**회원 API**

1. 회원 가입

|  |  |
| --- | --- |
| Method | POST |
| URL | /users/join |
| HTTP status code | 성공 201 |
| Request Body | {   email: “사용자가 입력한 이메일”,  Password: “사용자가 입력한 비밀번호”  } |
| Response Body |  |

1. 로그인

|  |  |
| --- | --- |
| Method | POST |
| URL | /users /join |
| HTTP status code | 성공 200 |
| Request Body | {   email: “사용자가 입력한 이메일”,  Password: “사용자가 입력한 비밀번호”  } |
| Response cookie | JWT Token |

1. 비밀번호 초기화 요청

|  |  |
| --- | --- |
| Method | POST |
| URL | /users /reset |
| HTTP status code | 성공 200 |
| Request Body | {   email: “사용자가 입력한 이메일”,  } |
| Response Body |  |

1. 비밀번호 초기화 (=수정)

|  |  |
| --- | --- |
| Method | PUT |
| URL | /users /reset |
| HTTP status code | 성공 200 |
| Request Body | {   password: “사용자가 입력한 비밀번호”  } |
| Response Body |  |

**도서 API**

1. 전체 도서 조회

|  |  |
| --- | --- |
| Method | GET |
| URL | /books |
| HTTP status code | 성공 200 |
| Request Body |  |
| Response Body | {  {  id: 도서 id,  title: “도서 제목”,  img: 이미지 id,  summary: “요약 설명”,  author: “도서 작가”,  price: 가격,  likes: 좋아요 수,  pubDate: “출간일”  },  {  id: 도서 id,  title: “도서 제목”,  img: 이미지 id,  summary: “요약 설명”,  author: “도서 작가”,  price: 가격,  likes: 좋아요 수,  pubDate: “출간일”  }  …  } |

1. 개별 도서 조회

|  |  |
| --- | --- |
| Method | GET |
| URL | /books/{bookId} |
| HTTP status code | 성공 200 |
| Request Body |  |
| Response Body | {  {  id: 도서 id,  title: “도서 제목”,  img: 이미지 id,  category: “카테고리”,  format: “포맷”,  isbn: “isbn”,  summary: “요약 설명”,  description: “상세 설명”,  author: “도서 작가”,  pages: 쪽 수  index: “목차”,  price: 가격,  likes: 좋아요 수,  liked: boolean  }  } |

1. 카테고리별 도서 목록 조회

* new: true => 신간 조회 (기준: 출간일 30일 이내)

// 이미지 경로, 8개씩 보내줘야함.

// 카테고리 id, 어떻게 알고 보내줄까?

|  |  |
| --- | --- |
| Method | GET |
| URL | /books/?{categoryId={categoryId}&new={boolean} |
| HTTP status code | 성공 200 |
| Request Body |  |
| Response Body | {  {  id: 도서 id,  title: “도서 제목”,  summary: “요약 설명”,  author: “도서 작가”,  price: 가격,  likes: 좋아요 수,  pubDate: “출간일”  },  {  id: 도서 id,  title: “도서 제목”,  summary: “요약 설명”,  author: “도서 작가”,  price: 가격,  likes: 좋아요 수,  pubDate: “출간일”  }  } |

**카테고리 API**

1. 카테고리 전체 조회

|  |  |
| --- | --- |
| Method | GET |
| URL | /category |
| HTTP status code | 성공 200 |
| Request Body |  |
| Response Body | {  {  id: 0,  name: “동화”  },  {  id: 1,  name: “소설”  },  } |

**좋아요 API**

1. 좋아요 추가

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

1. 좋아요 취소

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

**장바구니 API**

1. 장바구니 담기

|  |  |
| --- | --- |
| Method | POST |
| URL | /cart |
| HTTP status code | 성공 201 |
| Request Body | {  bookId: 도서 id,  count: 수량  } |
| Response Body |  |

1. 장바구니 조회

|  |  |
| --- | --- |
| Method | GET |
| URL | /cart |
| HTTP status code | 성공 200 |
| Request Body |  |
| Response Body | {  {  cartItemId: 장바구니 도서 id,  bookId: 도서 id,  title: “도서 제목”,  summary: “도서 요약”,  count: 수량,  price: 가격  },  {  cartItemId: 장바구니 도서 id,  bookId: 도서 id,  title: “도서 제목”,  summary: “도서 요약”,  count: 수량,  price: 가격  }  …  } |

1. 장바구니 도서 삭제

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

1. (장바구니에서 선택한) 주문 예상 상품 목록 조회

|  |  |
| --- | --- |
| Method | GET |
| URL | / |
| HTTP status code | 성공 200 |
| Request Body | { cartItemId, cartItemId, cartItemId, … } |
| Response Body | {  {  cartItemId: 장바구니 도서 id,  bookId: 도서 id,  title: “도서 제목”,  summary: “도서 요약”,  count: 수량,  price: 가격  },  {  cartItemId: 장바구니 도서 id,  bookId: 도서 id,  title: “도서 제목”,  summary: “도서 요약”,  count: 수량,  price: 가격  }  …  } |

**결제(주문) API**

1. 결제하기 = 주문하기 = 주문 등록 = 데이터베이스 주문 insert

= 장바구니에서 주문된 상품은 delete

|  |  |
| --- | --- |
| Method | POST |
| URL | /orders |
| HTTP status code | 성공 200 |
| Request Body | {  items: {{  cartItemId: 장바구니 도서 id  bookId: 도서 id,  count: 수량  }, {  cartItemId: 장바구니 도서 id  bookId: 도서 id,  count: 수량  }  },  delivery: {  address: “주소”,  receiver: “이름”,  contact: “010-0000-0000”  },  totalPrice: 총금액  } |
| Response Body |  |

1. 주문 목록(내역) 조회

|  |  |
| --- | --- |
| Method | GET |
| URL | /orders |
| HTTP status code | 성공 200 |
| Request Body |  |
| Response Body | {  {  created\_at: “주문일자”,  delivery : {  address: “주소”,  receiver: “이름”,  contact: “전화번호”,  },  bookTitle: “대표 책 제목”,  totalPrice: 결제 금액,  totalCount: 총 수량  }  } |

1. 주문 상세 상품 조회

|  |  |
| --- | --- |
| Method | GET |
| URL | /orders/{orderId} |
| HTTP status code | 성공 200 |
| Request Body |  |
| Response Body | {  {  bookId: 도서 id,  bookTitle: “도서 제목”,  author: “작가명”,  price: 가격  count: 수량  },  {  bookId: 도서 id,  bookTitle: “도서 제목”,  author: “작가명”,  price: 가격  count: 수량  }  } |