

# 회원

---

## 회원가입

- ACCOUNT\_ACCOUNT\_STATE 에 가입 및 활성이 들어가야 한다.

## 로그인

- ID 조회, PW 매치...
- 로그인 성공시 LAST\_LOGIN\_AT 시간 변경 할 수 있는 API 필요

## 회원관리

- ACCOUNT\_ACCOUNT\_STATE CRUD

## ID 조회

- 프로젝트 쪽에서 무결성 확인을 위해 호출되는 API

## 주의사항

- 회원 삭제 (탈퇴아님) 시 프로젝트 쪽 해당 데이터 전부 삭제 시키는 것 고려

ACCOUNT, ACCOUNT\_STATE, ACCOUNT\_ACCOUNT\_STATE

# 메시지

---

- json 포맷을 사용합니다.

## 요청 메시지

- json body를 포함하여 보내야 하는 메시지의 경우, Content-Type 헤더를 명시해야 합니다.
- 모든 요청에 Host를 명시합니다.
- 모든 요청에 Authorization 헤더를 포함해 요청합니다.

Content-Type : application/json

Host : localhost:3306

## 응답 메시지

# API Spec

---

## ACCOUNT-API > Account

### 회원 조회

**GET /account-api/accounts**

- 전체 계정 조회로 각 응답은 대략적인 계정 정보만 반환합니다.
- 멤버 목록을 응답

Request

- Parameters

항목명	항목명(영문)	항목크기(최소 값)	항목구 분	샘플데이터	비고
사용자 아이디	id	40(5)	필수	sampleId	회원 아이디로 검색
사용자 이메일	email	100(5)	옵션	sample@dooray.com	회원 이메일로 검색
페이지	page		옵션	0	페이지 번호, 목록 조회시 필수
한 페이지 결과 수	numOfRows		옵션	10	한 페이지 결과 수, 목록조 회사 필수

요청 예제

```
GET /account-api/accounts?page={page}&size={numOfRows}
```

Response

- Body

```
{
  "header": {
    "resultCode": 200,
    "resultMessage": "All account-account-states by account ID retrieved successfully",
    "successful": true
  },
  "result": [{
    "id": "id",
    "name": "name",
    "email": "email@email.com",
    "lastLoginAt": "2023-06-07T14:56:59",
    "createdAt": "2023-06-07T14:56:32"
  },
  {
    "id": "id",
    "name": "name",
    "email": "email@email.com",
    "lastLoginAt": "2023-06-07T14:56:59",
    "createdAt": "2023-06-07T14:56:32"
  }
}]
```

```
  ],  
  "totalCount": 2  
}
```

#### HTTP 응답 코드

- 404 NOT FOUND
  - NotFoundException

---

#### 회원 상세 조회

##### GET /account-api/accounts/{id}

- 개별 멤버를 응답
- 멤버 id로 접근하며, 디테일한 멤버 정보를 응답합니다.

#### Request

- Parameters

항목명	항목명(영문)	항목크기(최소값)	항목구분	샘플데이터	비고
사용자 아이디	id	40(5)	필수	sampleId	회원 아이디로 검색
사용자 이메일	email	100(5)	옵션	sample@dooray.com	회원 이메일로 검색

#### 요청 예제

```
GET /account-api/accounts/{id}  
  
GET /account-api/accounts/email/{email}  
  
GET /account-api/login/{id} // 회원 로그인 시간 갱신
```

#### Response

- Body

```
{  
  "header": {  
    "resultCode": 200,  
    "resultMessage": "All account-account-states by account ID retrieved  
successfully",  
    "successful": true  
  },  
}
```

```
"result": {
  "id": "id",
  "name": "name",
  "email": "email@email.com",
  "lastLoginAt": null,
  "createdAt": "2023-06-09T07:02:40"
},
"totalCount": 1
}
```

#### HTTP 응답 코드

- 200 OK
- 404 NOT FOUND
  - NotFoundException

---

## 회원 가입

### POST /account-api/accounts

- 사용자 정보를 추가합니다
- API 호출시 자동으로 가입 상태 변환

항목명	항목명(영문)	항목크기(최소값)	항목구분	샘플데이터	비고
사용자 아이디	id	40(5)	필수	sampleId	사용자 아이디
사용자 이름	name	20	필수	홍길동	사용자 이름
사용자 이메일	email	100(5)	필수	sample@dooray.com	사용자 이메일
사용자 패스워드	password		필수	1234	암호화

#### Request

- Body

```
{
  "id": "sampleId", /* 사용자의 아이디, 사용자 식별. */
  "name": "홍길동", /* 사용자 이름 */
  "email": "sample@dooray.com", /* 사용자 이메일, 사용자 식별 가능 */
  "password": "1234" /* 사용자 비밀번호. 암호화 알고리즘으로 변환되어 저장 */
}
```

#### Response

- Body

```
{
  "header": {
    "resultCode": 201,
    "resultMessage": "created successfully",
    "successful": true
  },
  "result": null,
  "totalCount": 0
}
```

## HTTP 응답코드

- 201 CREATED
- 
- 400 BAD REQUEST
  - InvalidIdFormatException
  - InvalidEmailFormatException
- 409 CONFLICT
  - DataAlreadyExistsException

---

## 회원 수정

- 이름
- 비밀번호
- 회원 로그인 시간

*PUT /account-api/accounts/update/{id}/name*

*PUT /account-api/accounts/update/{id}/password*

*PUT /account-api/login/{id}*

- 회원 상태 변경이 아닌, 회원 정보 변경만 처리

## 요청 예제

```
PUT /account-api/accounts/update/{id}/name
PUT /account-api/accounts/update/{id}/password
GET /account-api/login/{id}
```

## 회원정보 변경

### Request

- Parameter

항목명	항목명(영문)	항목크기(최소값)	항목구분	샘플데이터	비고
사용자 아이디	id	40(5)	필수	sampleId sampleId@dooray.com	사용자 이름 또는 이메일이 들어가는 항목

- Body - 이름

```
{
  "name": "{name}", /* 수정된 이름 */
}
```

- Body - 패스워드

```
{
  "password": "{password}" /* 수정된 비밀번호 */
}
```

## Response

- Body - 이름

```
{
  "header": {
    "resultCode": 200,
    "resultMessage": "updated successfully",
    "successful": true
  },
  "result": null,
  "totalCount": 0
}
```

- Body - 패스워드

```
{
  "header": {
    "resultCode": 200,
    "resultMessage": "updated successfully",
    "successful": true
  },
  "result": null,
  "totalCount": 0
}
```

- Body - 로그인 시간 갱신

```
{
  "header": {
    "resultCode": 200,
    "resultMessage": "updated successfully",
    "successful": true
  },
  "result": null,
  "totalCount": 0
}
```

#### HTTP 상태 코드

- 200 OK
- 400 BAD REQUEST
  - InvalidIdFormatException
  - InvalidRequestException
- 404 NOT FOUND
  - NotFoundException

## ACCOUNT-API > Account

### 회원상태 변경

- 가입
- 탈퇴
- 휴면
- 활동

#### 요청 예시

```
POST /account-api/account-account-state
```

#### Request

- 변경 시간은 현재 시간 기준으로 자동으로 입력
- Body

```
{
  "accountId" : "sampleId",
  "accountStateCode": "01",
}
```

## Response

- Body

```
{
  "header": {
    "resultCode": 201,
    "resultMessage": "AccountAccountState created successfully",
    "successful": true
  },
  "result": null,
  "totalCount": 0
}
```

## HTTP 응답코드

- 201 CREATED
- 400 BAD REQUEST
  - InvalidIdFormatException
- 404 NOT FOUND
  - NotFoundException

## 사용자 상태 조회

- 특정 사용자의 상태 변경 목록 조회

## 요청 예시

```
GET /account-account-state/{accountId}/list?page={page}&size={numOfRows}
```

## Request

- Parameter
  - 변경 시간은 현재 시간 기준으로 자동으로 입력

항목명	항목명(영문)	항목크기(최소값)	항목구분	샘플데이터	비고
사용자 아이디	accountId	40(5)	필수	sampleId	사용자 아이디
페이지	page		옵션	0	페이지 번호
한 페이지 결과수	numOfRows		옵션	10	한 페이지 결과 수

## Response

- Body



```
{
  "header": {
    "resultCode": 200,
    "resultMessage": "All account-account-states by account ID retrieved successfully",
    "successful": true
  },
  "result": [
    {
      "accountId": "accountId",
      "stateCode": "01",
      "changeAt": "2023-05-05T14:28:08"
    },
    {
      "accountId": "accountId",
      "stateCode": "03",
      "changeAt": "2023-06-06T18:43:39"
    }
  ],
  "totalCount": 5
}
```

#### HTTP 상태 코드

- 200 OK
- 400 BAD REQUEST
  - InvalidIdFormatException
- 404 NOT FOUND
  - NotFoundException