

metoda	ścieżka	request body	response body	opis	uprawnienia
POST	/api/v1/login/access-token	obiekt OAuth2PasswordRequestForm (username: str password: str)	obiekt Token (access_token: str token_type: str)	zwraca token JWT używany do autoryzacji użytkownika	WSZYSCY
GET	/api/v1/users/me	N/D	obiekt User	zwraca użytkownika przypisanego do wysłanego tokenu JWT User -> (id, full_name, email, is_admin)	WSZYSCY
GET	/api/v1/users/[id]	N/D	obiekt User o podanym id	zwraca użytkownika o danym id	ADMIN
POST	/api/v1/users	obiekt UserCreateDto (email: str password: str)	nowo utworzony obiekt User	tworzy nowy obiekt User przyjmuje UserCreateDto	ADMIN
GET	/api/v1/users	N/D	listę obiektów User	zwraca listę obiektów User	ADMIN
GET	/api/v1/cars	N/D	listę obiektów Car	zwraca wszystkie obiekty Cars	WSZYSCY
POST	/api/v1/cars/query	obiekt CarsQueryCriteria	listę obiektów Car na podstawie podanych kryteriów	zwraca listę obiektów Cars na podstawie kryteriów zawartych w obiekcie CarsSearchQuery CarsSearchQuery -> (model, typ, typ paliwa, typ skrzyni biegów, typ klimatyzacji, rodzaj napędu, liczba pasażerów(zakres), cena(zakres), daty dostępności)	WSZYSCY
GET	/api/v1/cars/[id]	N/D	obiekt Car	zwraca obiekt Car o podanym id	WSZYSCY
POST	/api/v1/cars	CreateCarDto	nowo utworzony obiekt Car	tworzy nowy obiekt Car przyjmuje CreateCarDto zgodne z obiektem Car (bez ID)	ADMIN
PUT	/api/v1/cars/[id]	UpdateCarDto	zaktualizowany obiekt Car	aktualizuje obiekt Car na podstawie UpdateCarDto	ADMIN
DELETE	/api/v1/cars/[id]	N/D	N/D	usuwa obiekt Car o danym id	ADMIN
GET	/api/v1/customers	N/D	listę obiektów Customer	zwraca listę obiektów Customer	WSZYSCY
GET	/api/v1/customers/[id]	N/D	Customer o podanym id	zwraca obiekt Customer o podanym id	WSZYSCY
POST	/api/v1/customers	CreateCustomerDto	nowo utworzony obiekt Customer	tworzy nowy obiekt Customer przyjmuje CreateCustomerDto zgodne z obiektem Customer (bez ID)	WSZYSCY
PUT	/api/v1/customers/[id]	UpdateCustomerDto	zaktualizowany obiekt Customer	aktualizuje obiekt Customer na podstawie UpdateCustomerDto	WSZYSCY
DELETE	/api/v1/customers/[id]	N/D	N/D	usuwa obiekt Customer o danym id	WSZYSCY
POST	/api/v1/reservations	obiekt CreateReservationDto	nowo utworzony obiekt Reservation	tworzy nową rezerwację samochodu CreateReservationDto -> (car_id, customer_id, start_date, end_date), status = NEW	WSZYSCY
GET	/api/v1/reservations/[id]	N/D	obiekt Reservation	zwraca obiekt Reservation o danym id	WSZYSCY
GET	/api/v1/reservations	N/D	listę obiektów Reservation	zwraca listę rezerwacji	WSZYSCY
PUT	/api/v1/reservations/[id]	UpdateReservationDto	zaktualizowany obiekt Reservation	aktualizuje obiekt Reservation na podstawie UpdateReservationDto	WSZYSCY
DELETE	/api/v1/reservations/[id]	N/D	N/D	usuwa obiekt Reservation o danym id	ADMIN
POST	/api/v1/rentals	obiekt CreateRentalDto	nowo utworzony obiekt Rental	tworzy nowe wypożyczenie samochodu CreateRentalDto -> (customer_id, car_id, start_date, end_date) status = IN_PROGRESS, może przyjmować reservation_id, wtedy customer_id i car_id muszą być zgodne z rezerwacją. Ustawia wtedy status podanej rezerwacji na COLLECTED	WSZYSCY
GET	/api/v1/rentals	N/D	listę obiektów Rental	zwraca listę wynajmów	WSZYSCY
GET	/api/v1/rentals/[id]	N/D	obiekt Rental	zwraca obiekt Rental o podanym id	WSZYSCY
PUT	/api/v1/rentals/[id]	UpdateRentalDto	zaktualizowany obiekt Rental	aktualizuje obiekt Rental na podstawie UpdateRentalDto	WSZYSCY
DELETE	/api/v1/rentals/[id]	N/D	status usuwania (sukces czy błąd)	usuwa obiekt Rental o danym id	ADMIN