

Conner Pappas, CS493: Assignment 4 - Build an Intermediate Restful API

URL: <https://cpcs493intermediateapi.appspot.com>

GET /ships

Description: Get all ships stored in the datastore. The response is an object containing an array of items, and a next link if there are more than 3 ships.

Example response:

Response code: 200

```
{
  "items": [
    {
      "name": "Test Ship 4",
      "type": "Ship 4",
      "length": 40,
      "id": "5143325141106688",
      "self": "https://cpcs493intermediateapi.appspot.com/ships/5143325141106688",
      "cargo": null
    },
    {
      "name": "Test Ship 2",
      "type": "Ship 2",
      "length": 20,
      "id": "5676533754626048",
      "self": "https://cpcs493intermediateapi.appspot.com/ships/5676533754626048",
      "cargo": null
    },
    {
      "name": "Wiggly Woo Ha",
      "type": "Slippery ship",
      "length": 42,
      "id": "5741549795147776",
      "self": "https://cpcs493intermediateapi.appspot.com/ships/5741549795147776",
      "cargo": [
        {
          "id": "5145373974724608",
          "self": "https://cpcs493intermediateapi.appspot.com/cargo/5145373974724608"
        },
        {
          "id": "5179846254657536",
          "self": "https://cpcs493intermediateapi.appspot.com/cargo/5179846254657536"
        }
      ]
    }
  ],
  "next": "http://cpcs493intermediateapi.appspot.com/ships?cursor=CjQSLmoYbX5jcGNzNDkzaW50ZXJtZWRpYXRiYXBpchlLEgVTSElQUxiAglDgi72ZCgwYACAA"
}
```

GET /ships/{shipID}

Description: Get a response containing a one element array containing an object that contains the properties of the requested shipID in the URL. Cargo property will be null if there are no cargo items loaded on it.

Example response:

Response code: 200

```
[
  {
    "name": "Wiggly Woo Ha",
    "type": "Slippery ship",
    "length": 42,
    "id": "5741549795147776",
    "self": "https://cpcs493intermediateapi.appspot.com/ships/5741549795147776",
    "cargo": [
      {
        "id": "5145373974724608",
        "self": "https://cpcs493intermediateapi.appspot.com/cargo/5145373974724608"
      },
      {
        "id": "5179846254657536",
        "self": "https://cpcs493intermediateapi.appspot.com/cargo/5179846254657536"
      }
    ]
  }
]
```

POST /ships

Description: Create a new ship containing the properties specified in the request body. The properties that can be specified by the user are the name of the ship, the type of ship, and the length of the ship. The response is an object containing the ID of the new ship.

Example request:

```
{
  "name": "Test Ship 1",
  "type": "Ship 1",
  "length": 10
}
```

Example response:

```
Response code: 200
{"id": 5959639107633152}
```

PATCH /ships/{shipID}

Description: Modify a ship, specified in the URL, with the properties specified in the request body. The response is a one element array containing an object with the ship's modified properties.

Example request:

```
{
  "length": 42,
  "name": "Wiggly Woo Ha",
  "type": "Slippery ship"
}
```

Example response:

```
Response code: 200
[
  {
    "name": "Wiggly Woo Ha",
    "type": "Slippery ship",
    "length": 42,
    "id": "5741549795147776",
    "self": "https://cpcs493intermediateapi.appspot.com/ships/5741549795147776",
    "cargo": null
  }
]
```

GET /cargo

Description: Get all cargo stored in the datastore. The response is an object containing an array of items, and a next link if there are more than 3 cargo items.

Example response:

```
Response code: 200
{
  "items": [
    {
      "weight": 10,
      "content": "HRMs",
      "delivery_date": "12/18/54",
      "id": "5145373974724608",

```

```

    "self": "https://cpcs493intermediateapi.appspot.com/cargo/5145373974724608",
    "carrier": null
  },
  {
    "weight": 200,
    "content": "Pumpkins",
    "delivery_date": "10/31/19",
    "id": "5179846254657536",
    "self": "https://cpcs493intermediateapi.appspot.com/cargo/5179846254657536",
    "carrier": null
  },
  {
    "weight": 420,
    "content": "Unironically cargo",
    "delivery_date": "04/01/69",
    "id": "5655266787852288",
    "self": "https://cpcs493intermediateapi.appspot.com/cargo/5655266787852288",
    "carrier": null
  }
],
"next":
"http://cpcs493intermediateapi.appspot.com/cargo?cursor=CjQSLmoYbX5jcGNzNDkzaW50ZXJtZWRpYXRiYXBpchlLEgVDQVJHTxiAgIDA9u2FCgwYACAA"
}

```

GET /cargo/{cargoID}

Description: Get a response containing a one element array containing an object that contains the properties of the requested cargoID in the URL.

Example response:

```

Response code: 200
[
  {
    "weight": 10,
    "content": "HRMs",
    "delivery_date": "12/18/54",
    "id": "5145373974724608",
    "self": "https://cpcs493intermediateapi.appspot.com/cargo/5145373974724608",
    "carrier": null
  }
]

```

POST /cargo

Description: Create a cargo item containing the properties specified in the request body. The response is an object containing the ID of the new cargo.

Example request:

```
{
  "content": "Pumpkins",
  "delivery_date": "10/31/19",
  "weight": 200
}
```

Example response:

```
Response code: 200
{"id": 5660980839186432}
```

PATCH /cargo/{cargoID}

Description: Modify a cargo item, specified in the URL, with the properties specified in the request body. This cannot change any information about the carrier. The response is a one element array containing an object with the slip's modified properties.

Example request:

```
{
  "weight": 5000,
  "content": "Wiggles",
  "delivery_date": "06/30/20"
}
```

Example response:

```
Response code: 200
[
  {
    "weight": 5000,
    "content": "Wiggles",
    "delivery_date": "06/30/20",
    "id": "5145373974724608",
    "self": "https://cpcs493intermediateapi.appspot.com/cargo/5145373974724608",
    "carrier": null
  }
]
```

GET /ships/{shipID}/cargo

Description: Get a response containing an object with an items property that is an array of cargo on the ship specified by shipID. If there are more than 3 items then the results are paginated.

Example response:

Response code: 200

```
{
  "items": [
    {
      "weight": 5000,
      "content": "Wiggles",
      "delivery_date": "06/30/20",
      "id": "5145373974724608",
      "self": "https://cpcs493intermediateapi.appspot.com/cargo/5145373974724608",
      "carrier": {
        "id": "5741549795147776",
        "name": "Wiggly Woo Ha",
        "self": "https://cpcs493intermediateapi.appspot.com/ships/5741549795147776"
      }
    },
    {
      "weight": 420,
      "content": "Unironically cargo",
      "delivery_date": "04/01/69",
      "id": "5655266787852288",
      "self": "https://cpcs493intermediateapi.appspot.com/cargo/5655266787852288",
      "carrier": {
        "id": "5741549795147776",
        "name": "Wiggly Woo Ha",
        "self": "https://cpcs493intermediateapi.appspot.com/ships/5741549795147776"
      }
    },
    {
      "weight": 3000,
      "content": "Ironically ships",
      "delivery_date": "08/09/45",
      "id": "5741031244955648",
      "self": "https://cpcs493intermediateapi.appspot.com/cargo/5741031244955648",
      "carrier": {
        "id": "5741549795147776",
        "name": "Wiggly Woo Ha",
        "self": "https://cpcs493intermediateapi.appspot.com/ships/5741549795147776"
      }
    }
  ], "next":
  "http://cpcs493intermediateapi.appspot.com/ships/5741549795147776/cargo?cursor=CjgSMmoYbX5jcGNzNDkzaW50ZXJtZWRpYXRiYXBpYXNleGltSElQ0FSR08YglCAwPb7IAoMGAAgAA=="
}
```

PUT ships/{shipID}/cargo/{cargoID}

Description: Load cargo, specified by cargoID, onto a ship, specified by shipID. The response is a one element array containing an object that is the ship that the cargo was loaded onto. If the cargo is on a ship and an attempt is made to load it on another ship then the server will respond with a 403.

Example response:

Response code: 200

```
[
  {
    "name": "Wiggly Woo Ha",
    "type": "Slippery ship",
    "length": 42,
    "id": "5741549795147776",
    "self": "https://cpcs493intermediateapi.appspot.com/ships/5741549795147776",
    "cargo": [
      {
        "id": "5145373974724608",
        "self": "https://cpcs493intermediateapi.appspot.com/cargo/5145373974724608"
      }
    ]
  }
]
```

DELETE ships/{shipID}/cargo/{cargoID}

Description: Unload cargo, specified by cargoID, from a ship, specified by shipID. If that cargo item wasn't on that ship then a 403 response is received.

Example response:

Response code: 200 (upon success)

Response code: 403 (upon failure)

DELETE ships/{shipID}

Descriptions: Delete a ship, specified by shipID, from the datastore. Deleting a ship also unloads any cargo that was on it.

Example response:

Response code: 200

DELETE cargo/{cargoID}

Descriptions: Delete an item of cargo, specified by cargoID, from the datastore. Deleting any cargo also unloads it from a ship it was on.

Example response:

Response code: 200