		lest Case	Template for Project								
		D	D 16 1D 1								
		Project Name	Restful Broker								
		Module Name	CRUD Operation								
		Created By	Vanessa								
		Created Date	13-July-2022								
		Reviewer	Roy Dutta								
				TestCase							
TestCaseID	Scenario	METHOD	Scenario Description	Description	PreCond	TestSteps	Expected Result	Actual Result	Execution S	PostMan Scripts	Executed QA Name
TC01	PING HEALTH CHECK		Status Code	Verify if you are getting status code of 201 for ping health check		1Login to POSTMAN tool 2.ln POSTMAN tool Navigate to My workspace>Collection- >Click Addnew request 3.Enter the PingHealth check url https://restInb-booker-herokuapp.com/ping and Select GET request and Click Send button 4.Veriry'if you are getting status code of 201 in response tab Scripting: pm.test("Status code is 201", function () { pmresponse.to.have.status(201); });	1.Successful Login 2.Successful Navigation 3.The user must be able to send request 4.The Status code in the Response tab must be 201			Scripting: pm.test("Status code is 201", function () { pm.response.to.have.status(201);));	Vanessa
TC02	PING HEALTH CHECK		Response String	Verify if you are getting response body string as Created for ping health check		LLogin to POSTMAN tool 2.In POSTMAN tool, Navigate to My workspace>Collection- >Click Addnew request 3.Enter the PingHealth check url https://restIn-booker herokuapp.com/ping and Select GET request and Click Send button 4/Verify if you are getting response body string as 'Created' Scripting: pm.test("Body matches string", function () { pm.expect(pm.response.text()).to.include("Created");]);	1.Successful Login 2.Successful Navigation 3.The user must be able to send request 4.The Response body must show string as 'Created'			Scripting: pm.test(*Body matches string*, function (pm.expect(pm.response.text()).to.includ]):	ol d Vanessa
						1.Login to POSTMAN tool 2.In POSTMAN tool Navigate to My workspace->Collection- >Click Addnew request S.Enter the PingHealth check url https://restful-booker-herokuapp.com/ping and Select GET request and Click Send button 4.Verify if you are getting response body string as 'Created' Scripting pm.test("Status code is 2011, function () {				Scripting:	
TC03	PING HEALTH CHECK		Response	Verify if you are getting response as 201 Created		pm.response.to.have.status(201);)): pm.test("Body matches string" function () { pm.expect(pm.response.text()).to.include("Created");)); pm.test("Content-Type is present", function () { pm.response.to.have.header("Content-Type");)):	1.Successful Login 2.Successful Navigation 3.The user must be able to send request 4.The Response body must show string as 'Created'			printest("Satus code is 201", function () { pmr.est("Satus code is 201", function () { pmr.est("Body matches string", function () pm.est("Body matches string", function () pm.est("Content-Type is present", function () pm.response.to.have.header("Content-Type is present", func	D d
TC04	PING HEALTH CHECK		Response Header Content Type	Verify if Response Header has content type		1.Login to POSTMAN tool 2.In POSTMAN tool Navigate to My workspace->Collection- >Click Addnew request Scher the PingHealth check url https://restful-booker.herokuapp.com/ping and Select GET request and Click Send button 4.Verify if Response Header has content type Scripting: pm.test(*Content-Type is present*, function () { pm.response.to.have.header(*Content-Type*); });	1.Successful Login 2.Successful Navigation 3.The user must be able to send request 4.The Response header must have a content type			Scripting: pm.test("Content-Type is present", functi pmresponse.to.have.header("Content-));	Vanessa
				Verify if Response time		1.Login to POSTMAN tool 2.In POSTMAN tool 2.In POSTMAN tool, Import the Curl command using Raw text formal curl - inttps://restful-booker.herokuapp.com/booking 3.Click send request button and Verify if Response time in less than 200ms Scripting: pm.test("Response time is less than 200ms", function () { pm.expect(pm.response.responseTime).to.be.below(200); }};	1.Successful Login 2.Successful Navigation 3.Successful action and The Response time must be less than 200ms			Scripting: pm.test("Response time is less than 200n pm.expect(pm.response.responseTime)	

					Login to POSTMAN tool In POSTMan tool, Import the Curl command using Raw text			
					2.In POSTMan tool ,Import the Curl command using Raw text formal			
					curl -i https://restful-booker.herokuapp.com/booking			
					3.Click send request button and Verify the response body			
					Scripting: pm.test("Body matches string", function () {			
					<pre>pm.test("Body matches string", function () { pm.expect(pm.response.text()).to.include("bookingid");</pre>	1.Successful Login	Scripting:	
				Verify if User is able to));	2.Successful Navigation	pm.test("Body matches string", function ()	
TC06	Get All Booking			Fetch All bookings successfully		3.The API must fetch all bookings	pm.expect(pm.response.text()).to.includ	Vanessa
.000	Oct All BOOKING			Successiony	1.Login to POSTMAN tool		Jir.	- unc338
					2.In POSTMan tool ,Import the Curl command using Raw text			
					formal curl -i https://restful-booker.herokuapp.com/booking			
					3.Click send request button and Verify the response body			
					Scripting:	1.Successful Login		
					pm.test("Body matches string", function () {	2.Successful Navigation	Scripting:	
				Verify JSON Response	<pre>pm.expect(pm.response.text()).to.include("bookingid"); }:</pre>	3.The Response body must show list of all bookings	pm.test("Body matches string", function () pm.expect(pm.response.text()).to.includ	
TC07	Get All Booking			for Get all Booking	J.,	BOOKINGS	});	Vanessa
					1.Login to POSTMAN tool			
					2.In POSTMan tool ,Import the Curl command using Raw text formal			
					curl -i https://restful-booker.herokuapp.com/booking			
					 Click send request button and Verify the status code when all bookings are deleted 			
					-	1.Successful Login		
				Verify Get All Booking	pm.test("Status code is 404", function () { pm.response.to.have.status(404);	2.Successful Navigation 3.The Status code must be 404 for all booking	pm.test("Status code is 404", function () {	
				status code When all));	deleted	pm.response.to.have.status(404);	
TC08	Get All Booking			bookings are deleted	AL : L DOCTMANU L		3);	Vanessa
					1.Login to POSTMAN tool 2.In POSTMan tool ,Import the Curl command using Raw text			
					formal			
					curl -i https://restful-booker.herokuapp.com/booking 3.Click send request button and Verify the status code when all			
					bookings are fetched successfully			
					Scripting pm.test("Status code is 200", function () {	1.Successful Login 2.Successful Navigation	Scripting pm.test("Status code is 200", function () {	
				Verify Status code for	pm.response.to.have.status(200);	3.The Status code must be 200 for all booking	pm.response.to.have.status(200);	
TC00	Cet All Packing	GET		Get All bookings when all booking id is fetched));	deleted	3);	Vanessa
TC09	Get All Booking	GEI		an booking id is retened				Validosa
					1.Login to POSTMAN tool			
					2.In POSTMan tool ,Import the Curl command using Raw text			
					formal curl -i https://restful-booker.herokuapp.com/booking/x			
					x>x can be your preferred booking id			
					3.Click send request button and Verify if Response time in less			
					than 200ms	1.Successful Login		
						Liouccessiui Eugiii		
					Scripting:	2.Successful Navigation		
					pm.test("Response time is less than 200ms", function () {	3.Successful action and The Response time	Scripting:	
				Verify if Response time	Scripting: pm.test("Response time is less than 200ms", function () { pm.expect(pm.response.responseTime).to.be.below(200);));	Successful Navigation Successful action and The Response time must be less than 200ms	Scripting: pm.test("Response time is less than 200m pm.expect(pm.response.responseTime).	
TC10	Get Single Booking		Response time	Verify if Response time is lessthan 200ms	<pre>pm.test("Response time is less than 200ms", function () { pm.expect(pm.response.responseTime).to.be.below(200);));</pre>	3.Successful action and The Response time	pm.test("Response time is less than 200m	Vanessa
TC10	Get Single Booking		Response time	Verify if Response time is lessthan 200ms	pm.test("Response time is less than 200ms"; function () { pm.expect(pm.response.responseTime).to.be.below(200);]); 1.Login to POSTMAN tool	3.Successful action and The Response time	pm.test("Response time is less than 200m	Vanessa
TC10	Get Single Booking		Response time	Verify if Response time is lessthan 200ms	pm.test("Response time is less than 200ms', function () { pm.expect(pm.response.responseTime).to.be.below(200);)); 1Login to POSTMAN tool 2.In POSTMan tool ,Import the Curl command using Raw text formal	3.Successful action and The Response time	pm.test("Response time is less than 200m	Vanessa
TC10	Get Single Booking		Response time	Verify if Response time is lessthan 200ms	pm.test("Response time is less than 200ms", function () { pm.expect(pm.response.responseTime).to.be.below(200); }); 1.Login to POSTMAN tool 2.ln POSTMan tool ,import the Curl command using Raw text	3.Successful action and The Response time must be less than 200ms	pm.test("Response time is less than 200m	Vanessa
TC10	Get Single Booking		Response time	Verify if Response time is lessthan 200ms	pm.test("Response time is less than 200ms', function () { pm.expect(pm.response.responseTime).to.be.below(200);)); 1Login to POSTMAN tool 2.In POSTMan tool ,Import the Curl command using Raw text formal	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation	pm.test("Response time is less than 200m	Vanessa
TC10	Get Single Booking		Response time	is lessthan 200ms	pm.test("Response time is less than 200ms', function () { pm.expect(pm.response.responseTime).to.be.below(200); }); 1.Login to POSTMAN tool 2.ln POSTMan tool, import the Curl command using Raw text formal curl -i https://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which	pm.test("Response time is less than 200m	Vanessa
TC10	Get Single Booking		Response time	Verify if Response time is lessthan 200ms Verify if User is able to Fetch Single bookings	pm.test("Response time is less than 200ms', function () { pm.expect(pm.response.responseTime).to.be.below(200); }); 1.Login to POSTMAN tool 2.ln POSTMAn tool, import the Curl command using Raw text formal curl-ihttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation	pm.test("Response time is less than 200m	Vanessa Vanessa
	Get Single Booking		Response time	is lessthan 200ms Verify if User is able to	pm.test("Response time is less than 200ms", function () { pm.expect(pm.response.responseTime).to.be.below(200); }; 1Login to POSTMAN tool 2.ln POSTMAn tool, import the Curl command using Raw text formal curl -i https://restful-booker.herokuapp.com/booking/x xxcan be your preferred booking id 3.Click send request button and Verify the response body 1Login to POSTMAN tool	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which	pm.test("Response time is less than 200m	
	Get Single Booking		Response time	is lessthan 200ms Verify if User is able to	pm.test("Response time is less than 200ms', function () { pm.expect(pm.response.responseTime).to.be.below(200); }): 1.Login to POSTMAN tool 2.ln POSTMAn tool Jimport the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool Login to POSTM	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which	pm.test("Response time is less than 200m	
	Get Single Booking		Response time	is lessthan 200ms Verify if User is able to	pm.test("Response time is less than 200ms", function () { pm.expect(pm.response.responseTime).to.be.below(200); }; 1Login to POSTMAN tool 2.ln POSTMAn tool, import the Curl command using Raw text formal curl -i https://restful-booker.herokuapp.com/booking/x xxcan be your preferred booking id 3.Click send request button and Verify the response body 1Login to POSTMAN tool	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which you requested	pm.test("Response time is less than 200m	
	Get Single Booking	GET	Response time	is lessthan 200ms Verify if User is able to	pm.test("Response time is less than 200ms', function () { pm.expect(pm.response.responseTime).to.be.below(200); }); 1.Login to POSTMAN tool 2.ln POSTMAN tool /import the Curl command using Raw text formal curl-ihttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool 1.ln POSTMAN tool /import the Curl command using Raw text formal curl-ihttps://restful-booker.herokuapp.com/booking/x	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which you requested 1.Successful Login	pm.test("Response time is less than 200m	
	Get Single Booking	GET	Response time	Verify if User is able to Fetch Single bookings	pm.test("Response time is less than 200ms', function () { pm.expect(pm.response.responseTime).to.be.below(200); }); 1.Login to POSTMAN tool 2.ln POSTMAN tool /import the Curl command using Raw text formal curl-ihttps://restful-booker.herokuapp.com/booking/x x->x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool /import the Curl command using Raw text formal curl-ihttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which you requested 1.Successful Login 2.Successful Navigation 3.The Response body must show the booking 3.The Response body must show the booking	pm.test("Response time is less than 200m	
TC11	Get Single Booking	GET	Response time	Verify if User is able to Fetch Single bookings VErify JSON Response for Get Single Booking	pm.test("Response time is less than 200ms', function () { pm.expect(pm.response.responseTime).to.be.below(200); }); 1.Login to POSTMAN tool 2.ln POSTMAN tool /import the Curl command using Raw text formal curl-ihttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool 1.ln POSTMAN tool /import the Curl command using Raw text formal curl-ihttps://restful-booker.herokuapp.com/booking/x	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which you requested 1.Successful Login 2.Successful Login 2.Successful Navigation	pm.test("Response time is less than 200m	Vanessa
	Get Single Booking	GET	Response time	Verify if User is able to Fetch Single bookings	pm.test("Response time is less than 200ms', function () { pm.expect(pm.response.responseTime).to.be.below(200); }); 1.Login to POSTMAN tool 2.ln POSTMAN tool /import the Curl command using Raw text formal curl -i https://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool /import the Curl command using Raw text formal curl -i https://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which you requested 1.Successful Login 2.Successful Navigation 3.The Response body must show the booking 3.The Response body must show the booking	pm.test("Response time is less than 200m	
TC11	Get Single Booking	GET	Response time	Verify if User is able to Fetch Single bookings VErify JSON Response for Get Single Booking	pm.test("Response time is less than 200ms', function () { pm.expect(pm.response.responseTime).to.be.below(200); }): 1.Login to POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool 3.Click Send request button and Verify the response body 1.Login to POSTMAN tool 3.Click Send request button and Verify the response body 1.Login to POSTMAN tool 3.Click Send request button and Verify the response body 1.Login to POSTMAN tool 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which you requested 1.Successful Login 2.Successful Navigation 3.The Response body must show the booking 3.The Response body must show the booking	pm.test("Response time is less than 200m	Vanessa
TC11	Get Single Booking	GET	Response time	Verify if User is able to Fetch Single bookings VErify JSON Response for Get Single Booking	pm.test("Response time is less than 200ms," function () { pm.expect(pm.response.responseTime).to.be.below(200); }): 1.Login to POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool sold in the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMan tool import the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMan tool import the Curl command using Raw text formal curl-introper terms to the propose body 1.Login to POSTMAN tool 2.ln POSTMan tool import the Curl command using Raw text formal	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which you requested 1.Successful Login 2.Successful Navigation 3.The Response body must show the booking 3.The Response body must show the booking	pm.test("Response time is less than 200m	Vanessa
TC11	Get Single Booking	GET	Response time	Verify if User is able to Fetch Single bookings VErify JSON Response for Get Single Booking	pm.test("Response time is less than 200ms," function () { pm.expect(pm.response.responseTime).to.be.below(200); }): 1.Login to POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool 3.Click send request button and verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool 1.Login to POSTMAN tool 2.ln POSTMan tool, import the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMan tool, import the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which you requested 1.Successful Login 2.Successful Navigation 3.The Response body must show the booking 3.The Response body must show the booking	pm.test("Response time is less than 200m	Vanessa
TC11	Get Single Booking	GET	Response time	Verify if User is able to Fetch Single bookings VErify JSON Response for Get Single Booking	pm.test("Response time is less than 200ms," function () { pm.expect(pm.response.responseTime).to.be.below(200); }): 1.Login to POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool sold in the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMan tool import the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMan tool import the Curl command using Raw text formal curl-introper terms to the propose body 1.Login to POSTMAN tool 2.ln POSTMan tool import the Curl command using Raw text formal	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which you requested 1.Successful Login 2.Successful Navigation 3.The Response body must show the booking 3.The Response body must show the booking	pm.test("Response time is less than 200m	Vanessa
TC11	Get Single Booking	GET	Response time	Verify if User is able to Fetch Single bookings VErify JSON Response for Get Single Booking	pm.test("Response time is less than 200ms," function () { pm.expect(pm.response.responseTime).to.be.below(200); }): 1.Login to POSTMAN tool 2.ln POSTMAN tool / Import the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool / Import the Curl command using Raw text formal curl-i+ittps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool / Inport the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response code	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which you requested 1.Successful Login 2.Successful Navigation 3.The Response body must show the booking 3.The Response body must show the booking	pm.test("Response time is less than 200m	Vanessa
тс11	Get Single Booking	GET	Response time	Verify if User is able to Fetch Single bookings VErify JSON Response for Get Single Booking	pm.test("Response time is less than 200ms," function () { pm.expect(pm.response.responseTime).to.be.below(200); }): 1.Login to POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool you'restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMan tool ,Import the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMan tool ,Import the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response code Scripting:	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which you requested 1.Successful Login 2.Successful Navigation 3.The Response body must show the booking 3.The Response body must show the booking	pm.test("Response time is Jess than 200m pm.expect(pm.response.response Time).]):	Vanessa
TC11	Get Single Booking	GET	Response time	Verify if User is able to Fetch Single bookings VERIFY JSON Response for Get Single Booking Booking	pm.test("Response time is less than 200ms," function () { pm.expect(pm.response.responseTime).to.be.below(200); }): 1.Login to POSTMAN tool 2.ln POSTMAN tool / Import the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool / Import the Curl command using Raw text formal curl-i+ittps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool / Inport the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response code	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which you requested 1.Successful Login 2.Successful Navigation 3.The Response body must show the booking requested	pm.test("Response time is Jess than 200m pm.expect(pm.response.responseTime)]); Scripting: pm.test("Status code is 404", function () {	Vanessa
TC11	Get Single Booking	GET	Response time	is lessthan 200ms Verify if User is able to Fetch Single bookings VErify JSON Response for Get Single Booking Booking	pm.test("Response time is less than 200ms," function () { pm.expect(pm.response.responseTime).to.be.below(200); }): 1.Login to POSTMAN tool 2.ln POSTMAN tool / Import the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool / Import the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response body 1.Login to POSTMAN tool 2.ln POSTMAN tool 2.ln POSTMAN tool 1.ln POSTMAN tool 2.ln POSTMAN tool / Inport the Curl command using Raw text formal curl-inttps://restful-booker.herokuapp.com/booking/x x>x can be your preferred booking id 3.Click send request button and Verify the response code Scripting: pm.test("Status code is 404", function () {	3.Successful action and The Response time must be less than 200ms 1.Successful Login 2.Successful Navigation 3.The API must fetch the single booking which you requested 1.Successful Login 2.Successful Navigation 3.The Response body must show the booking requested	pm.test("Response time is less than 200m pm.expect(pm.response.response Time).]):	Vanessa

				1.Login to POSTMAN tool				
				2.In POSTMan tool ,Import the Curl command using Raw text				
				formal				
				curl -i https://restful-booker.herokuapp.com/booking/x				
				x>x can be your preferred booking id				
				3.Click send request button and Verify the response code				
				Scripting:				
				pm.test(""Status code is 200"", function () {			Scripting:	
				pm.response.to.have.status(200);	1.Successful Login		pm.test(""Status code is 200"", function ()	
			.Verify Status code for	B);	2.Successful Navigation		pm.response.to.have.status(200);	
TC14			Get Single booking		3.The Response code must be 200		3);	Vanessa
								Vanessa
				1.Login to POSTMAN tool				
				2.In POSTMan tool ,Import the Curl command using Raw text				
				format				
				https://restful-booker.herokuapp.com/booking				
				3.Click send request button and Verify the response code				
				Scripting:				
				pm.test(""""Status code is 200"""", function () {				
				pm.response.to.have.status(200);				
				J));	1.Successful Login		pm.test(""""Status code is 200"""", functio	
	L		Verify if user is able to		2.Successful Navigation		pm.response.to.have.status(200);	
TC15	Create a New resource		Create New resource		3.The Response code must be 200		3);	Vanessa
					1.Successful Login			
					2.Successful Navigation 3.The Response must be			
					3.The Response must be			
					"bookingid": 1.			
					"booking":{			
					"firstname": "Jim",			
					"lastname": "Brown",			
					"totalprice": 111,			
					"depositpaid": true,			
					"bookingdates": { "checkin": "2018-01-01".			
				1.Login to POSTMAN tool	"checkout": "2019-01-01"			
				2.In POSTMan tool ,Import the Curl command using Raw text	}.			
		POST	Verify JSON Response	format	"additionalneeds": "Breakfast"			
			for Successful POST	https://restful-booker.herokuapp.com/booking]			
TC16			Request	3.Click send request button and Verify the response	}			Vanessa
				1.Login to POSTMAN tool				
				2.In POSTMan tool ,Import the Curl command using Raw text				
				format				
				https://restful-booker.herokuapp.com/booking				
				3.Click send request button and Verify the response code Scripting:				
			Verify the status code	pm.test("""Status code is 200""", function () {	1.Successful Login		pm.test(""""Status code is 200"""", functio	
			for successful POST	pm.response.to.have.status(200);	2.Successful Navigation		pm.response.to.have.status(200);	
TC17			Request	D;	3.The Response code must be 200		3);	Vanessa
				1.Login to POSTMAN tool				
				2.In POSTMan tool ,Import the Curl command using Raw text				
			Verify if user is not able	format				
			to create new resource	https://restful-booker.herokuapp.com/booking	Expected Result:			
			using POST request	3.	1.Successful Login			
			when Request json	In request body Enter only 4 out of 8 fields	2.Successful Navigation			
TC10			body sent has only 4 out of 8 fields	Click send request button and Verify the response code	3.The Response code must be 500			V
TC18			out of 8 fields					Vanessa
				1.Login to POSTMAN tool				
			V	2.In POSTMan tool ,Import the Curl command using Raw text				
			Verify the status code of POST request when	format https://restful-booker.herokuapp.com/booking	Expected Result:			
			Request is sent with	3	1.Successful Login			
			Request json body	In request body Enter only 4 out of 8 fields	2.Successful Navigation			
			containing only 4 out of	Click send request button and Verify the response code	3.The Response code must be 500			
TC19			8 fields					Vanessa
								Vanessa
			 	 1				

			 		,		
				1.Login to POSTMAN tool			
				2.In POSTMan tool ,Import the Curl command using Raw text format			
				https://restful-booker.herokuapp.com/booking/:id			
				id=tester preference			
				curl command:			
				curl -X PUT \			
				https://restful-booker.herokuapp.com/booking/1\			
				-H 'Content-Type: application/json' \			
				-H 'Accept: application/json' \ -H 'Cookie: token=abc123' \	1.Successful Login 2.Successful Navigation		
				-ri Cookie: token=abc123 \ -d '{	3.The Response code must be 200		
				"firstname": "James",	and response body must be		
				"lastname" : "Brown",	HTTP/1.1 200 OK		
				"totalprice": 111, "depositpaid": true,	ſ		
				"bookingdates" : {	"firstname" : "James",		
				"checkin": "2018-01-01",	"lastname" : "Brown",		
				"checkout": "2019-01-01"	"totalprice": 111, "depositpaid": true,		
				"additionalneeds": "Breakfast"	"bookingdates" : {		
				3'	"checkin" : "2018-01-01",		
				3.Click send request button and Verify the response body and	"checkout" : "2019-01-01"		
	Full update of a		Verify the Response body for Successful	code	}, "additionalneeds" : "Breakfast"		
TC20	resource		PUT request		}		Vanessa
				1.Login to POSTMAN tool			
				2.In POSTMan tool ,Import the Curl command using Raw text			
				format			
				https://restful-booker.herokuapp.com/booking/:id id=tester.preference			
				·			
				curl -X PUT \			
				https://restful-booker.herokuapp.com/booking/1\ -H'Content-Type: application/json'\			
				-H 'Accept: application/json' \			
				-H'Cookie: token=abc123'\			
				-d {			
				"firstname" : "Virat", "lastname" : "Kohli",			
				"totalprice": 111,			
				"depositpaid" : true,			
				"bookingdates": { "checkin": "2018-01-01",	46 (1)		
				"checkout": "2019-01-01"	1.Successful Login 2.Successful Navigation		
				3,	3.The Response code must be 200		
				"additionalneeds" : "Breakfast"	and response header must be present		
				3'3.Click send request button and Verify the response header Scripting:	Scripting: pm.test("Content-Type is present", function () {	Scripting:	
				pm.test("Content-Type is present", function () {	pm.response.to.have.header("Content-	pm.test("Content-Type is present", function	
			Verify the Response	pm.response.to.have.header("Content-Type");	Type");	pm.response.to.have.header("Content-1	
TC04	Full update of a		header for successful	3);	3);	D;	.,
TC21	resource	-	PUT request	41i-t- POCTMANIAI			Vanessa
				1.Login to POSTMAN tool 2.In POSTMan tool ,Import the Curl command using Raw text			
				format			
				https://restful-booker.herokuapp.com/booking/:id			
				id=tester preference			
				curl -X PUT \			
				https://restful-booker.herokuapp.com/booking/1\			
				-H 'Content-Type: application/json' \			
				-H 'Accept: application/json' \ -H 'Cookie: token=abc123' \			
				-d'{			
				"firstname" : "Virat",			
				"lastname" : "Kohli", "totalprice" : 111,			
				"totalprice": 111, "depositpaid": true,			
				"bookingdates" : {			
				"checkin": "2018-01-01",			
		PUT		"checkout": "2019-01-01"			
		101		"additionalneeds": "Breakfast"			
				3'3.Click send request button and Verify the status code			
				Scripting:	J		
				pm.test("Status code for successful PUT request is 200", function	4		
				pm.response.to.have.status(200);	1.Successful Login	Scripting:	
			Verify the Status code	3):	2.Successful Navigation	pm.test("Status code for successful PUT r	
			4 1 1 1 1 1 1 1	<i>"'</i>			
TC22	Full update of a resource		for successful PUT request	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	3.The Response code must be 200	pm.response.to.have.status(200);	Vanessa

1.Login to POSTMAN tool 2.Import the curl command using raw text format curl * XPUT\ https://restful-booker.herokuapp.com/booking/1\ + H*Content-Type: application/json'\ + H*Cooker token=abc123'\ - d*{ "firstname": "James", "lastname": "Brown', "totalprice": 111, // 'depositpaid": true, "bookingdates": { "checkin": "2018-01-01", "checkin": "2018-01-01", "checkin": "2018-01-01" } ** * ** ** ** ** *	
Verify if the user is not pm.test("Status code name has string, function) {	rt request successful ser must not be able to update the pm.test("Status code name has string", fur pm.response.to.have.status("Bad Reque ponse statuscode must be 400 bad)];
1.Login to POSTMAN tool 2.Import the curl command using raw text format curl -X-PUT\ https://restful-booken-berokuapp.com/booking/1\ +H Content-Type: application/gson'\ +H Accett: application/gson'\ +H Accett: application/gson'\ +H Accett: application/gson'\ +H Accett: application/gson'\ +H Cookie: token=abc123'\ -d. { "firstname": "James", "absname": "Brown, "totalprice": 111. // depositipaid": true, "bookingdates": { "checkout": "2018-01-01", "checkout": "2018-01-01" ", "checkout": "2019-01-01"], "additionalneeds": "Breakfast") 3.Click send request button/Verify if user is not able to successfully update the booking POSTMAN Scripting: pmtest("Status code is 400", function () { pmresponse.to have status(400);); : 1.Sucessfully update that user gets for trying Pommersponse.to have status(400) { pmresponse.to have status(400) { cmresponse.to hav	pm.test("Status code is 400", function () { pm.response.to.have.status(400); }; rt request successful pr.test("Status code name has string", fur te bookingid pm.test("Status code name has string", fur pm.response.to.have.status("Bad Reque pm.response.to.have.status("Bad Reque }); Vanessa
3.The user successful 1Login to POSTMAN tool 2.Import the curl command using raw text format curl-X-PUT\ https://restrub-booker.herokuapp.com/booking/938\ -H.*Content-Type: application/json'\ -H.*Accept: application/json'\ -H.*Cocket: paplication/json'\ -H.*Cocket: paplication/json'\ -H.*Cookie: token-abc123'\ -d.*I "firstname": "QAtesterFirstnam", "astname": "QAtesterFirstnam", "booking "sylverify if user is be able to partially update resource booking 3.Veirfy if user is be able to partially update resource booking 1.	rt request successful ser must be able to do Patch update still yand response status code must be

ТС27	Partial Update		Verify the status code for successful PATCH Request	1.Login to POSTMAN tool 2.Import the curl command using raw text format curl XPUT. https://restful-booker.herokuapp.com/booking/938 \ + H-Content-Type: application/json \ + Gristmame': "QATesterFirstmam", "lastmame': "QATesterFirstmam', "lastmame': "	1. Sucessful login 2. Import request successful 3. The user must be able to do Patch update successfull and response status code must be 200 OK 4. The response body must be as below { "firstname": "'QATesterFirstName", "lastname": "'QATesterLastName", "totalprice": 0, "depositipalit" false, "bookingdates""; "checkout": "'2018-01-07", "checkout": "'2019-01-07" }, "additionalneeds": "Breakfast,Lunch, Dinner"		pm.test("Successful PATCH Request", function () [pm.expect(pm.response.code).to.be. oneOf([200, 200]);]);	Vanessa
TC28	Partial Update	ратсн	Verify if user is able to perform PATCH request successfully without authorization Verify if user is able to perform PATCH request by providing JSON request body as	1.Login to POSTMAN tool 2.Import the curl command using raw text format curl X PUT https://restful-booker.herokuapp.com/booking/938 \ -H Content T-ype: application/json \ -H Content T-ype: application/json \ -H Cooklet: bloen-abc.123 \\ -d \ "Instramme": "QATesterFirstnam", "lastname": "QATesterLastName"] 3.Navigate to authorization tab and make sure alyou dont provide credentials to authorize the Patch request bl.Leave it as blank 4.Click send request button 5.Verify if user is not able to perform PATCH request	1.Sucessful login 2.Import request successful 3.No authorization provided for the patch request 4.Successful action 5.The patch request must fail and status code must be 403			Vanessa
TC29	Partial Update		Verify is user is getting correct error code for failed PATCH request	the requirement and send the patch request { "Helloname": "QA", "middlename": "QAlast" }	1.Sucessful login 2.Import request successful 3.No authorization provided for the patch request 4.Successful action 5.The patch request must fail and status code must be 403 6.Patch request must fail 7.Status code must be 400 for incorrect syntax			Vanessa
TC31	Delete		Verify if user is able to delete all bookings	1.Login to POSTMAN tool 2.Import the curl command using raw text format curl -X DELETE* https://restful-booken-herokuapp.com/booking/2952\ -H-Content-Pipe: application/json'\ -H-Content-Pipe: application/json'\ -H-Conkie: token-abc:123' 3.Click send request 4.Verify if user is able to delete the booking successfully 5.Verify the status code	1. Successful login 2. Import request successful 3. User must be able to perform SEND rrequest 4. The user must be able detele booking id 2952 successfully if it is not previously deleted 5. The status code must be 200			Vanessa
			 İ					