* DB architecture
* SQL queries
* Selecting [datatype length](https://dev.mysql.com/doc/refman/8.0/en/data-types.html)
  + Static data
    - number:  small int (16 bit)
    - contract\_name: varchar(32) world capitals names
    - name: varchar(32) with reference to contract\_name
    - address: varchar(32) with reference to contract\_name
    - Location

Searched for [DECIMAL](https://dev.mysql.com/doc/refman/8.0/en/fixed-point-types.html), [POINT](https://dev.mysql.com/doc/refman/8.0/en/gis-class-point.html) datatypes, settled for float as per [google developer suggestion](https://developers.google.com/maps/documentation/javascript/mysql-to-maps#createtable)

* + - * Lat: float(10,6)
      * Lng: float(10,6)
    - bonus: Boolean (auto formatted to TINYINT)
    - Banking: Boolean (auto formatted to TINYINT)
    - entry\_timestamp: timestamp timestamp default value given [0000-00-00 00:00:00] not accepted
  + Dynamic data
    - bike\_stands: small int (16 bit)
    - available\_bike\_stands: small int (16 bit)
    - available\_bikes: small int (16 bit)
    - status: varchar(8) OPEN CLOSE
    - last\_update: timestamp
    - entry\_timestamp: timestamp
  + [Weather dynamic data](https://openweathermap.org/current)

If you do not see some of the parameters in your API response it means that these weather phenomena are just not happened for the time of measurement for the city or location chosen. Only really measured or calculated data is displayed in API response.

* + - coord
      * coord.lon City geo location, longitude float(10,6)
      * coord.lat City geo location, latitude float(10,6)
    - weather (more info Weather condition codes)
      * weather.id Weather condition id small int (16 bit)
      * weather.main Group of weather parameters (Rain, Snow, Extreme etc.) varchar(16)
      * weather.description Weather condition within the group. You can get the output in your language. varchar(32)
      * weather.icon Weather icon id varchar(16)
    - base Internal parameter varchar(16)
    - main
      * main.temp Temperature. Unit Default: Kelvin, Metric: Celsius, Imperial: Fahrenheit. float(5,2)
      * main.feels\_like Temperature. This temperature parameter accounts for the human perception of weather. Unit Default: Kelvin, Metric: Celsius, Imperial: Fahrenheit. float(5,2)
      * main.pressure Atmospheric pressure (on the sea level, if there is no sea\_level or grnd\_level data), hPa SMALLINT
      * main.humidity Humidity, % SMALLINT
      * main.temp\_min Minimum temperature at the moment. This is deviation from current temp that is possible for large cities and megalopolises geographically expanded (use these parameter optionally). Unit Default: Kelvin, Metric: Celsius, Imperial: Fahrenheit. float(5,2)
      * main.temp\_max Maximum temperature at the moment. This is deviation from current temp that is possible for large cities and megalopolises geographically expanded (use these parameter optionally). Unit Default: Kelvin, Metric: Celsius, Imperial: Fahrenheit. float(5,2)
      * main.sea\_level Atmospheric pressure on the sea level, hPa SMALLINT
      * main.grnd\_level Atmospheric pressure on the ground level, hPa SMALLINT
    - wind
      * wind.speed Wind speed. Unit Default: meter/sec, Metric: meter/sec, Imperial: miles/hour. float(5,2)
      * wind.deg Wind direction, degrees (meteorological) SMALLINT
    - clouds
      * clouds.all Cloudiness, % SMALLINT
    - Visibility SMALLINT
    - rain
      * rain.1h Rain volume for the last 1 hour, mm float(5,2)
      * rain.3h Rain volume for the last 3 hours, mm float(5,2)
    - snow
      * snow.1h Snow volume for the last 1 hour, mm float(5,2)
      * snow.3h Snow volume for the last 3 hours, mm float(5,2)
    - dt Time of data calculation, unix, UTC TIMESTAMP
    - sys
      * sys.type Internal parameter SMALLINT
      * sys.id Internal parameter SMALLINT
      * sys.message Internal parameter varchar(8)
      * sys.country Country code (GB, JP etc.) varchar(8)
      * sys.sunrise Sunrise time, unix, UTC TIMESTAMP
      * sys.sunset Sunset time, unix, UTC TIMESTAMP
    - timezone Shift in seconds from UTC TIMESTAMP
    - id City ID varchar(16)
    - name City name varchar(32)
    - cod Internal parameter SMALLINT
* API call python
  + Json
  + request