Database Query Specifications

LutzenH (Bloomdex.org)

Contents

1. Important Information

- All of the following queries require HTTP-basic-authentication in order to work. If a query needs authorities other than *ROLE_USER* it will be mentioned in the specification of the given query. User passwords are encrypted using BCrypt's password-encoder.

- If the title of a specification contains '(paged)' it means that this query is similar to another query by the same name but split into pages. In order to browse through these pages the parameter keys *page* and *size* can be used. Paged queries have a slightly different URL than their counterpart.

- **Chapter 4. Machine Queries** are queries only used by machines and can only be run by users with the authority level *ROLE_ROBOT*

- Bare in mind that **Cross-Origin Resource Sharing** has been configured to work in any circumstance. So it does not matter what origin the API-request comes from.

- If an `application/xml` should be returned instead of an `application/json` add the following http-header to your request: `Accept: application/xml`

2. Weather-Stations

2.1. Retrieving a weather-station by ID

Input

```
GET /api/v1/stations/{station_id}/
```

Key	Explanation	Required	Type
station_id	The ID of the station you want to retrieve information from.	Yes	integer

Output

```json
{
    "id": 727260,
    "name": "PARENT (MARS)",
    "country": "CANADA",
    "latitude": 47.917,
    "longitude": -74.617,
    "elevation": 439.0
}
```

2.2. Retrieving a list of weather-station by coordinates

WARNING: The item-count of this query can be greater than 5000 results! Use with caution!

Input

```
GET /api/v1/stations
    ?min_latitude={min_latitude}
    &max_latitude={max_latitude}
    &min_longitude={min_longitude}
    &max_longitude={max_longitude}
```

Key	Explanation	Required	Type
min_latitude	The latitude of the top-left-position of your map.	Yes	double
max_latitude	The latitude of the bottom-right-position of your map.	Yes	double
min_longitude	The longitude of the top-left-position of your map.	Yes	double
max_longitude	The longitude of the bottom-right-position of your map.	Yes	double

Output

```json
[
    {
        "id": 506390,
        "name": "DATSAV3 VIRT STATION (CHILE)",
        "country": "CHILE",
        "latitude": -26.317,
        "longitude": -70.617,
        "elevation": 0.0
    },
    {
        "id": 749027,
        "name": "LA CHORRERA  PANAMA",
        "country": "PANAMA",
        "latitude": 8.85,
        "longitude": -79.783,
        "elevation": 94.0
    },
    {
        "id": 749028,
        "name": "PACORA  PANAMA LAJOYA",
        "country": "PANAMA",
        "latitude": 9.133,
        "longitude": -79.233,
        "elevation": 48.0
    },
    {
        "id": 749035,
        "name": "AGUADULCE  PANAMA",
        "country": "PANAMA",
        "latitude": 8.25,
        "longitude": -80.566,
        "elevation": 46.0
    }
]
```

2.3. Retrieving a list of weather-station by coordinates (paged)

Input

```
GET /api/v1/paged/stations
    ?min_latitude={min_latitude}
    &max_latitude={max_latitude}
    &min_longitude={min_longitude}
    &max_longitude={max_longitude}
    [&page={page}]
    [&size={size}]
```

Key	Explanation	Required	Type
min_latitude	The latitude of the top-left-position of your map.	Yes	double
max_latitude	The latitude of the bottom-right-position of your map.	Yes	double
min_longitude	The longitude of the top-left-position of your map.	Yes	double
max_longitude	The longitude of the bottom-right-position of your map.	Yes	double
page	The current page you are on.	No	integer
size	The size of the pages.	No	integer

Output

```
{
    "content": [
        {
            "id": 518950,
            "name": "DATSAV3 VIRT STATION (CHILE)",
            "country": "CHILE",
            "latitude": -26.317,
            "longitude": -70.617,
            "elevation": 0.0
        },
        {
            "id": 525460,
            "name": "DATSAV3 VIRT STATION (CHILE)",
            "country": "CHILE",
            "latitude": -26.317,
            "longitude": -70.617,
            "elevation": 0.0
        },
        {
            "id": 526790,
            "name": "DATSAV3 VIRT STATION (CHILE)",
            "country": "CHILE",
            "latitude": -26.317,
            "longitude": -70.617,
            "elevation": 0.0
        },
        {
            "id": 534200,
            "name": "DATSAV3 VIRT STATION (CHILE)",
            "country": "CHILE",
            "latitude": -26.317,
            "longitude": -70.617,
            "elevation": 0.0
        },
        {
```

```json
            "id": 535460,
            "name": "DATSAV3 VIRT STATION (CHILE)",
            "country": "CHILE",
            "latitude": -26.317,
            "longitude": -70.617,
            "elevation": 0.0
        }
    ],
    "pageable": {
        "sort": {
            "sorted": false,
            "unsorted": true,
            "empty": true
        },
        "pageNumber": 1,
        "pageSize": 5,
        "offset": 5,
        "paged": true,
        "unpaged": false
    },
    "totalPages": 41,
    "totalElements": 203,
    "last": false,
    "first": false,
    "sort": {
        "sorted": false,
        "unsorted": true,
        "empty": true
    },
    "size": 5,
    "number": 1,
    "numberOfElements": 5,
    "empty": false
}
```

2.4. Retrieving a list of weather-station by radius

Input

Either `radius_km` or `radius_deg` should be used, not both!

```
GET /api/v1/stations/radius
    ?latitude={latitude}
    &longitude={longitude}
    [&radius_km={radius_km}]
    [&radius_deg={radius_deg}]
```

Key	Explanation	Required	Type
latitude	The latitude where the weather-stations should be close to.	Yes	double
longitude	The longitude where the weather-stations should be close to.	Yes	double
radius_km	The radius in kilometers from the given coordinate.	No	double
radius_deg	The radius in degrees from the given coordinate.	No	double

Output

```
[
    {
        "id": 801100,
        "name": "MEDELLIN/OLAYA HERR",
        "country": "COLOMBIA",
        "latitude": 6.217,
        "longitude": -75.6,
        "elevation": 1499.0
    },
    {
        "id": 801120,
        "name": "RIONEGRO/J.M.CORDOV",
        "country": "COLOMBIA",
        "latitude": 6.133,
        "longitude": -75.433,
        "elevation": 2142.0
    }
]
```

2.5. Retrieving a list of weather-station by radius (paged)

Input

Either `radius_km` or `radius_deg` should be used, not both!

```
GET /api/v1/stations/radius
    ?latitude={latitude}
    &longitude={longitude}
    [&radius_km={radius_km}]
    [&radius_deg={radius_deg}]
    [&page={page}]
    [&size={size}]
```

Key	Explanation	Required	Type
latitude	The latitude where the weather-stations should be close to.	Yes	double
longitude	The longitude where the weather-stations should be close to.	Yes	double
radius_km	The radius in kilometers from the given coordinate.	No	double
radius_deg	The radius in degrees from the given coordinate.	No	double
page	The current page you are on.	No	double
size	The size of the pages.	No	double

Output

```json
{
    "content": [
        {
            "id": 801100,
            "name": "MEDELLIN/OLAYA HERR",
            "country": "COLOMBIA",
            "latitude": 6.217,
            "longitude": -75.6,
            "elevation": 1499.0
        },
        {
            "id": 801120,
            "name": "RIONEGRO/J.M.CORDOV",
            "country": "COLOMBIA",
            "latitude": 6.133,
            "longitude": -75.433,
            "elevation": 2142.0
        }
    ],
    "pageable": {
        "sort": {
            "sorted": false,
            "unsorted": true,
            "empty": true
        },
        "offset": 0,
        "pageNumber": 0,
        "pageSize": 20,
        "paged": true,
        "unpaged": false
    },
    "totalPages": 1,
    "totalElements": 2,
```

```json
    "last": true,
    "size": 20,
    "number": 0,
    "sort": {
        "sorted": false,
        "unsorted": true,
        "empty": true
    },
    "first": true,
    "numberOfElements": 2,
    "empty": false
}
```

3. Measurements

3.1. Retrieving a measurement by ID

Input

```
GET /api/v1/measurements/{measurement_id}/
```

Key	Explanation	Required	Type
measurement_id	The ID of the measurement you want to retrieve information from.	Yes	integer

Output

```json
{
    "id": 1,
    "station_id": 5325,
    "date": "2020-01-23 12:52:20",
    "temperature": 25.0,
    "dew_point": 20.0,
    "air_pressure_station": 50.0,
    "air_pressure_sea": 100.0,
    "visibility": 30.0,
    "wind_speed": 15.0,
    "wind_direction": 180.0,
    "rainfall": 15.0,
    "snowfall": 15.0,
    "cloud_coverage": 30,
    "freeze": false,
    "rain": true,
    "snow": false,
    "hail": false,
    "storm": false,
    "tornado": true
}
```

3.2. Retrieving a list of all measurements (by station id) (paged)

Input

```
GET /api/v1/paged/measurements?station_id={station_id}
```

Key	Explanation	Required	Type
station_id	The ID of the station you want to retrieve information from.	No	integer

Output

```
{
    "content": [
        {
            "id": 24094,
            "stationId": 103820,
            "date": "2020-01-26 12:52:24",
            "temperature": -0.5,
            "dew_point": -3.5,
            "air_pressure_station": 1007.1,
            "air_pressure_sea": 1006.9,
            "visibility": 9.4,
            "wind_speed": 13.0,
            "wind_direction": 89.0,
            "rainfall": 0.06,
            "snowfall": 0.3,
            "cloud_coverage": 56.7,
            "freeze": false,
            "rain": false,
            "snow": false,
            "hail": false,
            "storm": false,
            "tornado": false
        }
        ...
    ],
    "pageable": {
        "sort": {
            "sorted": false,
            "unsorted": true,
            "empty": true
        },
        "pageNumber": 0,
        "pageSize": 2,
        "offset": 0,
        "paged": true,
        "unpaged": false
    },
    "last": false,
    "totalPages": 13,
    "totalElements": 26,
    "first": true,
    "number": 0,
    "sort": {
        "sorted": false,
        "unsorted": true,
        "empty": true
```

```json
    },
    "numberOfElements": 2,
    "size": 2,
    "empty": false
}
```

3.3. Retrieving averages of measurements by time-frame grouped by station ID (paged)

Input

```
GET /api/v1/measurements/average
```

Body:

```
{
    "startDate": "1970-01-01 00:00:00",
    "endDate": "2020-01-26 14:00:00"
}
```

Key	Explanation	Required	Type
startDate	The minimum date used in the averages.	Yes	string-pattern: `yyyy-MM-dd HH:mm:ss`
endDate	The maximum date used in the averages.	Yes	string-pattern: `yyyy-MM-dd HH:mm:ss`

Output

```
[
    {
        "stationId": 10010,
        "date": "2020-01-26 12:00:00",
        "rainfall": 0.10,
        "visibility": 28.88,
        "snowfall": 0.0,
        "airPressureStation": 1000.63,
        "cloudCoverage": 51.42,
        "temperature": -2.25,
        "windDirection": 78.37,
        "dewPoint": -4.94,
        "windSpeed": 25.40,
        "airPressureSea": 1004.0
    },
    {
        "stationId": 10015,
        "date": "2020-01-26 12:00:00",
        "rainfall": 0.0,
        "visibility": 10.41,
        "snowfall": 0.0,
        "airPressureStation": 989.33,
        "cloudCoverage": 1.74,
        "temperature": -0.45,
        "windDirection": 59.23,
        "dewPoint": -1.78,
        "windSpeed": 4.68,
        "airPressureSea": 972.08
    }
]
```

3.4. Retrieving averages of measurements by time-frame and station ID

Input

```
GET /api/v1/stations/{stationId}/measurements/average
```

Body:

```
{
    "startDate": "1970-01-01 00:00:00",
    "endDate": "2020-01-26 14:00:00"
}
```

Key	Explanation	Required	Type
station_id	The id of the station you want to retrieve an average from	Yes	integer
startDate	The minimum date used in the averages.	Yes	string-pattern: `yyyy-MM-dd HH:mm:ss`
endDate	The maximum date used in the averages.	Yes	string-pattern: `yyyy-MM-dd HH:mm:ss`

Output

```
{
    "stationId": 740790,
    "rainfall": 0.0,
    "visibility": 29.72,
    "snowfall": 0.0,
    "airPressureStation": 986.83,
    "airPressureSea": 1019.41,
    "cloudCoverage": 99.60,
    "temperature": -26.74,
    "windDirection": 160.05,
    "dewPoint": -32.23,
    "windSpeed": 11.03
}
```

3.5. Retrieving averages per day from last month by station ID

Input

```
GET /api/v1/stations/{station_id}/measurements/average/month?type={type}
```

Key	Explanation	Required	Type
station_id	The id of the station you want to retrieve averages from	Yes	integer
type	The average-type you want to retrieve from the database, current possibilities are: `vegaflor` which returns a set of specified averages from the database	No	string

Output

Unspecified type:

```
[
    {
        "stationId": 160640,
        "year": 2020,
        "month": 1,
        "day": 28,
        "rainfall": 0.0,
        "visibility": 5.16,
        "snowfall": 0.0,
        "cloudCoverage": 28.82,
        "dewPoint": -2.88,
        "airPressureStation": 997.76,
        "temperature": -0.59,
        "windDirection": 7.47,
        "windSpeed": 10.50,
        "airPressureSea": 1003.12
    },
    ...
]
```

`vegaflor` type:

```
[
    {
        "year": 2020,
        "month": 1,
        "day": 28,
        "avgTemperature": -0.59,
        "minTemperature": -0.6,
        "maxTemperature": -0.2,
        "avgRainfall": 0.0,
        "minRainfall": 0.0,
        "maxRainfall": 0.0,
        "avgDewPoint": -2.88
    },
    ...
]
```

3.6. Retrieving desirable locations by humidity and temperature

Input

```
GET /api/v1/measurements/average/month/desirable?limit={limit}
```

Key	Explanation	Required	Type
limit	The maximum amount of results that should be returned.	No	integer

Output

```
[
    {
        "maxTemperature": 26.0,
        "avgRainfall": 0.11647450194555746,
        "minTemperature": 6.1,
        "avgDewPoint": 20.711086480421926,
        "avgHumidity": 80.07556310373644,
        "maxRainfall": 0.39,
        "avgTemperature": 24.36962307001693,
        "StationId": 800010,
        "minRainfall": 0.01
    },
    {
        "maxTemperature": 26.0,
        "avgRainfall": 0.0,
        "minTemperature": 8.4,
        "avgDewPoint": 20.094900266558525,
        "avgHumidity": 80.6307035441871,
        "maxRainfall": 0.0,
        "avgTemperature": 23.621507665799093,
        "StationId": 824110,
        "minRainfall": 0.0
    },
    ...
]
```

4. Machine Queries

The queries in the following chapter can only be used by users with the authority level *ROLE_ROBOT*

4.1. Request Connection

The response of the server may vary depending on the state it is currently in.

Input

```
GET /api/v1/connection?type={connection_type}
```

Key	Explanation	Required	Type
connection_type	The type of connection this machine wants to make with the database. Current possibilities: `request_connection` and `stop_server`	Yes	string

Output

`request_connection` type:

```
{
    "port": 9999,
    "response": "LISTENING",
    "ip_address": "xxx.xxx.xxx.xxx"
}
```

`stop_server` type:

```
{ "response": "SUCCESS" }
```

5. Administrator Queries

The queries in the following chapter can only be used by users with the authority level *ROLE_ADMIN*

5.1. Retrieving information about a user

Input

```
GET /api/v1/users/{username}/
```

Key	Explanation	Required	Type
username	The username of the user where information should be retrieved from.	Yes	string

Output

```json
{
    "username": "user1",
    "enabled": true,
    "authorities": [ "ROLE_USER", "ROLE_ADMIN" ]
}
```

5.2. Retrieving a list of all users

Input

```
GET /api/v1/users/
```

Output

```json
[
    {
        "username": "user_1",
        "enabled": true,
        "authorities": [ "ROLE_ADMIN", "ROLE_USER" ]
    },
    {
        "username": "user_2",
        "enabled": true,
        "authorities": [ "ROLE_USER" ]
    },
    {
        "username": "robot_1",
        "enabled": true,
        "authorities": [ "ROLE_ROBOT" ]
    }
]
```

5.3. Retrieving a list of all users (paged)

Input

```
GET /api/v1/paged/users
    [?page={page}]
    [&size={size}]
```

Key	Explanation	Required	Type
page	The current page you are on.	No	integer
size	The size of the pages.	No	integer

Output

```
{
    "content": [
        {
            "username": "user_1",
            "enabled": true,
            "authorities": [ "ROLE_ADMIN", "ROLE_USER" ]
        },
        {
            "username": "user_2",
            "enabled": true,
            "authorities": [ "ROLE_USER" ]
        }
    ],
    "pageable": {
        "sort": {
            "sorted": false,
            "unsorted": true,
            "empty": true
        },
        "pageNumber": 0,
        "pageSize": 20,
        "offset": 0,
        "paged": true,
        "unpaged": false
    },
    "totalPages": 1,
    "totalElements": 2,
    "last": true,
    "first": true,
    "sort": {
        "sorted": false,
        "unsorted": true,
        "empty": true
    },
    "number": 0,
    "numberOfElements": 2,
    "size": 20,
    "empty": false
}
```

5.4. Deleting a user

Input

```
DELETE /api/v1/users/{username}/
```

Key	Explanation	Required	Type
username	The username of the user that should be removed from the database.	Yes	string

Output

```
Status: 200 OK
```

5.5. Adding a new user

Input

```
POST /api/v1/users
```

Body:

```
{
    "username": "{username}",
    "password": "{password}",
    "enabled": {enabled}
}
```

Key	Explanation	Required	Type
username	The username of the user that should be added to the database.	Yes	string
password	The password this user should have.	Yes	string
enabled	If the user should be enabled.	Yes	boolean

Output

```
Status: 201 Created
Location: http://DOMAIN/api/v1/users/{username}/
```

5.6. Updating user information

Usernames can not be modified.

Input

```
PUT /api/v1/users/{username}/
```

Body:

```
{ "password": "{password}", "enabled": {enabled} }
```

Key	Explanation	Required	Type
username	The username of the user that should be updated.	Yes	string
password	The new password this user should have.	No	string
enabled	If the user should be enabled.	No	boolean

Output

```
Status: 200 OK
```

5.7. Removing authority from a user

Input

`DELETE /api/v1/users/authority/{username}?role={role}`

Key	Explanation	Required	Type
username	The username of the user where a role should be taken from.	Yes	string
role	The role that should be taken from this user. Available roles are: `ROLE_USER`, `ROLE_ROBOT` and `ROLE_ADMIN`.	Yes	string

Output

`Status: 200 OK`

5.8. Adding authority to a user

Input

`POST /api/v1/users/authority/{username}?role={role}`

Key	Explanation	Required	Type
username	The username of the user where a role should be taken from.	Yes	string
role	The role that should be taken from this user. Available roles are: `ROLE_USER`, `ROLE_ROBOT` and `ROLE_ADMIN`.	Yes	string

Output

```
Status: 201 Created
Location: http://DOMAIN/api/v1/users/{username}/
```

6. User Queries

6.1. Logout

This method will invalidate the cookie created by the authentication (which makes the user logout).

Input

```
GET /api/v1/logout
```

Output

```
Status: 200 OK
```

6.2. Retrieving information about current logged in user

Input

```
GET /api/v1/me
```

Output

```json
{
    "username": "currentuser",
    "enabled": true,
    "authorities": [
        "ROLE_USER"
    ]
}
```