**1. 회원가입 API**

|  |  |
| --- | --- |
| method | POST |
| URL | http://{SERVER\_URL}/join |

Request Header

|  |  |
| --- | --- |
| Content-Type | application/json |

Request Body

|  |
| --- |
| {      "userId": String(nullable = false, length = 10),      "password": String(nullable = false, length = 프론트에서 결정 필요),      "nickname": String(nullable = false, length = 10)  } |

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

+ String 빈문자열 “” 가 요청되지 않도록 프론트에서 처리해줘야 함

+ “password” 글자 수 제한 몇 글자로 할 건지 정해서 프론트에서 처리해줘야 함(서버에서 ”password” String 길이를 60으로 설정한 이유는 BCrypt 알고리즘을 사용하여 비밀번호를 해싱하면 항상 60글자의 해시 문자열이 생성되고 이 값을 DB users테이블에 저장해야 하기 때문임)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: null |

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명 이미 존재하는 userId인 경우

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명 서버에서 요청을 처리하는 중 오류가 발생한 경우

**2. 로그인 API**

|  |  |
| --- | --- |
| method | POST |
| URL | http://{SERVER\_URL}/login |

Request Header

|  |  |
| --- | --- |
| Content-Type | application/json |

Request Body

|  |
| --- |
| {      "userId": String(nullable = false, length = 10),      "password": String(nullable = false, length = 프론트에서 결정 필요)  } |

텍스트, 폰트, 스크린샷, 번호이(가) 표시된 사진

자동 생성된 설명

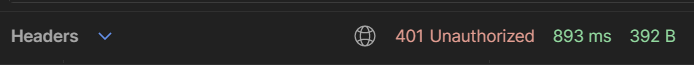
+ “password” 글자 수 제한 몇 글자로 할 건지 정해서 프론트에서 처리해줘야 함(서버에서 ”password” String 길이를 60으로 설정한 이유는 BCrypt 알고리즘을 사용하여 비밀번호를 해싱하면 항상 60글자의 해시 문자열이 생성되고 이 값을 DB users테이블에 저장해야 하기 때문임)

Response Headers

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명 로그인 성공하는 경우

🡪 “Authorization”값(JWT 토큰), “userId”값(로그인한 사용자의 userId) 기기에 저장하기

 로그인 실패하는 경우

**3. nickname update API: nickname 수정**

|  |  |
| --- | --- |
| method | PATCH |
| URL | http://{SERVER\_URL}/users/{userId}/nickname |

{userId}부분에 수정하려는 nickname의 userId 넣기

Request Header

|  |  |
| --- | --- |
| Content-Type | application/json |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Request Body

|  |
| --- |
| {      "nickname": String(nullable = false, length = 10)  } |

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

+ String 빈문자열 “” 가 요청되지 않도록 프론트에서 처리해줘야 함

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: null |

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 nickname이 변경되어 DB에 저장됨

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 요청한 userId의 리소스를 찾을 수 없는 경우

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 서버에서 요청을 처리하는 중 오류가 발생한 경우텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명 JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**4. password update API: password 수정**

|  |  |
| --- | --- |
| method | PATCH |
| URL | http://{SERVER\_URL}/users/{userId}/password |

{userId}부분에 수정하려는 password의 userId 넣기

Request Header

|  |  |
| --- | --- |
| Content-Type | application/json |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Request Body

|  |
| --- |
| {      "password": String(nullable = false, length = 프론트에서 결정 필요)  } |

텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

+ String 빈문자열 “” 가 요청되지 않도록 프론트에서 처리해줘야 함

+ “password” 글자 수 제한 몇 글자로 할 건지 정해서 프론트에서 처리해줘야 함(서버에서 ”password” String 길이를 60으로 설정한 이유는 BCrypt 알고리즘을 사용하여 비밀번호를 해싱하면 항상 60글자의 해시 문자열이 생성되고 이 값을 DB users테이블에 저장해야 하기 때문임)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: null |

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

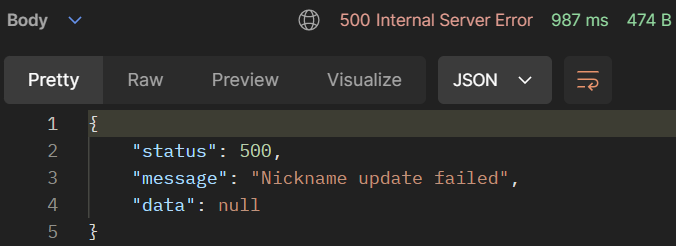
자동 생성된 설명 password가 변경되어 DB에 저장됨

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 요청한 userId의 리소스를 찾을 수 없는 경우

서버에서 요청을 처리하는 중 오류가 발생한 경우텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명 JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우



**5. nickname 조회 API**

|  |  |
| --- | --- |
| method | GET |
| URL | http://{SERVER\_URL}/users/{userId}/nickname |

{userId} 부분에 조회하려는 nickname의 userId 넣기

Request Header

|  |  |
| --- | --- |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: String |

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명 해당 userId의 nickname 조회 성공하는 경우

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명요청한 userId의 리소스를 찾을 수 없는 경우

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 서버에서 요청을 처리하는 중 오류가 발생한 경우

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명 JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**6.password check API: 확인하고자하는 password가 저장되어있는 password와 같은지 비교**

|  |  |
| --- | --- |
| method | POST |
| URL | http://{SERVER\_URL}/users/check/password |

Request Header

|  |  |
| --- | --- |
| Content-Type | application/json |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Request Body

|  |
| --- |
| {      "userId": " String(nullable = false, length = 10),      "password": String(nullable = false, length = 프론트에서 결정 필요)  } |

텍스트, 폰트, 스크린샷, 번호이(가) 표시된 사진

자동 생성된 설명

+ “password” 글자 수 제한 몇 글자로 할 건지 정해서 프론트에서 처리해줘야 함(서버에서 ”password” String 길이를 60으로 설정한 이유는 BCrypt 알고리즘을 사용하여 비밀번호를 해싱하면 항상 60글자의 해시 문자열이 생성되고 이 값을 DB users테이블에 저장해야 하기 때문임)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: String |

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 입력한 password 값 = 입력한 userId의 password 값

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 입력한 password 값 ≠ 입력한 userId의 password 값

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 요청한 userId의 리소스를 찾을 수 없는 경우

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명 서버에서 요청을 처리하는 중 오류가 발생한 경우

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**7.user delete: 유저 탈퇴시 해당 유저 정보 삭제**

|  |  |
| --- | --- |
| method | DELETE |
| URL | http://{SERVER\_URL}/users/{userId} |

{userId} 부분에 삭제하려는 유저의 userId 넣기

Request Header

|  |  |
| --- | --- |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: null |

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

해당 userId의 모든 정보 삭제됨, 해당 유저가 등록한 반려동물 정보도 모두 삭제됨

**텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명** 요청한 userId의 리소스를 찾을 수 없는 경우

**텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명** 서버에서 요청을 처리하는 중 오류가 발생한 경우

**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명** JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**8. pet add API: 반려동물 새 프로필 추가, DB의 pets 테이블에 값 저장(내부 로직 수정)**

|  |  |
| --- | --- |
| method | POST |
| URL | http://{SERVER\_URL}/users/{userId}/pet |

{userId} 부분에 반려동물을 등록하려는 유저의 userId 넣기

Request Header

|  |  |
| --- | --- |
| Content-Type | application/json |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Request Body

|  |
| --- |
| {  "petName": String(nullable = false, length = 10),  "breed": String(nullable = false, length = 6),  "age": Byte(nullable = false),  "currentWeight": BigDecimal(nullable = false, precision =4, scale = 1),  "isNeutered": Boolean(nullable = false),  "gender": Character(nullable = false, length = 1, male이면 m, female이면 f로 저장),  "hasDiabetes": Boolean(nullable = false),  "insulinTime1": LocalTime(nullable = true),  "insulinTime2": LocalTime(nullable = true),  "insulinTime3": LocalTime(nullable = true),  "heartwormShotDate": LocalDate(nullable = true),  "heartwormMedicineDate": LocalDate(nullable = true)  } |

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

+ String 빈문자열 “” 가 요청되지 않도록 프론트에서 처리해줘야 함

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: null |

**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

pets 테이블에서 petId는 userId+petName으로 자동 생성해서 저장됨, Boolean 값은 true는 1로 저장, false는 0으로 저장됨

동시에 weight\_records 테이블에 현재날짜로 currentWeight 값이 weight으로 저장됨

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명 이미 존재하는 petId(userId+petName)인 경우

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

insulinTime1, insulinTime2, insulinTime3에 값이 앞에서부터 입력되지 않은 경우

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명



요청한 userId의 리소스를 찾을 수 없는 경우 / 해당 petId의 리소스를 찾을 수 없는 경우

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 서버에서 요청을 처리하는 중 오류가 발생한 경우

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**9. petId 조회 API: 해당 userId의 등록한 반려동물들의 petId들 조회**

|  |  |
| --- | --- |
| method | GET |
| URL | http://{SERVER\_URL}/users/{userId}/petId |

{userId} 부분에 petId들을 조회하려는 유저의 userId 넣기

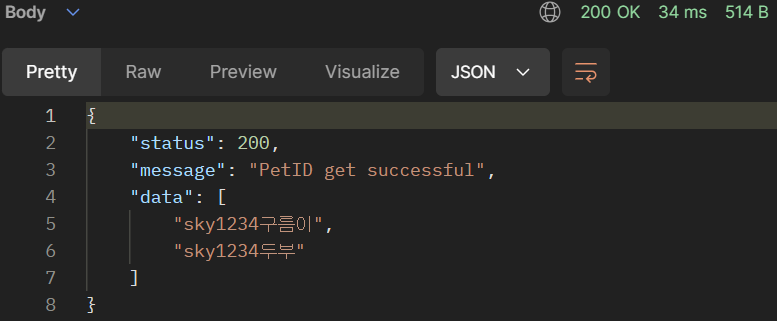
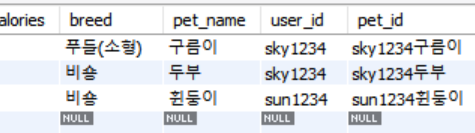
Request Header

|  |  |
| --- | --- |
| Authorization | {"Bearer " + token} |

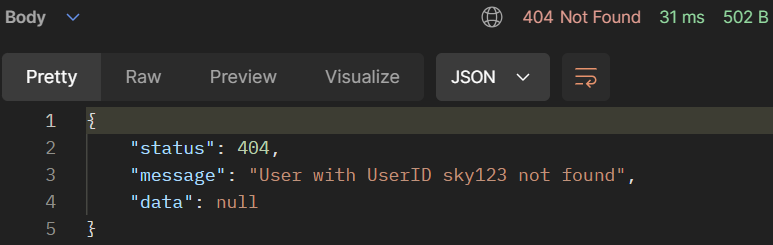
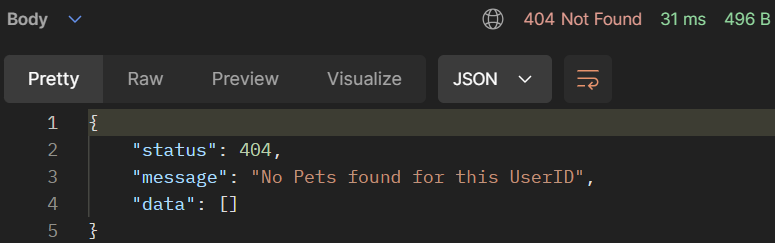
{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: List<String> |

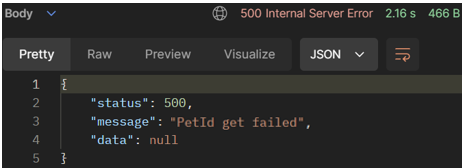
해당 userId의 유저가 등록한 모든 반려동물의 petId 조회 성공하는 경우

요청한 userId에 해당하는 리소스를 찾을 수 없는 경우 / 해당 userId가 등록한 반려동물이 없는 경우

서버에서 요청을 처리하는 중 오류가 발생한 경우텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명 JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우



**10. pet profile 조회 API: pet 기본 프로필 정보 가져오기**

|  |  |
| --- | --- |
| method | GET |
| URL | http://{SERVER\_URL}/pets/{petId}/profile |

{petId} 부분에 pet profile을 조회하려는 반려동물의 petId 넣기

Request Header

|  |  |
| --- | --- |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: {          "petName": String,          "breed": String,          "gender": Character,          "age": Byte,          "currentWeight": BigDecimal,          "isNeutered": Boolean,          "hasDiabetes": Boolean  } |

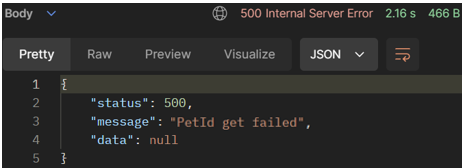
텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 pet profile 조회 성공하는 경우

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 요청한 petId의 리소스를 찾을 수 없는 경우

서버에서 요청을 처리하는 중 오류가 발생한 경우



텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명 JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**11.pet update API: 반려동물 프로필 수정, DB의 pets 테이블에 값 수정(내부 로직 수정)**

|  |  |
| --- | --- |
| method | PATCH |
| URL | http://{SERVER\_URL}/pets/{petId} |

{petId} 부분에 프로필을 수정하려는 반려동물의 petId 넣기

Request Header

|  |  |
| --- | --- |
| Content-Type | application/json |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Request Body

|  |
| --- |
| {  "breed": String(nullable = true, length = 6),  "age": Byte(nullable = true),  "currentWeight": BigDecimal(nullable = true, precision =4, scale = 1),  "isNeutered": Boolean(nullable = true),  "gender": Character(nullable = true, length = 1, male이면 m, female이면 f로 저장),  "hasDiabetes": Boolean(nullable = true),  "insulinTime1": LocalTime(nullable = true),  "insulinTime2": LocalTime(nullable = true),  "insulinTime3": LocalTime(nullable = true),  "heartwormShotDate": LocalDate(nullable = true),  "heartwormMedicineDate": LocalDate(nullable = true)  } |

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

+ String 빈문자열 “” 가 요청되지 않도록 프론트에서 처리해줘야 함

+ 바꾸지 않을 값은 null로 함

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: null |

**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

response body의 각 입력 값 중 null값인 아닌 값들만 변경되어 저장됨, null값인 값들은 저장되어 있는 값 그대로 유지됨

만약 currentWeight가 null값인 아니라면 pets 테이블의 current\_weight 값이 변경되어 저장되고, 해당 petId의 현재날짜로 weight\_records 테이블에 저장되어 있는 데이터가 있다면 해당 데이터의 weight 값만 변경되어 저장되고 만약 없다면 해당 petId, 현재날짜로 새롭게 weight 기록을 weight\_records 테이블에 추가함

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

요청한 petId(ex: sky1234두)의 리소스를 찾을 수 없는 경우

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

insulinTime1, insulinTime2, insulinTime3에 값이 앞에서부터 저장되지 않은 경우

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

서버에서 요청을 처리하는 중 오류가 발생한 경우

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**12. pet delete API: 반려동물 프로필 삭제, DB의 pets 테이블에서 삭제**

|  |  |
| --- | --- |
| method | DELETE |
| URL | http://{SERVER\_URL}/pets/{petId} |

{petId} 부분에 프로필을 삭제하려는 반려동물의 petId 넣기

Request Header

|  |  |
| --- | --- |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: null |

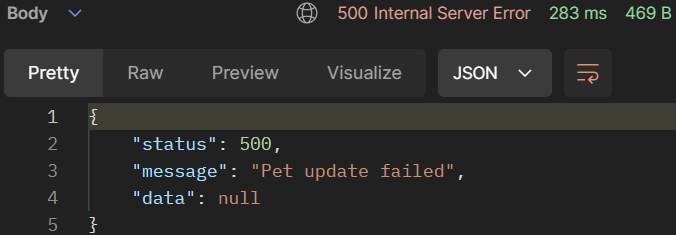
텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 해당 petId의 모든 정보 삭제됨

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 요청한 petId의 리소스를 찾을 수 없는 경우

서버에서 요청을 처리하는 중 오류가 발생한 경우



텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명 JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**13. daily-record add API: 날짜별 ①처방전(진단명, 처방약), ②몸무게, ③혈당, ④특이사항 각 DB의 prescription\_records, weight\_records, blood\_sugar\_level\_records, special\_note\_records 테이블에 값 저장, pets 테이블의 current\_weight, current\_blood\_sugar\_level 업데이트**

|  |  |
| --- | --- |
| method | POST |
| URL | http://{SERVER\_URL}/pets/{petId}/daily-record |

{petId} 부분에 날짜별 기록을 할 반려동물의 petId 넣기

Request Header

|  |  |
| --- | --- |
| Content-Type | application/json |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Request Body

|  |
| --- |
| {  "recordDate": LocalDate(nullable = false),  “diagnosis”: String(nullable = true, length = 20),  “medicine”: String(nullable = true, length = 20),  “weight”: BigDecimal(nullable = true, precision = 4, scale = 1),  “bloodSugarLevel”: Short(nullable = true),  “specialNote”: String(nullable = true, length = 200)  } |

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

+ “diagnosis”, “weight”, “bloodSugarLevel”, “specialNote” 중 적어도 한가지는 입력하는 것이 필수인 것을 프론트에서 안내해줘야 함

+ “medicine”에 값을 입력하는 경우 무조건 “diagnosis”에도 값을 입력하는 것이 필수인 것을 프론트에서 안내해줘야 함

+ “diagnosis”에 값을 입력하는 경우 “medicine”에 값을 입력하는 것은 선택사항임을 프론트에서 안내해줘야 함

+ String 빈문자열 “” 가 요청되지 않도록 프론트에서 처리해줘야 함

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: null |

**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

해당 petId의 날짜별 처방전(진단명, 처방약), 몸무게, 혈당, 특이사항 각 DB의 해당 테이블에 값 저장됨

만약 몸무게, 혈당의 값이 저장된다면 각 해당 테이블에서 최신 값을 다시 조회해 pets 테이블의 current\_weight, current\_blood\_sugar\_level에 저장됨

텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 요청한 petId의 리소스를 찾을 수 없는 경우

**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명** 요청한 petId, "recordDate"로 이미 daily-record를 등록한 경우

**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

“diagnosis”, “weight”, “bloodSugarLevel”, “specialNote” 모두 null인 경우 / “diagnosis”가 null이면서 “medicine”은 null이 아닌 경우

**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명** 서버에서 요청을 처리하는 중 오류가 발생한 경우

**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명** JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**14. daily-record update API: 해당 날짜별 ①처방전(진단명, 처방약), ②몸무게, ③혈당, ④특이사항 각 DB의 prescription\_records, weight\_records, blood\_sugar\_level\_records, special\_note\_records 테이블에 값 수정, pets 테이블의 current\_weight, current\_blood\_sugar\_level 업데이트**

|  |  |
| --- | --- |
| method | PATCH |
| URL | http://{SERVER\_URL}/pets/{petId}/daily-record |

{petId} 부분에 날짜별 기록을 수정할 반려동물의 petId 넣기

Request Header

|  |  |
| --- | --- |
| Content-Type | application/json |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Request Body

|  |
| --- |
| {  "recordDate": LocalDate(nullable = false),  “diagnosis”: String(nullable = true, length = 20),  “medicine”: String(nullable = true, length = 20),  “weight”: BigDecimal(nullable = true, precision = 4, scale = 1),  “bloodSugarLevel”: Short(nullable = true),  “specialNote”: String(nullable = true, length = 200)  } |

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

+ “diagnosis”, “medicine”, “weight”, “bloodSugarLevel”, “specialNote” 중 적어도 한가지는 입력하는 것이 필수인 것을 프론트에서 안내해줘야 함

+ 해당 날짜에 처방전 기록이 없는 경우 “medicine”에 값을 입력한다면 무조건 “diagnosis”에도 값을 입력하는 것이 필수인 것을 프론트에서 안내해줘야 함

+ String 빈문자열 “” 가 요청되지 않도록 프론트에서 처리해줘야 함

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: null |

**텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

해당 petId의 날짜별 처방전(진단명, 처방약), 몸무게, 혈당, 특이사항 각 DB의 해당 테이블에 값 업데이트 또는 생성됨

만약 몸무게, 혈당의 값이 업데이트 또는 생성된다면 각 해당 테이블에서 최신 값을 다시 조회해 pets 테이블의 current\_weight, current\_blood\_sugar\_level에 저장됨

**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 멀티미디어 소프트웨어, 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

요청한 petId의 리소스를 찾을 수 없는 경우 / 요청한 (petId, "recordDate")의 리소스를 찾을 수 없는 경우

**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

“diagnosis”, “medicine”, “weight”, “bloodSugarLevel”, “specialNote” 모두 null인 경우 / 해당 petId, "recordDate"에 처방전 기록은 없는데(다른 daily-record기록은 적어도 한가지 있음) “medicine”은 null이 아니고 “diagnosis”는 null인 경우

**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명** 서버에서 요청을 처리하는 중 오류가 발생한 경우

**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명** JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**15. daily-record 조회 API: 해당 날짜의 ①처방전(진단명, 처방약), ②몸무게, ③혈당, ④특이사항 조회**

|  |  |
| --- | --- |
| method | GET |
| URL | http://{SERVER\_URL}/pets/{petId}/daily-record/{recordDate} |

{petId} 부분에 daily-record를 조회하려는 반려동물의 petId 넣기 {recordDate} 부분에 daily-record를 조회하려는 해당 날짜 넣기(형식: 0000-00-00, 예시: 2024-08-11)

Request Header

|  |  |
| --- | --- |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: {  "recordDate": LocalDate,  “diagnosis”: String,  “medicine”: String,  “weight”: BigDecimal,  “bloodSugarLevel”: Short,  “specialNote”: String  } |

**텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

해당 petId, "recordDate"로 daily-record 조회 성공하는 경우

해당 petId, "recordDate”로 저장된 처방전(진단명, 처방약), 몸무게, 혈당, 특이사항 조회됨 저장되어 있지 않은 값은 null로 조회됨

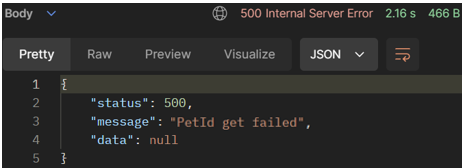
텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

요청한 petId의 리소스를 찾을 수 없는 경우 / 요청한 petId, "recordDate"의 리소스를 찾을 수 없는 경우

서버에서 요청을 처리하는 중 오류가 발생한 경우



**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명** JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**16. daily-record delete API: 해당 날짜의 ①처방전(진단명, 처방약), ②몸무게, ③혈당, ④특이사항 각 DB의 prescription\_records, weight\_records, blood\_sugar\_level\_records, special\_note\_records 테이블에서 삭제, pets 테이블의 current\_weight, current\_blood\_sugar\_level 업데이트**

|  |  |
| --- | --- |
| method | DELETE |
| URL | http://{SERVER\_URL}/pets/{petId}/daily-record/{recordDate} |

{petId} 부분에 daily-record를 삭제하려는 반려동물의 petId 넣기 {recordDate} 부분에 daily-record를 삭제하려는 해당 날짜 넣기(형식: 0000-00-00, 예시: 2024-08-11)

Request Header

|  |  |
| --- | --- |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: null |

**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

해당 petId의 날짜별 처방전(진단명, 처방약), 몸무게, 혈당, 특이사항 각 DB의 해당 테이블 레코드 삭제됨

만약 몸무게, 혈당의 값이 삭제된다면 각 해당 테이블에서 최신 값을 다시 조회해 pets 테이블의 current\_weight, current\_blood\_sugar\_level에 저장됨

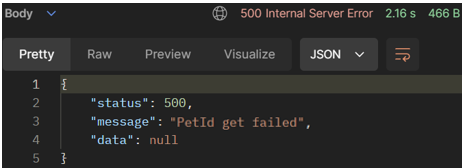
텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명 텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

요청한 petId 의 리소스를 찾을 수 없는 경우 / 요청한 (petId, "recordDate")의 리소스를 찾을 수 없는 경우

서버에서 요청을 처리하는 중 오류가 발생한 경우



**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명** JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**17. pet** **profile summary 조회 API: pet 기본 프로필 정보 중에 반려동물이름, 몸무게, 나이 가져오기**

|  |  |
| --- | --- |
| method | GET |
| URL | http://{SERVER\_URL}/pets/{petId}/profile/summary |

{petId} 부분에 pet summary profile을 조회하려는 반려동물의 petId 넣기

Request Header

|  |  |
| --- | --- |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: {          "petName": String,          "age": Byte,          "currentWeight": BigDecimal,  } |

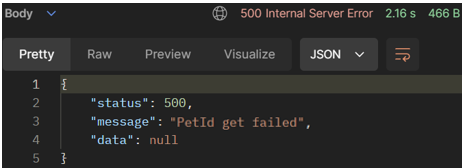
**텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명** pet profile summary 조회 성공하는 경우

**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명** 요청한 petId의 리소스를 찾을 수 없는 경우

서버에서 요청을 처리하는 중 오류가 발생한 경우



**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명** JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**18. 인슐린 주사시간1 delete API: 인슐린 주사시간(최대 3개) pets 테이블의 칼럼(insulin\_time1)값 삭제, 나머지 insulin\_time 저장위치 조정**

|  |  |
| --- | --- |
| method | DELETE |
| URL | http://{SERVER\_URL}/pets/{petId}/insulin-time1 |

{petId} 부분에 insulin-time1의 값을 삭제하려는 반려동물의 petId 넣기

Request Header

|  |  |
| --- | --- |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: null |

**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

insulin-time1의 값이 삭제되고, insulin-time2에 저장된 값이 insulin-time1에 다시 저장되고 insulin-time3에 저장된 값이 insulin-time2에 다시 저장되고 insulin-time3는 null이 됨

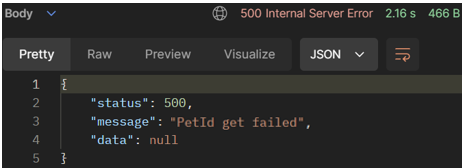
**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명** 요청한 petId의 리소스를 찾을 수 없는 경우

**텍스트, 스크린샷, 멀티미디어 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명** 요청한 petId의 insulin\_time1이 존재하지 않는 경우

서버에서 요청을 처리하는 중 오류가 발생한 경우



**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명** JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**19. 인슐린 주사시간2 delete API: 인슐린 주사시간(최대 3개) pets 테이블의 칼럼(insulin\_time2)값 삭제, 나머지 insulin\_time 저장위치 조정**

|  |  |
| --- | --- |
| method | DELETE |
| URL | http://{SERVER\_URL}/pets/{petId}/insulin-time2 |

{petId} 부분에 insulin-time2의 값을 삭제하려는 반려동물의 petId 넣기

Request Header

|  |  |
| --- | --- |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: null |

**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명**

insulin-time2의 값이 삭제되고, insulin-time3에 저장된 값이 insulin-time2에 다시 저장되고 insulin-time3는 null이 됨

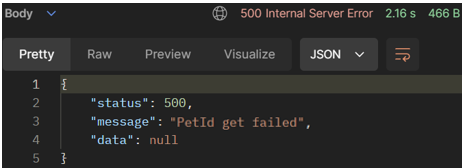
**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명** 요청한 petId의 리소스를 찾을 수 없는 경우

텍스트, 스크린샷, 멀티미디어 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명 요청한 petId의 insulin\_time2가 존재하지 않는 경우

서버에서 요청을 처리하는 중 오류가 발생한 경우



**텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명** JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우

**20. 인슐린 주사시간3 delete API: 인슐린 주사시간(최대 3개) pets 테이블의 칼럼(insulin\_time3)값 삭제, 나머지 insulin\_time 저장위치 조정**

|  |  |
| --- | --- |
| method | DELETE |
| URL | http://{SERVER\_URL}/pets/{petId}/insulin-time3 |

{petId} 부분에 insulin-time3의 값을 삭제하려는 반려동물의 petId 넣기

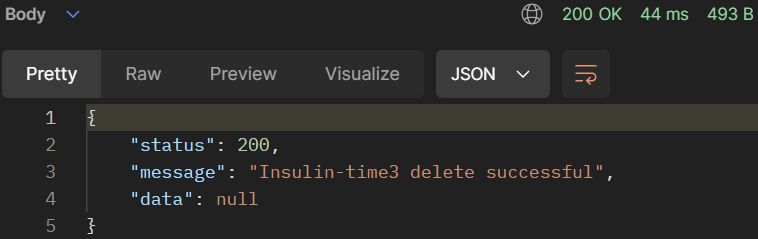
Request Header

|  |  |
| --- | --- |
| Authorization | {"Bearer " + token} |

{"Bearer " + token}은 기기에 저장되어 있는 JWT토큰(String)

Response Body

|  |
| --- |
| “status”: int  “message”: String  “data”: null |

**** insulin-time3의 값이 삭제됨(값이 null로 수정됨)

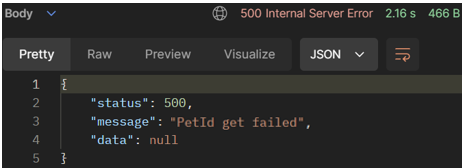
**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명** 요청한 petId의 리소스를 찾을 수 없는 경우

**텍스트, 스크린샷, 멀티미디어 소프트웨어, 소프트웨어이(가) 표시된 사진

자동 생성된 설명** 요청한 petId의 insulin\_time3가 존재하지 않는 경우

서버에서 요청을 처리하는 중 오류가 발생한 경우



**텍스트, 스크린샷, 폰트, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명** JWT토큰의 유효기간이 만료된 경우, JWT토큰이 null인 경우