Web API Documentation for Saxy-Repair-Tracker

http://localhost:8080/saxophone

Http Verb Method	Endpoint	<u>Description</u>
GET	/customer	READ: Gets all customers and returns them.
GET	/customer/{first_name & last_name}	READ: Gets a customer given the first and last name.
GET	/customer/{first_name}	READ: Gets a specific customer or list of customers all that match the given first name.
POST	/customer	CREATE: Creates a new customer with a given first name, last name, and phone number.
PUT	/customer	UPDATE: Goes in via with the customer primary key and updates everything else associated with a customer (first name, last name, phone number).

Http Verb Method	Endpoint	<u>Description</u>
GET	/saxophones	READ: Gets all saxophones and returns them.
GET	/saxophones/{type}	READ: Gets all saxophones that are of the entered saxophone type and returns them.
GET	/saxophones/{customer_fk}	READ: Gets the saxophones that are associated with a particular customer via the customer's foreign key.
POST	/saxophones	CREATE: Creates a brand new saxophone within the database.
PUT	/saxophones	UPDATE: Updates a saxophone via the saxophones assigned saxophone primary key.

Http Verb Method	Endpoint	<u>Description</u>
GET	/servicelineitem	READ: Gets all of the Service Line Items and returns them.
GET	/servicelineitem/{service_ticket _fk}	READ: Gets all of the Service Line Items associated with a single ticket and returns them.
POST	/servicelineitem	CREATE: Creates a Service Line Item.
PUT	/servicelineitem	UPDATE: Goes in and update a single Service Line Item via the Service Line Item's Primary key.

Http Verb Method	Endpoint	<u>Description</u>
GET	/serviceTickets	READ: Returns a Service Ticket By Status.
GET	/serviceTickets/all	READ: Returns all Service Tickets
DELETE	/serviceTickets/servicePK	DELETE: Deletes a service ticket with a service primary key given.

POST	/serviceTickets/Create	CREATE: Creates a service ticket with a given customer foregin key, description, status, estimated cost, and actual cost.
PUT	/serviceTickets	UPDATE: Goes in via with the service ticket primary key and updates everything else associated with a service ticket (customer foregin key, description, status, estimated cost, and actual cost).

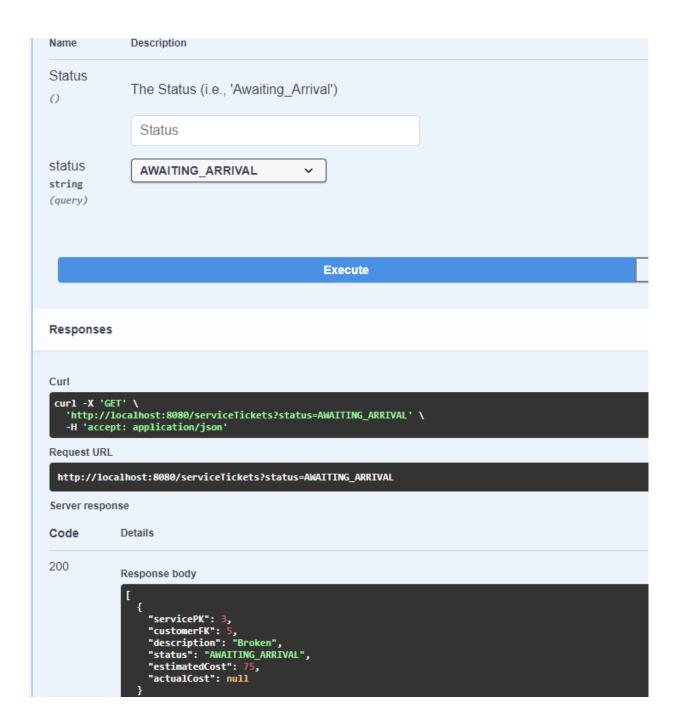
Http Verb Method	Endpoint	<u>Description</u>
GET	/serviceLineItemStatus/all	READ: Gets all of the Service Line Item Statuses and returns them.
GET	/serviceLineItemStatus	READ: Gets the Service Line Item Statuses associated with a single ticket primary key and service foregin key then returns them.
POST	/serviceLineItemStatus	CREATE: Creates a Service Line Item Status given a service

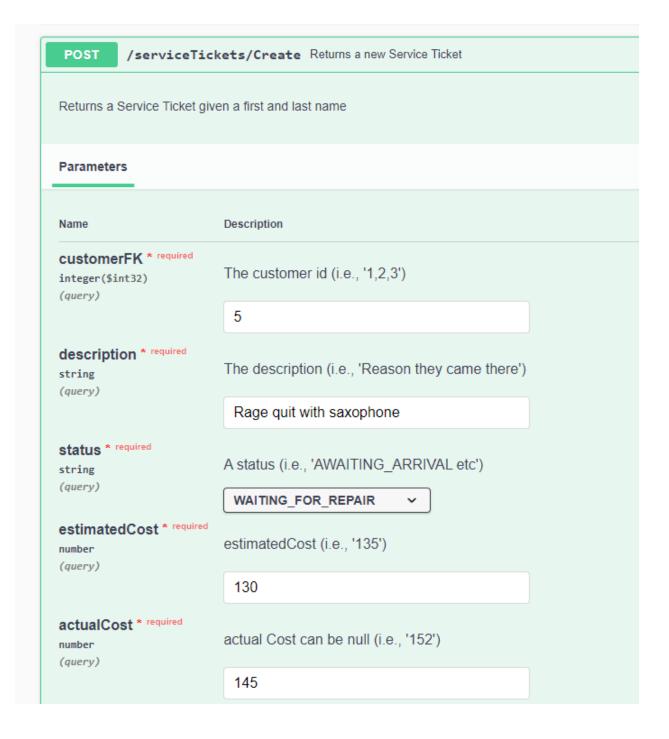
		foregin key, line item foregin key, and updates.
PUT	/serviceLineItemStatus	UPDATE: Goes in and updates a single Service Line Item Status via the Service Line Item Statuses Primary key.

Http Verb Method	<u>Endpoint</u>	<u>Description</u>
GET	/employees	READ: Returns an employee by first name and last name.
GET	/employees/all	READ: Returns all employees
DELETE	/employees/employeePK	DELETE: Deletes an employee with an employee primary key given.
POST	/employees/Create	CREATE: Creates an employee with a given first name, last name and pay rate.
PUT	/employees	UPDATE: Goes in via with the employee primary key and updates everything else associated with a service ticket

	(first name, last name and pay
	rate.)

default-service-ticket-controller GET /serviceTickets Returns a Service Ticket By Status PUT /serviceTickets Returns an updated ServiceTicket POST /serviceTickets/Create Returns a new Service Ticket GET /serviceTickets/all Returns all Service Tickets DELETE /serviceTickets/servicePK Deletes an Service Ticket





```
Code Details

Response body

{
    "servicePK": 7,
    "customerFK": 5,
    "description": "Rage quit with saxophone",
    "status": "WAITING_FOR_REPAIR",
    "estimatedCost": 130,
    "actualCost": 145
}

Response headers
```

Get all method

Response body

```
[
{
    "servicePK": 1,
    "customerFK": 4,
    "description": "things are sticking",
    "status": "WAITING_FOR_REPAIR",
    "estimatedCost": 45,
    "actualCost": null
},
{
    "servicePK": 2,
    "customerFK": 3,
    "description": "needs new pads",
    "status": "BEING_REPAIRED",
    "estimatedCost": 85,
    "actualCost": null
},
{
    "servicePK": 3,
    "customerFK": 5,
    "description": "Broken",
    "status": "AWAITING_ARRIVAL",
    "estimatedCost": 75,
    "actualCost": null
},
{
    "servicePK": 4,
    "customerFK": 6,
```

```
"description": "key fell off",
  "status": "WAITING_FOR_PARTS",
  "estimatedCost": 90,
  "actualCost": null
  "servicePK": 5,
  "customerFK": 7,
"description": "cleaning",
  "status": "PICKED_UP",
  "estimatedCost": 25,
  "actualCost": 25
},
  "servicePK": 6,
  "customerFK": 1,
"description": "key bent fell off",
  "status": "READY_FOR_PICKUP",
  "estimatedCost": 85,
  "actualCost": 115
},
  "servicePK": 7,
  "customerFK": 5,
"description": "Rage quit with saxophone",
  "status": "WAITING_FOR_REPAIR",
  "estimatedCost": 130,
  "actualCost": 145
```

GET	/serviceLineItemStatus Returns a Service Ticket Line Status
PUT	/serviceLineItemStatus Returns an updated Service Ticket Line Item Status
POST	/serviceLineItemStatus Returns a new Service Ticket Line Item Status
GET	/serviceLineItemStatus/all Returns all Service Ticket Line Statuses
GET /s	erviceLineItemStatus Returns a Service Ticket Line Status
Returns a Serv	ice Ticket Line Status given a updatePK and serviceFK
	ice Ticket Line Status given a updatePK and serviceFK
	ice Ticket Line Status given a updatePK and serviceFK
Parameters	ice Ticket Line Status given a updatePK and serviceFK Description
Parameters Name updatePK integer(\$int32	
Parameters Name	Description
Parameters Name updatePK integer(\$int32 (query) ServiceFK integer(\$int32	Description)The updatePK (i.e., '1,2,3')
Parameters Name updatePK integer(\$int32 (query)	Description)The updatePK (i.e., '1,2,3')
Parameters Name updatePK integer(\$int32 (query) ServiceFK integer(\$int32	Description The updatePK (i.e., '1,2,3') 1 The serviceFK (i.e., '1,2,3')

Parameters		
Name	Description	
serviceFK * required integer(\$int32) (query)	The serviceFK (i.e., '1,2,3')	
,,,,,	3	
lineItemFK * required integer (\$int32)	The lineItemFK (i.e., '1,2,3')	
.,,,,	3	
updates * required string (query)	Update comment (i.e., 'Completed key replacem	ent')
	Trying my best not to overload this with ss	
	Execute	

```
curl -X 'POST' \
        http://localhost:8080/serviceLineItemStatus?serviceFK=3&lineItemFK=3&updates=Trying%20my%20best%20not%
       -H 'accept: application/json' \
       -d ''
    Request URL
      http://localhost:8080/serviceLineItemStatus?serviceFK=3&lineItemFK=3&updates=Trying%20my%20best%20not%20t
    Server response
    Code
                   Details
    201
                   Response body
                      "updatePK": 3,
"serviceFK": 3,
                      "lineItemFK": 3,
"updates": "Trying my best not to overload this with ss",
"updateTime": null
                   Response headers
                      connection: keep-alive
                      content-type: application/json
date: Fri,08 Oct 2021 22:02:51 GMT
                      keep-alive: timeout=60
                      transfer-encoding: chunked
    Responses
    Code
                   Description
          /serviceLineItemStatus/all Returns all Service Ticket Line Statuses
Returns a List of Service Ticket Line Statuses
Parameters
No parameters
                                              Execute
```

Curl

Responses

200

Response body

Response headers

default-employee-controller

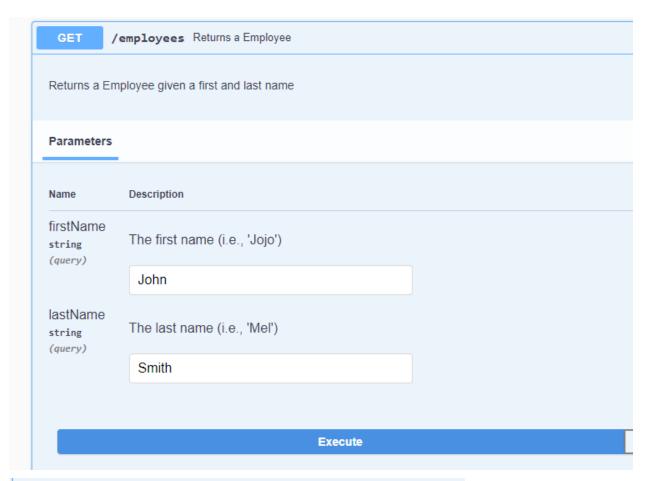
GET /employees Returns a Employee

PUT /employees Returns an updated Employee

POST /employees Returns a new Employee

GET /employees/all Returns all Employees

DELETE /employees/employeePK Deletes an Employee



```
Response body

[
{
    "employeePK": 2,
    "firstName": "John",
    "lastName": "Smith",
    "payRate": 22
    }
]

Response headers
```

```
POST
           /employees Returns a new Employee
Returns a Employee given a first and last name
Parameters
Name
                     Description
firstName * required
                     The first name (i.e., 'Jojo')
string
(query)
                       Test
lastName * required
                     The last name (i.e., 'Mel')
string
(query)
                       Testing
payRate * required
                      Pay rate (i.e., '15')
number
(query)
                       22
                                                Execute
```

```
Response body

{
    "employeePK": 8,
    "firstName": "Test",
    "lastName": "Testing",
    "payRate": 22
}

Response headers
```

GET /employees/all Returns all Employees

Returns a List of Employees

Parameters

200

Response body

```
[
    "employeePK": 1,
    "firstName": "Bavid",
    "lastName": "Bailey",
    "payRate": 35
},
    "employeePK": 2,
    "firstName": "John",
    "lastName": "Smith",
    "payRate": 22
},
    "employeePK": 3,
    "firstName": "Sam",
    "lastName": "James",
    "payRate": 21
},
    "employeePK": 4,
    "firstName": "Brent",
    "lastName": "Jamie",
    "payRate": 18
},
    "employeePK": 5,
    "firstName": "Nate",
```

```
200
```

Response body

```
"lastName": "Jamie",
    "payRate": 18
},
{
    "employeePK": 5,
    "firstName": "Nate",
    "payRate": 15
},
{
    "employeePK": 6,
    "firstName": "Alison",
    "lastName": "Abbes",
    "payRate": 25
},
{
    "employeePK": 7,
    "firstName": "Charlie",
    "lastName": "Chop",
    "payRate": 20
},
{
    "employeePK": 8,
    "firstName": "Test",
    "lastName": "Testing",
    "payRate": 22
}
]
```

Response headers

DELETE /employees/employeePK Deletes an Employee

Deletes an Employee given an id

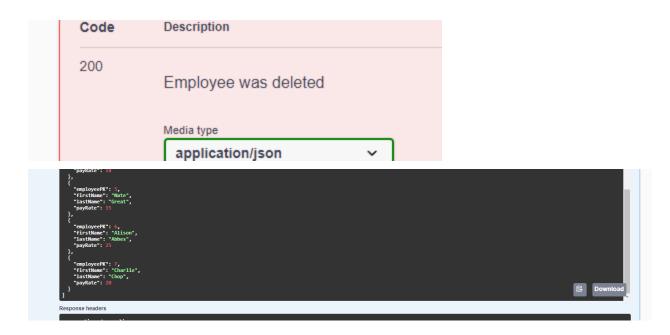
Name Description

employeePK
() employeePK (i.e., 3)

8

deleteId * required integer (\$int32) (query)

Execute



Github

https://github.com/JoleneMel/SaxySaxophoneRepairTracker