

UPDATED API

"""

/line GET request will return lines in JSON format back to the user

Args:

- line_id=<int> : filters results by id given (Optional)
- name=<string> : filters results by searching for substring (Optional)

Returns:

- JSONObject: {"result": [Line...]}
- Line Object: {"id": <int>,
 "name": <string>}

"""

"""

/line PUT request will write or edit line objects into the database

Content-type: application/json

Payload:

- JSONArray: [Line...]
- Line Object: {"id": <int>, (Optional: if given, overrides set in database. If excluded, creates new line)
 "name": <string>,
 "password": <string>}

"""

"""

/line DELETE request will delete only 1 line in the database

Content-type: application/json

Payload:

- JSONObject: {"line_id": <int>,
 "password": <string>}

"""

"""

/trap GET request will return traps in JSON format back to the user

Args (Optional):

- line_id=<int> : filters results by line id given
- trap_id=<int> : filters results by trap id given

Returns:

- JSONObject: {'result': [Trap...]}
- Trap Object: {'id': <int>,
 'rebait_time': <long int>,
 'lat': <float>,
 'long': <float>,
 'line_id': <int>,
 'line_order': <int>,
 'path_side': <int>,
 'broken': <boolean>,
 'moved': <boolean>}

"""

"""

/trap PUT request will write or edit trap objects into the database

Content-type: application/json

Payload:

```
- JSONObject: {"line_id": <int>
               "password": <string>,
               "traps": [Trap...]}

- Trap Object: {'id': <int>, (Optional: if given, overrides set in database. If
excluded, creates new line)
               're bait_time': <long int>,
               'lat': <float>,
               'long': <float>,
               'line_id': <int>,
               'line_order': <int>,
               'path_side': <int>,
               'broken': <boolean>, (Optional on editing set, don't include if
creating new trap)
               'moved': <boolean> (Optional on editing set, don't include if
creating new trap)}
```

"""

"""

/trap DELETE request will delete multiple traps in the database belonging to one line

Content-type: application/json

Payload:

```
- JSONObject: {"line_number": <int>,
               "password": <string>,
               "traps": [<int>...]}
```

"""

"""

/catch GET request will return catches in JSON format back to the user

Args:

- line_id=<int> : filters results by line id given by relationship query to a trap (Optional)
- trap_id=<int> : filters results by trap id given (Optional)

Returned:

```
JSONObject: {'result': [catch...]}
Catch Object: {'id': <int>,
               'trap_id': <int>,
               'animal_id': <int>,
               'time': <long int>}
```

"""

"""

/catch PUT request will write or edit catch objects into the database

Content-type: application/json

Payload:

```
JSONObject: {"line_id": <int>
               "password": <string>,
               "catches": [Catch...]}
```

```
    Catch Object: {'id': <int>, (Optional: if given, overrides set in database. If
excluded, creates new line)
        'trap_id': <int>,
        'animal_id': <int>,
        'time': <long int> (Ignored when overriding set in database)}
"""
```

REST API

	Params	Input/Result (example)
GET /line Returns a list of lines.	Optional: line_id name	{“result”: [{“id”:0, “name”: “ex0”}]}
PUT /line Create a line or list of lines. If “id” is given, it will overwrite any existing line with that id. Can be used to edit lines.	Required: pass	[{“password”: “expass”, “name”: “ex”}] {“result”: [{“id”: 3, “name”: “ex”}]}
[[POST /line]] Not needed? Edit a line	Required: pass line_id	{“name”: “ex”} {“result”: [{“id”: 3, “name”: “ex”}]}
GET /trap Returns a list of traps.	Optional: line_id trap_id	{“result”: [{“id”: 0, “line_id”: 0, “re bait_time”: 1471314737, “lat”: 123.5, “long”: 635.7, “line_order”: 5, “path_side”: -1, “broken”: false, “moved”: false}]}
PUT /trap Create a trap or list of traps. If “id” is given in a trap it will overwrite any existing trap with that id. Can therefore be used to edit traps.	Required: pass	[{“line_id”: 0, “lat”: 123.5, “long”: 635.7, “line_order”: 5, “path_side”: -1, “re bait_time”: 1471314737}] {“result”: [{“id”: 0, “line_id”: 0, “re bait_time”: 1471314737, “lat”: 123.5, “long”: 635.7, “line_order”: 5, “path_side”: -1, “broken”: false, “moved”: false}]}
[[POST /trap]] Not needed? Edit a trap.	Required: pass trap_id	{“lat”: 432.6} {“result”: [{“id”: 0, “line_id”: 0, “re bait_time”: 1471314737, “lat”: 123.5, “long”: 635.7, “line_order”: 5, “path_side”: -1, “broken”: false, “moved”: false}]}
GET /catch Returns a list of catches.	Optional: line_id	{“result”: [{“id”: 0, “trap_id”: 0, “animal_id”: 0, “time”:

	trap_id catch_id animal_id min_time max_time	1471314737, "image_id": 0}}}
PUT /catch Create a catch or list of catches. If "id" is given in a catch it will overwrite any existing catch with that id. Can therefore be used to edit catches.	Required: pass	{"trap_id": 0, "animal_id": 0, "time": 1471314737} {"result": [{"id": 0, "trap_id": 0, "animal_id": 0, "time": 1471314737}]}
[[POST /catch]] Edit a catch	Required: pass catch_id	{"animal_id": 3} {"result": [{"id": 0, "trap_id": 0, "animal_id": 3, "time": 1471314737}]}