1. 교환 데이터 표준 포맷 (HTTP Response Format)

모든 Request에 대한 Response는 성공 처리와 오류 처리 2가지로 분류하고, 오류처리는 예외 처리로 구현함

1. 성공 처리

1) Field & Value

* “success”: true
* “response”: 응답 데이터  
   [ 응답데이터 종류에 따른 표현 방식 ]  
   - 단일 객체 : JSON Object  
   - Collection : JSON Array
* “error”: null

2) 정상 처리 예시 : 상품 조회

|  |
| --- |
| {  "success": true,  "response": {  "seq": 1,  "productId": 001,  "productName": "[톡별] 안동 명품 사과",  ...  },  "error": null } |

1. 오류 처리

1) Field & Value

* “success”: false
* “response”: null
* “error”:   
  - “status”: HTTP Response status code  
  - “message”: 오류 메시지 (조별 정의)

2) 오류처리 예시: 로그인 아이디 입력 누락

|  |
| --- |
| {  "success": false,  "response": null,  "error": {  "status": 400,  "message": "principal must be provided"  } } |

2. API 명세 (API Specification)

1. 회원 인증 API

|  |  |  |
| --- | --- | --- |
| Description | Method | URI |
| 회원 가입 | POST | /join |
| 로그인 | POST | /login |

* 회원 가입

|  |  |
| --- | --- |
| Method | POST |
| URI | /join |
| Specification | Request로 받은 데이터를 DB에 저장합니다. |
| Request Body | **[필드]**   * email : 사용자가 입력한 이메일 * userName : 사용자가 입력한 이름 * password : 사용자가 입력한 비밀번호   **[형태 – JSON]**  {  “email” : 문자열,  “userName” : 문자열,  “password” : 문자열  } |
| Response | - |

* 로그인

|  |  |
| --- | --- |
| Method | POST |
| URI | /login |
| Specification | Request로 받은 데이터를 DB와 비교하고 인증/인가합니다. |
| Request Body | **[필드]**   * email : 사용자가 입력한 이메일 * password : 사용자가 입력한 비밀번호   **[형태 – JSON]**  {  “email” : 문자열,  “password” : 문자열  } |
| Response Body | **[필드]**   * JWT token : 인증 토큰   **[형태]**  문자열 |

1. 상품 조회 API

|  |  |  |
| --- | --- | --- |
| Description | Method | URI |
| 전체 상품 조회 | GET | /products?offset={offset}&limit={limit} |
| 개별 상품 조회 | GET | /products/{productId} |

* 개별 상품은 상품 번호(productId)로 조회
* 무한스크롤 구현 파라미터는 레퍼런스 코드 고도화 구현에 따라 세부 수정 필요
* 전체 상품 조회

|  |  |
| --- | --- |
| Method | GET |
| URI | /products?offset={offset}&limit={limit} |
| Specification | DB에서 전체 상품 데이터를 반환합니다. |
| Request |  |
| Response Body | **[필드]**   * productId : 상품 번호 * productName : 상품 이름 * description : 상품 설명 * image : 상품 이미지 경로 * price : 상품 가격 * page : pagination 기준 페이지 번호 * totalCount : 지금까지 반환된 상품 개수   **[형태 – JSON Array]**  [ {  “productId” : 정수,  “productName” : 문자열,  “description” : 문자열,  “image” : 문자열,  “price” : 정수,  “page” : 정수,  “totalCount” : 정수  }, {  “productId” : 정수,  … }  ] |

* 개별 상품 조회

|  |  |
| --- | --- |
| Method | GET |
| URI | /products/{productId} |
| Specification | DB에서 특정 상품 데이터를 반환합니다. |
| Request | - |
| Response Body | **[필드]**   * productId : 상품 번호 * productName : 상품 이름 * description : 상품 설명 * image : 상품 이미지 경로 * price : 상품 가격   **[형태 – JSON]** {  “productId” : 정수,  “productName” : 문자열,  “description” : 문자열,  “image” : 문자열,  “price” : 정수  } |

1. 옵션 조회 API

|  |  |  |
| --- | --- | --- |
| Description | Method | URI |
| 상품 별 옵션 조회 | GET | /products/{productId}/option |

* 상품번호로 해당 상품에 대한 모든 옵션 정보 조회
* 상품 별 옵션 조회

|  |  |
| --- | --- |
| Method | GET |
| URI | products/{productId}/option |
| Specification | DB에서 특정 상품의 옵션 데이터를 반환합니다. |
| Request | - |
| Response Body | **[필드]**   * optionId : 옵션 번호 * productId : 상품 번호 * optionName : 옵션 이름 * price : 옵션 가격   **[형태 – JSON Array]**  [  {  “optionId” : 정수,  “productId” : 정수,  “optionName” : 문자열,  “price” : 정수  },  {  “optionId” : 정수,   …  } ] |

1. 장바구니 API

|  |  |  |
| --- | --- | --- |
| Description | Method | URI |
| 장바구니 담기 | POST | /cart/add |
| 장바구니 수정 | POST | /cart/update |
| 장바구니 조회 | GET | /cart |

* 장바구니는 유저 id 당 1개로 운영
* 장바구니 담기

|  |  |
| --- | --- |
| Method | POST |
| URI | /cart/add |
| Specification | Request로 받은 데이터를 DB에 저장합니다. |
| Request Body | **[필드]**   * cartListId : 장바구니에 담긴 옵션 인덱스 * cartId : 장바구니 번호 (회원 당 1개) * optionId : 옵션 번호 * quantity : 해당 옵션 수량 * price : 옵션 가격   **[형태 – JSON Array]**  [  {  “cartListId” : 정수,  “cartId” : 정수,   “optionId” : 정수,  “quantity” : 정수,   “price” : 정수,   },  {  “cartListId” : 정수,   …  } ] |
| Response | - |

* 장바구니 수정

|  |  |
| --- | --- |
| Method | POST |
| URI | /cart/update |
| Specification | Request로 받은 데이터를 DB와 비교하여 수정합니다. |
| Request Body | **[필드]**   * cartListId : 장바구니에 담긴 옵션 인덱스 * cartId : 장바구니 번호 (회원 당 1개) * optionId : 옵션 번호 * quantity : 해당 옵션 수량 * price : 옵션 가격   **[형태 – JSON Array]**  [  {  “cartListId” : 정수,  “cartId” : 정수,   “optionId” : 정수,  “quantity” : 정수,   “price” : 정수,   },  {  “cartListId” : 정수,   …  } ] |
| Response | - |

* 장바구니 조회

|  |  |
| --- | --- |
| Method | GET |
| URI | /cart |
| Specification | DB에서 장바구니에 담긴 상품들의 데이터와 상품명, 옵션명을 반환합니다. |
| Request | - |
| Response Body | **[필드]**   * cartListId : 장바구니에 담긴 옵션 인덱스 * cartId : 장바구니 번호 (회원 당 1개) * optionId : 옵션 번호 * quantity : 해당 옵션 수량 * price : 옵션 가격 * productName : 옵션 대표(상품) 이름 * optionName : 옵션 이름   **[형태 – JSON Array]**  [  {  “cartListId” : 정수,  “cartId” : 정수,   “optionId” : 정수,  “quantity” : 정수,   “price” : 정수,  “productName” : 문자열,  “optionName” : 문자열  },  {  “cartListId” : 정수,  …  } ] |

1. 상품 결제 및 주문 결과 API

|  |  |  |
| --- | --- | --- |
| Description | Method | URI |
| 상품 결제 | POST | /orders/save |
| 주문 결과 확인 | GET | /orders/{orderId} |

* 상품 결제

|  |  |
| --- | --- |
| Method | POST |
| URI | /orders/save |
| Specification | Request로 받은 데이터를 DB에 저장합니다. |
| Request Body | **[필드]**   * orderListId : 주문 리스트 인덱스 * optionId : 옵션 번호 * quantity : 주문 수량 * price : 옵션 가격   **[형태 – JSON Array]**  [  {  “orderListId” : 정수,  “optionId” : 정수,  “quantity” : 정수,  “price” : 정수  },  {  “orderListId” : 정수,  …  }  ] |
| Response | - |

* 주문 결과 확인

|  |  |
| --- | --- |
| Method | GET |
| URI | /orders/{orderId} |
| Specification | DB에서 주문 내역(상품명, 옵션명, 금액)을 반환합니다. |
| Request | - |
| Response Body | **[필드]**   * orderId : 주문 번호 * price : 총 주문 금액 * productName : 주문 대표 상품 이름 * optionName : 주문 내역 첫번째 옵션 이름   **[형태 – JSON]**  {  “orderId” : 정수,  “price” : 정수,  “productName” : 문자열,  “optionName” : 문자열  } |