Case Test Case Name	State	Description	Precondition(Given)	Design(when)	Postcondition(then)	Conclusion with pseudo expected conditions S	tatus Code given
			, ,		{"id":"1",	·	
					"employee_name":"PB" ,"employee_salary":"20003"		
		Gets the info of a single	Valid employee	When	,"employee_age":"1",		
1 DummyGetSingle	Passed	employee		GET http://dummy.restapiexample.com/api/v1/employee/1	"profile_image":""}		200
1.1 DummyGetSingle	Passed	Gets the info of a single employee	Not valid employee	When GET http://dummy.restapiexample.com/api/v1/employee/2	FALSE		200
1.1 Dulling Getoligie	1 assect	Gets the info of a all		When	TALUE		200
1 DummyGetAll	Passed	employee	None	GET http://dummy.restapiexample.com/api/v1/employees	ALL the jsons	Assertion of total length recommended	200
		Creates employee with		When			
1 DummyGetCreate	Passed	empty info	None	POST http://dummy.restapiexample.com/api/v1/create	"Integrity constraint violation: 1048 Column 'employee_name' cannot be		200
		Creates employee with				Because no other employee is named "potatoe" it passes Id field not created by POST	
2 DummyGetCreate	Passed	correct info	First time running	POST"http://dummy.restapiexample.com/api/v1/create" and body info is {"potatoe", "69420","32", id:"619"}	{"id":"74826","employee_name":"potatoe","employee_salary":"69420","e	74826Eg of generated ld	200
				When			
2.1 DummyGetCreate	Daccad	Tries to create same employee again	Test 2 DummyCetCrea	POST"http://dummy.restapiexample.com/api/v1/create" te and body info is {"potatoe", "69420", "32"}	Integrity constraint violation: 1062 Duplicate entry 'potatoe' for key 'employee name unique'}}	Makes sense to me not to allow duplicates, it passes	200
Z.1 Duniny Octoreate	1 assect	employee again	Test 2 Dunning Octorea	When	employee_name_unique ;;	makes sense to me not to allow duplicates. It passes	200
				send GET "http://dummy.restapiexample.com/api/v1/employee/74826"		Returns the info like first established. Gives confidence of the GET m	
2.2 DummyGetCreate	Bassad	Gets info of created employee "potatoe"	Toot 2.1 DummuCotCr	with body {"name": "potatoe", e "salary":"69420","age": "32", "id": "619"}	{"id":"74826","employee_name":"potatoe","employee_salary":"69420","e	Body does not affect	200
2.2 DuninyGetCreate	Passeu	employee potatoe	Test 2.1 DummyGetCr	e salary . 69420 , age . 32 , Id . 619 }	{ id . 74826 , employee_name . potatoe , employee_salary . 69420 , e	Appears to have deleted correctly	200
				When		74826Eg of generated Id	
2.3 DummyGetCreate	Passed	Deletes	Test 2.2 DummyGetCr	e send DELETE http://dummy.restapiexample.com/api/v1/delete/74826	{"success":{"text":"successfully! deleted Records"}}		200
2.4 DummvGetCreate	Desert	Confirms deletion	T+ 2 2 D	When	FALSE	Employee deleted successfully	200
2.4 DuninyGetCreate	Passeu		Test 2.3 DuningGetCre	send GET "http://dummy.restapiexample.com/api/v1/employee/74826" When	FALSE	Employee deleted successfully	200
				POST"http://dummy.restapiexample.com/api/v1/create"			
		Confirms that no id		and body info is			
		is needed to create employee	First time running	{"name": "lump",			
3 DummyGetCreate	Passed	oroate employee	st anie ranning	"age": "32"}	["id":"74935","employee_name":"lump","employee_salary":"69420","employee_salary	No issue	200
		Confirms that no id					
		is needed to create employee	Test 3 DummyGetCrea	at when		74935 Egof generated Id	
3.1 DummyGetCreate	Passed	create employee	rest 3 Dunning GetCrea	GET "http://dummy.restapiexample.com/api/v1/employee/74935"	["id":"74935","employee_name":"lump","employee_salary":"69420","emp	No issue. ID cant be created and it given by DB. makes sense to me	200
				when		,	
				POST "http://dummy.restapiexample.com/api/v1/create" with body			
	Inconclusive	Tests Ints in name		{"name": "lumpprincess404", "salary":"69420",		No issue generating the employee	
4 DummyGetCreate	IIICOIICIUSIVE	resis into in name	First time running	"age": "32"}	["id":"74963","employee_name":"lumpprincess404","employee_salary":"	Inconclusive because not sure if numbers should be accepted	200
				when			
				POST "http://dummy.restapiexample.com/api/v1/create" with body			
	Fails	Tests weird chars in name		{"name": "lump\$ëri*c/ess404", "salary": "69420",		No issue generating the employee	
5 DummyGetCreate	· diib	read word area in name	First time running	"age": "32"}	["id":"74988","employee_name":"lump\$\u00eari*c\/ess404","employee_s	weird Chars compatible with json format	200
				when			
	Passes	Tests missing salary		POST "http://dummy.restapiexample.com/api/v1/create" with body {"name": "finnthehuman"			
6 DummyGetCreate	rasses	rests missing salary	First time running	"age": "32"}	Integrity constraint violation: 1048 Column 'employee_salary' cannot be	Issue with creating with empty salary	200
, i			Ť	when	.,	• ., ,	
				POST "http://dummy.restapiexample.com/api/v1/create" with body {"name": "finnthehuman",			
	Fails	Tests missing salary		age": "32",			
6.1 DummyGetCreate		with ""	First time running	"salary": ""}	{"name":"finnthehuman","age":"32","salary":"","id":"81389"}	Should not be able to create if empty salary is submitted	200
				when			
				POST "http://dummy.restapiexample.com/api/v1/create" with body {"name": "finnthehooman"			
	Fails			"salary":"friendship",	then get the response		
7 DummyGetCreate		Tests chars in salary	First time running	"age": "17"}	{"name":"finnthehooman","salary":"friendship","age":"17","id":"75062"}	Salary should not be able to accept chars	200
				when POST "http://dummy.restapiexample.com/api/v1/create" with body			
	Fails			{"name": "iakethedog", "salary": "10000".	then		
8 DummyGetCreate		Tests chars in age	First time running	"age": "30 but in dog years"}	{"name":"jakethedog","salary":"10000","age":"30 but in dog years","id":"7	Age should not be able to accept chars	200
				when			
				POST "http://dummy.restapiexample.com/api/v1/createz" with body {"name": "PB",			
	Passes			"age": "40",			
9 DummyGetCreate		Tests wrong url	First time running	"salary": "82863"}	PAGE NOT FOUND		404
				when POST "http://dummy.restapiexample.com/api/v1/create" with body			
				{"name": "PB",			
40 0	Passes	T	El-11	"age": "40",		and the state of t	
10 DummyGetCreate		Tests wrong body	First time running	"salary": "82863"	("error":{"text":SQLSTATE[23000]: Integrity constraint violation: 1048 Co	missing } char in the end	200
				when POST "http://dummy.restapiexample.com/api/v1/create" with body			
				{"name": 73,			
11 DummyGetCreate	Fails	Tests int in name	First time running	"salary":"69420", "age": "32"}	{"name":"73","salary":"69420","age":"32","id":"76201"}	Name should not accept int	200
i i DullinyGetCreate		rooto int in nanite	i ii st uiiie Tufffillig	"age": "32"} when		reams amoutu not accept int	200
				POST "http://dummy.restapiexample.com/api/v1/create" with body			
				("name": true.			
12 DummyGetCreate	raiis	Tests name is bool	First time running	"salary":"69420", "age": "32"}	{"name":"1","salary":"69420","age":"32","id":"76229"}	Name should not accept bool	200
12 Dunning Get Greate		roote name to DUUI	. not une fulfilling	when	[name . 1 , salary . 00420 , age . 32 , lu . 10229]	Trains should not accept book	200
				POST "http://dummy.restapiexample.com/api/v1/create" with body			
				("name": 76, "salary": false,			
					I and the second		1
				"age": "32"}			ı
				"age": "32"}			
13 DummyGetCreate	Fails	Tests salary is bool	First time running	salaty : latse, "age": "32"}	{"name":"76","salary":"","age":"32","id":'76274"}	salary should not accept bool	200

Test Case	Test Case Name	State	Description	Precondition(Given)	Design(when)	Postcondition(then)	Conclusion with pseudo expected conditions	Status Code given
		Foile			when POST "http://dummy.restapiexample.com/api/v1/create" with body ("name": 77, "salary: true, "age": false)			
14	DummyGetCreate	Falls	Tests ageis bool	First time running		{"name":"77","salary":"","age":"","id":"76299"}	ageshould not accept bool	200
15	DummyGetCreate	Fails	Tests name is arrav	First time running	when POST "http://dummy.restapiexample.com/api/v1/create" with body ("name"; [79.80], "salary"; false, "age"; false)	("name":"Array","salary":","age":","id":"76299")	Name should not be array	200
		Fails	,		when "http://dummy.restapiexample.com/api/v1/create* with body ("name": ("name":80, "salary false, "age":false), "salary: false, "age":false)			
16	DummyGetCreate		Tests name is object	First time running	when	Uknown HTTP status	should recognize format and not give a 520	520
17	DummyGetCreate	Passes	Null in name	First time running	Wilei Thttp://dummy.restapiexample.com/api/v1/create* with body ("name": null, "salary": false, "salary": false)	Integrity constraint violation: 1048 Column 'employee_name' cannot be	null}	200
18	DummyGetCreate	Passes	wrong method issued	First time running	when (GET "http://dummy.restapiexample.com/api/v1/create" with body ("name": Marceline, "salary: "souls", "age": "2030")	Method not allowed. Must be one of: POST		405
19	DummyGetCreate	Passes	creates with picture profile	First time running	when POST "http://dummy.restapiexample.com/api/v1/create" with body ("name": "Marceline", "salary: "souls", "age: "1000", "inage: "https://images-wixmp-ed30a86b8c4ca887773594c2.wixmp.o		ges-wixmp-ed30a86b8c4ca887773594c2.wixmp.com\/f\/a7d01285-80	200
19.1	DummyGetCreate	Fails	verifies profile pic	Test 19 DummyGetCrea	when GET "http://dummy.restapiexample.com/api/v1/employee/79314" with t	{"id":"79314","employee_name":"Marceline","employee_salary":"0","emp	profile pic is not added	200
20	DummyGetCreate	Passes	adds new field	First time running	when POST "http://dummy.restapiexample.com/api/v1/create" with body ("name": "Rainicom", "salary": "colours", "age": "enless", "attributes": "slim"}	("name"."Rainicom", "salary"."colours", "age"."endless", "attributes". "slim"	not added	200
20.1	DummyGetCreate	Passes	checks if new field added	Test 20 DummyGetCrea	when GET "http://dummy.restapiexample.com/api/v1/employee/79402" with b	{"id":"79314","employee_name":"Marceline","employee_salary":"0","emp	not added	200
	DummyGetCreate	Passes	age is empty	First time running	when POST "http://dummy.restapiexample.com/api/v1/create" with body ("name": "corn", "salary": "yellow")	("error":("text":SQLSTATE[23000]: Integrity constraint violation: 1048 Co	makes sense to me	200
21.1	DummyGetCreate	Fails	age is ""	First time running	when POST "http://dummy.restapiexample.com/api/v1/create" with body ("name". "corn", "salary": "yellow", "age": "")	{"name"."corn","salary"."yellow","age":"-","id"."82146"}		200
1	DummyPutUpdate	Fails	employee is updated	employee already exists	When PUT"http://dummy.restapiexample.com/api/v1/update/76649" with body ("name": "Buns", "salary": 2, "age": "ripe")	{"name":"Buns","salary":"2","age":"ripe"}	age should not be a character	200
		Passes			when GET*http://dummy.restapiexample.com/api/v1/update/76649* with body			
1.1	DummyPutUpdate		employee is updated	Test 1 DUmmyPutUpda	The state of the s	{"id":"76649","employee_name":"Buns","employee_salary":"2","employe	e_age":"0","profile_image":""}	200
	DummyPutUpdate	Passes	employee is updated with negative	Eiret time running	when GET*http://dummy.restapiexample.com/api/v1/update/1" with body	{"id":"1","employee_name":"ada","employee_salary":"1236","employee_	age"-" " "profile image"-""\	200
	DummyPutUpdate	Fails	employee is updated with negative		When snd PUT with "http://dummy.restapiexample.com/api/v1/employee/1" with ("name":"PB", "salary": "science", "age":-7)	{"name"."PB", "salary". "science", "age". "-7"}	age . 0 , prome_image } Should not accept negative ints	200
	DummyPutUpdate	Fails	employee is updated with negative		when GET"http://dummy.restapiexample.com/api/v1/update/1" with body	{"name":"PB","salary":"science", "age":"-7"}	Should not accept negative ints salary should not be updated with char and this passes	200
		Passes			When soft Put with "http://dummy.restapiexample.com/api/v1/employee/1" with "name": "Bubble", "salany", "science", "agae", "math" "jrd": 69420)	("error": "text": SQLSTATE[23000]: Integrity constraint violation: 1062 Du		
2.3	DummyPutUpdate		employee is updated with new id	Test 2.3 DummyPutUpd		'employee_name_unique'}}	Should not accept negative ints	200
3	DummyPutUpdate	Passes	employee is updated with url	First time running	When PUT with "http://dummy.restapiexample.com/api/v1/employee/bacon" with ("name": "Bacon", "salary": "crispy", "age": "salt")	Method not allowed. Must be one of: GET		405
3.1	DummyPutUpdate	Passes	employee is updated with url	Test 3 DummyPutUpdat	when GET"http://dummy.restapiexample.com/api/v1/employee/bacon" with b	{"error":{"text":SQLSTATE[42S22]: Column not found: 1054 Unknown co	olumn 'bacon' in 'where clause'}	200

Case	Test Case Name	State	Description	Precondition(Given)	Design(when)	Postcondition(then)	Conclusion with pseudo expected conditions	Status Code given
					when PUT "http://dummy.restapiexample.com/api/v1/update/1" with body ["name": "PB", "salary": -80000,			
					"age": 1,			
4	DummyPutUpdate	rasses	employee is updated with profile pi	First time running	"profile_image":"https://theicekingisapatoot.files.wordpress.com/2013/	"name":"PB","salary":"1","age":"1","profile_image":"https:\/\/theicekingi	isapatoot.files.wordpress.com\/2013\/03\/a_kiss_from_princess_bubble	200
		Faile			when GET*http://dummy.restapiexample.com/api/v1/employee/1*			
4.1	DummyPutUpdate	i and	employee is updated with profile pi	Test 4 DummyPutUpdat		{"id":"1","employee_name":"PB","employee_salary":"-80000","employe	e_age":"1","profile_image":""}	200
					PUT "http://dummy.restapiexample.com/api/v1/update/1" with body {"name": "PB",			
		Enile			"salarv": 1.			
5	DummyPutUpdate	raiis	employee is updated with new colu	First time running	"age": 1, "bubble": "gum"}	{"name":"PB","salary":"1","age":"1","bubble":"gum"}	not added	200
		Passes			when GET"http://dummy.restapiexample.com/api/v1/update/1"			
5.1	DummyPutUpdate		employee is updated with new colu		t	{"id":"1","employee_name":"PB","employee_salary":"1","employee_age		200
1	DummyDeletesEmp	Fails	Deletes all employees	First time running	When DELETE with http://dummy.restapiexample.com/api/v1/delete/*	"success":{"text":"successfully! deleted Records"}}	no delete occurs	200
		Passes			when GET"http://dummy.restapiexample.com/api/v1/employees" with body			
1.1	DummyDeletesEmp	1 03303	Deletes all employees			all employees		200
1	DummyDeletesEmp	Passes	Deletes one employee	Id existence verified	When DELETE with http://dummy.restapiexample.com/api/v1/delete/79402	"success":{"text":"successfully! deleted Records"}}		200
	,	_			when	, , , , , , , , , , , , , , , , , , , ,		
1.1	DummyDeletesEmp	Passes	Deletes one employee	Test 1 DummyDeletesE	GET"http://dummy.restapiexample.com/api/v1/employee/79402" with b	FALSE		200
			 					
			 					

Test Case	Test Case Name	State	Description	Precondition(Given)	Design(when) Postcondition(then)	Conclusion with pseudo expected conditions	Status Code giv	en
								1
								1
								1
								1
								1
								+
								+
								+
								+
								+
								+
								+
								4
								4
								4
								1
								1
								1
								1
								1
								1
								1
								+
								+
								+
								+
								+
								+
								4
								4
								1
								1
								1
								1
								+
								+
								+
								+
								+
								+
								+
								4
								4
								1
								1
								1
								1
								1
								1
								+
								1
								+
							-	+
								+
								+
								4
							1	

T 0	T O N		B	D		0	0	
rest Case	rest Case Name	State	Description	Precondition(Given)	Design(when) Postcondition(then)	Conclusion with pseudo expected conditions	Status Code giv	en
								+
								+
								+
								+
								1
								1
								1
								1
								1
								1
								1
								İ
								1
								1
								1
								1
								1
								1
								1
								4
								4
								+
								-
								+
								1
								-
								+
								+
								+
								+
								+
								+
								+
								†
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
							-	

T 0	T 0 N		Description	D	Design(when) Postcondition(then)	Conclusion with pseudo expected conditions	0	
Test Case	Test Case Name	State	Description	Precondition(Given)	Design(when) Postcondition(then)	Conclusion with pseudo expected conditions	Status Code giv	en
								+
								+
								+
								+
								+
								+
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								4
								4
								+
								+
								+
								-
								+
								+
								+
								+
								+
								†
								†
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1

T	T 0 N		D	D		0	0	
rest Case	rest Case Name	State	Description	Precondition(Given)	Design(when) Postcondition(then)	Conclusion with pseudo expected conditions	Status Code giv	en
								+
								+
								+
								+
								1
								1
								1
								1
								1
								1
								İ
								1
								1
								1
								1
								1
								1
								1
								4
								+
								+
								-
								+
								1
								-
								+
								+
								+
								+
								+
								+
								+
								†
								1
								†
								†
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1

	T		D	D		0	0	
rest Case	rest Case Name	State	Description	Precondition(Given)	Design(when) Postcondition(then)	Conclusion with pseudo expected conditions	Status Code giv	en
								+
								+
								+
								+
								1
								1
								1
								1
								1
								1
								İ
								1
								1
								1
								1
								1
								1
								1
								4
								4
								+
								-
								+
								1
								-
								+
								+
								+
								+
								+
								+
								+
								†
								1
								†
								†
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1

T	T 0 N		B	D		0	0	
rest Case	rest Case Name	State	Description	Precondition(Given)	Design(when) Postcondition(then)	Conclusion with pseudo expected conditions	Status Code giv	en
								+
								+
								+
								+
								1
								1
								1
								1
								1
								1
								1
								İ
								1
								1
								1
								1
								1
								1
								4
								4
								+
								+
								+
								+
								-
								+
								+
								+
								+
								+
								+
								+
								†
								1
								1
								†
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1

T	T 0 N		B	D		0	0	
rest Case	rest Case Name	State	Description	Precondition(Given)	Design(when) Postcondition(then)	Conclusion with pseudo expected conditions	Status Code giv	en
								+
								+
								+
								+
								+
								1
								1
								1
								1
								1
								1
								1
								İ
								1
								1
								1
								1
								1
								1
								1
								4
								4
								+
								+
								+
								1
								-
								+
								+
								+
								+
								+
								+
								+
								†
								1
								1
								†
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1

	T		D	D		0	0	
rest Case	rest Case Name	State	Description	Precondition(Given)	Design(when) Postcondition(then)	Conclusion with pseudo expected conditions	Status Code giv	en
								+
								+
								+
								+
								1
								1
								1
								1
								1
								1
								1
								1
								1
								İ
								1
								1
								1
								1
								1
								1
								1
								4
								4
								+
								-
								+
								1
								-
								+
								+
								+
								+
								+
								+
								+
								†
								1
								1
								†
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1

Test Case	Test Case Name	State	Description	Precondition(Given)	Design(when) Postcondition(then)	Conclusion with pseudo expected conditions	Status Code giv	en
								1
								1
								1
								1
								4
								1
								1
								1
								1
								1
								4
								1
								1
								1
								1
								1
								1
								1
								1
								4
								1
								1
								1
								1
								1
								1
								4
								1
								1
								1
								1
								1
								1
								1
								1
								4
								1
								1
								1
								1
								1
								4
								1
								1
								1

Test Case	Test Case Name	State	Description	Precondition(Given)	Design(when) Postcondition(then)	Conclusion with pseudo expected conditions	Status Code giv	en
								1
								1
								1
								1
								4
								1
								1
								1
								1
								1
								4
								1
								1
								1
								1
								1
								1
								1
								4
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								4
								1
								4

T	T 0 N		D	D		0	0	
rest Case	rest Case Name	State	Description	Precondition(Given)	Design(when) Postcondition(then)	Conclusion with pseudo expected conditions	Status Code giv	en
								+
								+
								+
								+
								+
								1
								1
								1
								1
								1
								İ
								1
								1
								1
								1
								1
								1
								1
								4
								4
								+
								+
								+
								1
								-
								+
								+
								+
								+
								+
								+
								+
								†
								1
								t
								†
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1
								1

Test Case	Test Case Name	State	Description	Precondition(Given)	Design(when) Postcondition(then)	Conclusion with pseudo expected conditions	Status Code giv	en
								1
								1
								1
								1
								+
								1
								+
								1
								1
								†
								1
								1
								1
								+
								1
								4
								1
								1
								1
								+
								+
								+
								4
								1
								1
								1
								1
								+
								+
								+
								1
								1
								1
								1
								1
								1
								+
								1
								1
								1
								1
								+
								+
								1
								4
								1

est Case	Test Case Name	State	Description	Precondition(Given)	Design(when)	Postcondition(then)	Conclusion with pseudo expected conditions	Status Code giv	en
					• ,	, ,	·		
									-
									1
									-
									1
									1