20T14:48:30.043",
      "membershipStatus":"N",
      "point":1029
    }
  ],
  "error":null
}

오류 처리
{
  "success":false,
  "response":null,
  "error":{
    "message":"membershipId must be provided",
    "status":400
  }
}
```

1. 멤버십 전체조회하기 API

요청

항목	값
URL	GET /api/v1/membership

응답

응답 내용

이름	타입	필수	설명
response	object	O	멤버십 전체조회하기에 대한 값

이름	타입	필수	설명
membership	List (Object)	O	멤버십 정보

응답 예시

```
{
  "success":true,
  "response":[
    {
      "seq":1,
      "membershipId":"spc",
      "userId":"test1",
      "membershipName":"happypoint",
      "startDate":"2021-06-20T14:48:29.831",
      "membershipStatus":"Y",
      "point":120
    },
    {
      "seq":2,
      "membershipId":"shinsegae",
      "userId":"test1",
      "membershipName":"shinsegaepoint",
      "startDate":"2021-06-20T14:48:30.011",
      "membershipStatus":"Y",
      "point":3500
    },
    {
      "seq":3,
      "membershipId":"cj",
      "userId":"test1",
      "membershipName":"cjone",
      "startDate":"2021-06-20T14:48:30.043",
      "membershipStatus":"N",
      "point":1029
    }
  ],
  "error":null
}
```

2. 멤버십 등록하기 API

요청

항목	값
URL	POST /api/v1/membership

요청 항목

이름	타입	필수	설명
membershipId	string	O	멤버십 ID
membershipName	string	O	멤버십 이름
point	int	O	포인트

요청 예시

```
{
  "membershipId": "cj",
  "membershipName": "cjone",
  "point": 5210
}
```

응답

응답 내용

이름	타입	필수	설명
response	object	O	멤버십 전체조회하기에 대한 값

응답 예시

```
{
  "success": true,
  "response": {
    "seq": 3,
    "membershipId": "cj",
    "userId": "test1",
    "membershipName": "cjone",
    "startDate": "2021-07-01T16:33:21.291",
    "membershipStatus": "Y",
    "point": 5210
  },
  "error": null
}
```

3. 멤버십 삭제(비활성화)하기 API

요청

항목	값	설명
URL	DELETE /api/v1/membership/{membershipId}	{membershipId} = 멤버십 ID

요청 항목

이름	타입	필수	설명
membershipId	string	O	멤버십 ID

응답

응답 내용

이름	타입	필수	설명
----	----	----	----

이름	타입	필수	설명
response	boolean	O	성공여부

응답 예시

```
{
  "success":true,
  "response":true,
  "error":null
}
```

4. 멤버십 상세 조회하기 API

요청

항목	값	설명
URL	GET /api/v1/membership/{membershipId}	{membershipId} = 멤버십 ID

응답

응답 내용

이름	타입	필수	설명
response	object	O	멤버십 상세조회하기에 대한 값
membership	object	O	멤버십 정보

응답 예시

```
{
  "success":true,
  "response":{
    "seq":1,
    "membershipId":"spc",
    "membershipName":"happypoint",
    "userId":"test1",
    "point":120,
    "startDate":"2021-07-01T16:33:21.291",
    "membershipStatus":"Y"
  },
  "error":null
}
```

5. 포인트 적립하기 API

요청

항목	값
----	---

항목	값
URL	PUT /api/v1/membership/point

요청 항목

이름	타입	필수	설명
membershipId	string	O	멤버십 ID
amount	int	O	사용금액

요청 예시

```
{
  "membershipId": "cj",
  "amount": 100
}
```

응답

응답 내용

이름	타입	필수	설명
response	boolean	O	성공여부

응답 예시

```
{
  "success": true,
  "response": true,
  "error": null
}
```