Название теста	Предусловие	Шаги	Ожидаемый результат	Статус
Создание нового питомца (POST /pet)	Postman открыт, выбран метод POST, заголовок Content-Type = application/json	1. Перейти в Postman.	Код ответа 200 или 201. В теле ответа — JSON объекта с полями id, name, status, совпадающими с отправленными	Pass
		2. Ввести URL: https://petstore.swagger.io/v2/pet 3. Установить метод: POST 4. В Headers указать: Content-Type = application/json 5. В Body (raw, JSON) вставить:		
		"id": 987654321,		
		"id": 1, "name": "dog"		
		"name": "Sharik", "photoUrls": [
], "tags":[
		{ "id": 1, "name": "cute"		
)], "status": "available"		
		} 6. Нажать Send		
Получение информации о питомце по ID (GET /pet/{petId})	Существующий petId	1. Ввести URL: https://petstore.swagger.io/v2/pet/987654321	Код 200. В теле ответа отображаются корректные данные питомца с id = 987654321	Pass
		2. Метод: GET		
Обиордония	Cymporthylounia polld = 007654004		Vog 070070 200 0	
Обновление данных питомца (PUT /pet)	оуществующии репа = 98/654321		код ответа 200. Ответ содержит обновленные данные питомца	Pass
		2. URL: https://petstore.swagger.io/v2/pet		
		 Header: Content-Type = application/json В body меняем имя питомца с "Sharik" на "Мухтар" 		
	Создание нового питомца (POST /pet) Получение информации о питомце по ID (GET /pet/{petId})	Получение информации о питомце по ID (GET /pet/(petid)) Существующий petid Существующий petid = 987654321	Postman открыт, выбран метод POST, aronosox Content-Type = application/json 1. Перейти в Postman. 2. Ввести URL: https://petstore.swagger.toh/2/pet 3. Установить метод: POST 4. В Headers указать: Content-Type = application/json 5. В Body (гам, JSON) вставить: { "id": 987654321, "category"; "id": 1, "name": "otog" }, "name": "Sharik", "photolife": ["id": 1, "name": "Sharik", "photolife": ["id": 1, "name": "cute" }, } . Вызать Send Существующий petid Обновление данных питомца (PUT /pet) 2. Ивтод: PUT 1. Метод: PUT 2. URL: https://petstore.swagger.toh/2/pet 3. Нажать Send 1. Метод: PUT 2. URL: https://petstore.swagger.toh/2/pet 3. Нажать Send 4. В воду концентивующий ретом (предоставной питомца (PUT /pet)) 2. URL: https://petstore.swagger.toh/2/pet 3. Header: Content-Type = application/json 4. В воду меняем мая питомца с "Sharik" на	Coagasive socore nurrowata (POST /pet) Postman orrowat, выбран метод POST, paroneox Content-Type = application/json 1. Перейги в Postman. Koag orsera 200 или 201. В теле ответа — JSON oбъекта с полями (в лагие, status proposed p

	1		5. Нажать Send		
TC-4	Удаление питомца (DELETE /pet/{petId})	Существующий petId = 987654321		Код ответа 200. Ответ содержит сообщение об успешном удалении	Pass
			2. URL: https://petstore.swagger.io/v2/pet/987654321		
			3. Нажать Send		
TC-5	Проверка, что питомец не существует после удаления	Питомец с ID 987654321 предварительно удален через метод DELETE	1. Метод: GET	Код ответа 404 Not found	Pass
	удаления		2. https://petstore.swagger.io/v2/pet/987654321		
			3. Нажать Send		
TC-6	Получение списка питомцев со статусом 'available'	Метод GET		Статус 200. Возвращается список питомцев со статусом 'available'	Pass
TC-7	Получение списка питомцев со статусом 'pending'	Метод GET		Статус 200. Возвращается список питомцев со статусом 'pending'	Pass
TC-8	Получение списка питомцев со статусом 'sold'	Метод GET		Статус 200. Возвращается список питомцев со статусом 'sold'	Pass
TC-9	Получение списка питомцев с несуществующим статусом	Метод GET		Статус 200. Возвращается пустой массив или сообщение об ошибке	Pass