

First Resonance Coding Challenge Design Doc

After activating the server:

There are three in-built data types – Cats, Dogs, Humans, with 2 of each in the dataset

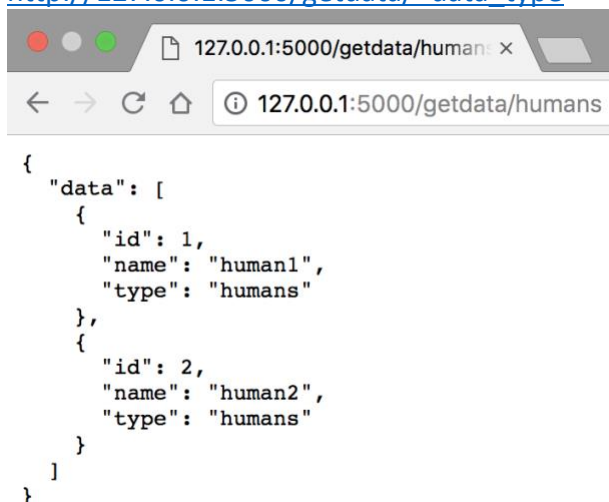
The entire data can be viewed by:

<http://127.0.0.1:5000/getdata>



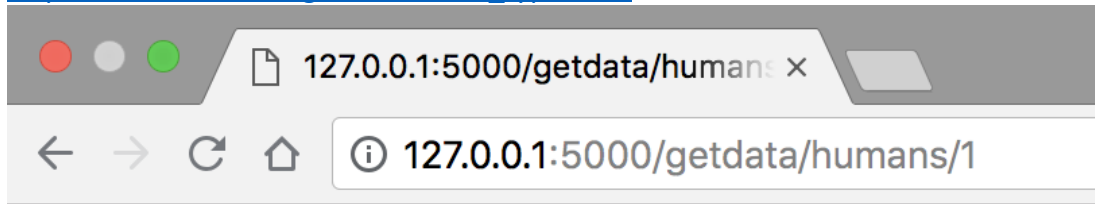
To get specific data based on type:

http://127.0.0.1:5000/getdata/<data_type>



To get specific id from the data type:

http://127.0.0.1:5000/getdata/<data_type>/<id>



```
{
  "data": [
    {
      "id": 1,
      "name": "human1",
      "type": "humans"
    }
  ]
}
```

To add data to the database:

http://127.0.0.1:5000/postdata/<data_type>

The id is set automatically and the parameter is 'name'

```
➔ curl -i -H "Content-Type: application/json" -X POST -d '{"name":"human3"}' http://localhost:5000/postdata/humans
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length: 549
Server: Werkzeug/0.14.1 Python/3.7.0
Date: Thu, 13 Sep 2018 21:38:40 GMT

{
  "data": [
    {
      "id": 1,
      "name": "cat1",
      "type": "cats"
    },
    {
      "id": 2,
      "name": "cat2",
      "type": "cats"
    },
    {
      "id": 1,
      "name": "dog1",
      "type": "dogs"
    },
    {
      "id": 2,
      "name": "dog2",
      "type": "dogs"
    },
    {
      "id": 1,
      "name": "human1",
      "type": "humans"
    },
    {
      "id": 2,
      "name": "human2",
      "type": "humans"
    },
    {
      "id": 3,
      "name": "human3",
      "type": "humans"
    }
  ]
}

➔ curl -i -H "Content-Type: application/json" -X POST -d '{"name":"cat3"}' http://localhost:5000/postdata/cats
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length: 622
Server: Werkzeug/0.14.1 Python/3.7.0
Date: Thu, 13 Sep 2018 21:38:57 GMT

{
  "data": [
    {
      "id": 1,
      "name": "cat1",
      "type": "cats"
    },
    {
      "id": 2,
      "name": "cat2",
      "type": "cats"
    },
    {
      "id": 1,
      "name": "dog1",
      "type": "dogs"
    },
    {
      "id": 2,
      "name": "dog2",
      "type": "dogs"
    },
    {
      "id": 1,
      "name": "human1",
      "type": "humans"
    },
    {
      "id": 2,
      "name": "human2",
      "type": "humans"
    },
    {
      "id": 3,
      "name": "human3",
      "type": "humans"
    },
    {
      "id": 3,
      "name": "cat3",
      "type": "cats"
    }
  ]
}
```

To modify data in the database:

http://127.0.0.1:5000/modifydata/<data_type>

Only the cat and dog classes are modifiable

Parameters are id and name, although name is not a requirement, and will create an empty field for name

```
➔ ~ curl -i -H "Content-Type: application/json" -X PUT -d '{"id": "3", "name": "catsdkjfh1"}' http://localhost:5000/modifydata/humans
HTTP/1.0 400 BAD REQUEST
Content-Type: text/html
Content-Length: 140
Server: Werkzeug/0.14.1 Python/3.7.0
Date: Thu, 13 Sep 2018 21:35:22 GMT

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<title>400 Bad Request</title>
<h1>Bad Request</h1>
<p>humans is not modifiable</p>
➔ ~ curl -i -H "Content-Type: application/json" -X PUT -d '{"id": "3", "name": "catsdkjfh1"}' http://localhost:5000/modifydata/cats
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length: 628
Server: Werkzeug/0.14.1 Python/3.7.0
Date: Thu, 13 Sep 2018 21:35:35 GMT

{
  "data": [
    {
      "id": 1,
      "name": "cat1",
      "type": "cats"
    },
    {
      "id": 2,
      "name": "cat2",
      "type": "cats"
    },
    {
      "id": 1,
      "name": "dog1",
      "type": "dogs"
    },
    {
      "id": 2,
      "name": "dog2",
      "type": "dogs"
    },
    {
      "id": 1,
      "name": "human1",
      "type": "humans"
    },
    {
      "id": 2,
      "name": "human2",
      "type": "humans"
    },
    {
      "id": 3,
      "name": "human3",
      "type": "humans"
    },
    {
      "id": 3,
      "name": "catsdkjfh1",
      "type": "cats"
    }
  ]
}
```

To delete data in the database:

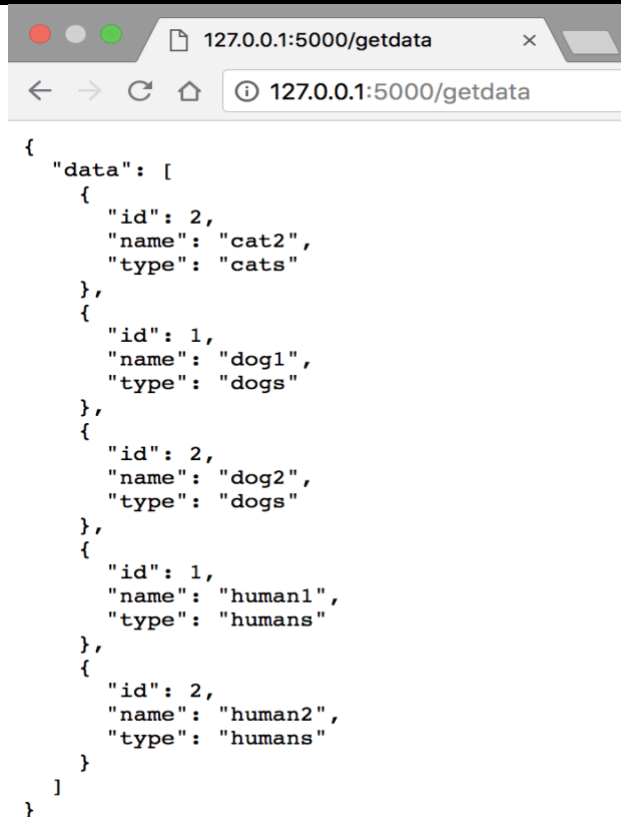
http://127.0.0.1:5000/deletedata/<data_type>

Only the cat class is deletable

There was a slight error in the delete class which makes it off by one, since it uses the 'id' passed as the parameter as an index rather than a value.

```
➔ ~ curl -i -H "Content-Type: application/json" -X DELETE -d '{"id": "0"}' http://localhost:5000/deletedata/cats
HTTP/1.0 200 OK
Content-Type: application/json
Content-Length: 21
Server: Werkzeug/0.14.1 Python/3.7.0
Date: Thu, 13 Sep 2018 21:49:20 GMT

{
  "result": true
}
```



```
{
  "data": [
    {
      "id": 2,
      "name": "cat2",
      "type": "cats"
    },
    {
      "id": 1,
      "name": "dog1",
      "type": "dogs"
    },
    {
      "id": 2,
      "name": "dog2",
      "type": "dogs"
    },
    {
      "id": 1,
      "name": "human1",
      "type": "humans"
    },
    {
      "id": 2,
      "name": "human2",
      "type": "humans"
    }
  ]
}
```