프로젝트 2 API 명세서

**회원 API**

1. 회원가입

|  |  |
| --- | --- |
| Method | POST |
| URI | /users/join |
| HTTP status code | 성공 201 / 실패 400 |
| Request Body | {  email: 사용자 입력 이메일,  password: 사용자 입력 패스워드  } |
| Response Body |  |

1. 로그인

|  |  |
| --- | --- |
| Method | POST |
| URI | /users/login |
| HTTP status code | 성공 200 / 실패 400 or 401 |
| Request Body | {  email: 사용자 입력 이메일,  password: 사용자 입력 패스워드  } |
| Response Body | cookie: JWT Token |

1. 비밀번호 초기화 요청

|  |  |
| --- | --- |
| Method | POST |
| URI | /users/reset |
| HTTP status code | 성공 200 / 실패 400 or 401 |
| Request Body | {  email: 사용자 입력 이메일,  } |
| Response Body | {  email: 입력받은 이메일,  } |

1. 비밀번호 초기화 (=수정)

|  |  |
| --- | --- |
| Method | PUT |
| URI | /users/reset |
| HTTP status code | 성공 200 / 실패 400 |
| Request Body | {  email: 아까 돌려받은 이메일,  password: 사용자 입력 패스워드  } |
| Response Body |  |

**도서 API**

1. 전체 도서 조회

페이징: n개씩 조회, m페이지

recent=true일 경우 신간만 조회 (신간: 출간일 기준 1달 이내)

category=카테고리 있으면 해당 카테고리만 조회, 없으면 전체 도서 조회

|  |  |
| --- | --- |
| Method | GET |
| URI | /books/?recent=true&category={category}&currentPage={현재페이지}&limit={페이지당도서수} |
| HTTP status code | 성공 200 / 실패 404 |
| Request Body |  |
| Response Body | // 전체 도서 목록에는 도서의 상세 정보 포함.  // 프론트엔드에서 필요한 데이터만 선별하여 구현할 것.  [  {  book\_id: 도서 id  main\_img: 도서 대표이미지 링크,  title: 도서 제목,  author: 작가 이름,  summary: 도서 요약 설명,  price: 도서 가격,  likes: 좋아요 수  published\_date: 출간일  category: 카테고리  },  {  book\_id: 도서 id  main\_img: 도서 대표이미지 링크,  title: 도서 제목,  author: 작가 이름,  summary: 도서 요약 설명,  price: 도서 가격,  likes: 좋아요 수  published\_date: 출간일  },  ….  ] |
|  |  |

1. 개별 도서 조회(이미지 여러장 경로 추가 필요)

|  |  |
| --- | --- |
| Method | GET |
| URI | /books/{bookId} |
| HTTP status code | 성공 200 / 실패 404 |
| Request Body |  |
| Response Body | {  book\_id: 도서 id  title: 도서 제목,  category: 카테고리,  format: 포맷(종이책/전자책 등),  author: 작가 이름,  isbn: isbn,  pages: 쪽수,  index: 목차,  summary: 도서 요약 설명,  description: 도서 상세 설명,  price: 도서 가격,  likes: 좋아요 수  liked: 내가 좋아요를 했는지 여부  published\_date: 출간일  img: 이미지 경로의 id (picsum img #id)  } |

**카테고리 API**

1. 카테고리 목록 조회

|  |  |
| --- | --- |
| Method | GET |
| URI | /categories |
| HTTP status code | 성공 200 / 실패 400 or 404 |
| Request Body |  |
| Response Body | [ “카테고리1”, “카테고리2”, … ] |

**좋아요 API**

1. 좋아요 추가

|  |  |
| --- | --- |
| Method | POST |
| URI | /likes/{bookId} |
| HTTP status code | 성공 200 / 실패 404 |
| Request Body |  |
| Response Body |  |

1. 좋아요 삭제

|  |  |
| --- | --- |
| Method | DELETE |
| URI | /likes/{bookId} |
| HTTP status code | 성공 200 / 실패 404 |
| Request Body |  |
| Response Body |  |

**장바구니 API**

1. 장바구니 담기

|  |  |
| --- | --- |
| Method | POST |
| URI | /carts |
| HTTP status code | 성공 201 / 실패 500 |
| Request Body | {  id: 장바구니 도서 id,  bookId: 도서 id,  count: 도서 수량,  } |
| Response Body |  |

1. 장바구니 조회

|  |  |
| --- | --- |
| Method | GET |
| URI | /carts |
| HTTP status code | 성공 200 / 실패 404 |
| Request Body |  |
| Response Body | [  {  id: 장바구니 도서 id,  bookId: 도서 id,  count: 도서 수량,  title: 도서 제목,  summary: 도서 요약 설명,  price: 도서 가격,  main\_img: 대표 이미지 경로  },  {  id: 장바구니 도서 id,  bookId: 도서 id,  count: 도서 수량,  title: 도서 제목,  summary: 도서 요약 설명,  price: 도서 가격,  main\_img: 대표 이미지 경로  },  …  ] |

1. 장바구니 도서 삭제

|  |  |
| --- | --- |
| Method | DELETE |
| URI | /cart/{bookId} |
| HTTP status code | 성공 200 / 실패 500 |
| Request Body |  |
| Response Body |  |

1. 체크한 상품 조회

|  |  |
| --- | --- |
| Method | GET |
| URI | /cart?checked=true |
| HTTP status code | 성공 200 / 실패 404 |
| Request Body | [cartItemId, cartItemId, …] |
| Response Body | [  {  cartItemId: 장바구니 도서 id,  bookId: 도서 id,  count: 도서 수량,  title: 도서 제목,  summary: 도서 요약 설명,  price: 도서 가격,  main\_img: 대표 이미지 경로  },  {  cartItemId: 장바구니 도서 id,  bookId: 도서 id,  count: 도서 수량,  title: 도서 제목,  summary: 도서 요약 설명,  price: 도서 가격,  main\_img: 대표 이미지 경로  },  …  ] |

**결제 API**

1. 결제하기

결제하고 나면 장바구니에서 주문한 상품은 사라져야 한다.

|  |  |
| --- | --- |
| Method | POST |
| URI | /orders |
| HTTP status code | 성공 200 / 실패 500 |
| Request Body | {  items: [  {  cart\_id: 장바구니 id,  book\_id: 도서 id,  count: 수량  }, …  ],  delivery: {  address: 주소,  receiver: 이름,  contact: “010-0000-0000”,  }  totalPrice: 총 금액  } |
| Response Body |  |

1. 주문 목록 조회

|  |  |
| --- | --- |
| Method | GET |
| URI | /orders |
| HTTP status code | 성공 200 / 실패 500 |
| Request Body |  |
| Response Body | [  {  created\_at: 주문일자,  delivery: {  address: 주소,  receiver: 이름,  contact, 연락처,  },  bookTitle: 대표 책 제목,  totalPrice: 결제 금액,  totalCount: 총 수량  }, ….  ] |

1. 주문 상세 상품 조회

|  |  |
| --- | --- |
| Method | GET |
| URI | /orders/{orderId} |
| HTTP status code | 성공 200 / 실패 500 |
| Request Body |  |
| Response Body | [  {  book\_id: 도서 id,  bookTitle: 도서 제목,  author: 작가명,  price: 가격,  count: 수량  }, …  ] |