

Adam Kniffin
OSU ID#: 931492308
Email: kniffina@oregonstate.edu

CS 496 – REST Planning and Implementation

URL path to the running instance on Google App Engine: **<https://boatsnslips.appspot.com>**

Boats:

POST - /boat

- Creates a new boat that allows the parameters of 'name', 'type', and 'length' in JSON format.
- The 'id' of the boat is set on the creation of the Boat object.
- The 'at_sea' variable is set to 'True' because the upon creation the boat is still at sea and not docked in a slip.
- Sets header status to '201 Created' when successful.

GET - /boat/{boat_id}

- Where the {boat_id} is the 'id' of the boat that you want to return.
- Returns the entire Boat entity in JSON format.
- Sets header status to '200 OK' when successful.

DELETE - /boat/{boat_id}

- Where the {boat_id} is the 'id' of the boat that you want to return.
- Removes the object from the list of boats.
- Sets header status to '204 No Content' when successful.

PATCH - /boat/{boat_id}

- Where the {boat_id} is the 'id' of the boat that you want to return.
- Must provide a JSON object that contains the variables 'name', 'type', and 'length'.
 - NOTE: Length is of type integer so no quotations are necessary.
 - The format for this is

```
{
    "name": "some name",
    "type": "name1",
    "length": 100
}
```
- Sets status header of '200 OK' when successful.

PUT - /boat/{boat_id}

- Where the {boat_id} is the 'id' of the boat that you want to return.
- Must provide a JSON object that contains the variables 'name', 'type', and 'length'.

- NOTE: Length is of type integer so no quotations are necessary.
- The format for this is


```
{
        "name": "some name",
        "type": "name1",
        "length": 100
      }
```
- Sets status header of '200 OK' when successful.

Return All Instances of the class Boat

GET - /allboats

- Returns an array of objects that contains every instance of a Boat object.
- Each instance of a boat is represent as a number in the returned array.
 - I.E. array[0] is the first Boat object in the array.

Slips:

POST - /slip

- Creates a new Slip that allows the parameters of 'number' (integer), 'current_boat' (String), 'arrival_date' (string), and departure_history (is an object which holds an array of objects. The objects inside are departure_date (string), and "departed_boat" (string)) in JSON format.
 - The id and number are set at the creation of the Slip object.
 - You can pass in parameters for number, current_boat, and arrival_date and those will be saved to the Slip object.
 - Parameters can be passed to the departure_history class but will be set to 'None', or null because that will be taken care of by the BoatArrival class when a boat is sent to and from a slip.
 - The 'number' variable inside of the Slip object can be given a number.
 - If the number is already in use, a number will be generated to take its place.
 - If no number variable is given, one will be generated as well.
- The 'id' of the Slip is set on the creation of the object.
- Sets header status to '201 Created' when successful.

GET - /slip{slip_id}

- Where the {slip_id} is the 'id' of the slip that you want to return.
- Returns the entire Slip entity in JSON format.
- Sets header status to '200 OK' when successful.

DELETE - /slip/{slip_id}

- Where the {slip_id} is the 'id' of the slip that you want to return.
- Removes the Slip object from the list of slips.

- Sets status header of '204 No Content' when successful.

PATCH - /slip/{slip_id}

- Where the {slip_id} is the 'id' of the slip that you want to return.
- If you want to change the values of a certain field inside of the slip you can format the JSON to do so.
 - An Example is:


```
{
    "current_boat": "Boat Patched",
    "arrival_date": "4/30/2017"
}
```
 - You can pass any variable (or none, but that is pointless) into a PATCH request body and it will be successful.
 - You can pass in departure_history variables but, again, those will not be saved because that is for use in the BoatArrival class.
- Sets return headers to '200 OK' when successful.

PUT - /slip/{slip_id}

- Where the {slip_id} is the 'id' of the slip that you want to return.
- If you want to change the values of a certain field inside of the slip you can format the JSON to do so.
 - An Example is:


```
{
    "current_boat": "Boat Patched",
    "arrival_date": "4/30/2017"
}
```
 - You can pass any variable (or none, but that is pointless) into a PUT request body and it will be successful.
 - You can pass in departure_history variables but, again, those will not be saved because that is for use in the BoatArrival class.
- Sets return headers to '200 OK' when successful.

Return All Instances of the class Slip

GET - /allslips

- Returns an array of objects that contains every instance of a Slip object.
- Each instance of a Slip is represent as a number in the returned array.
 - I.E. array[0] is the Slip Boat object in the array.

BoatArrival Class

PUT - /boatarrival/boat/{boat_id}

- Where the boat_id is equal to the boat's 'id' variable that you want to pass into the API call.

- This is used to ADD the boat to the given slip and set it as the 'current_boat'. The boat id represents the boat that will be put in the slip.
 - The number that you want to put the boat into will be passed in with the body. This is the slip number.
 - Example:


```
{
    "number": 1
}
```
- Sets status headers to '200 OK' upon successful completion
 - Will also return the entire boat object in JSON format (with new values that have been added / edited)

PATCH - /boatarrival/boat/{ {boat_id} }

- Where the boat_id is equal to the boat's 'id' variable that you want to pass into the API call.
- This is used to change the 'departure_history' variable of the slip that was specified in the body.
 - The number that you want to put the boat into will be passed in with the body.
 - Example:


```
{
    "number": 1
}
```
 - This will put the boat with the id given into the 'departed_boat' variable and the 'departure_date' variable will be saved (dynamically allocated).
- Sets status headers to '200 OK' upon successful completion
 - Will also return the entire boat object in JSON format (with new values that have been added / edited)