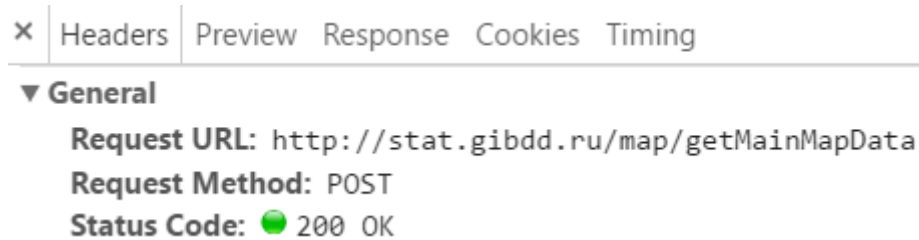


Изучаем поведение сайта: Chrome и DevTools

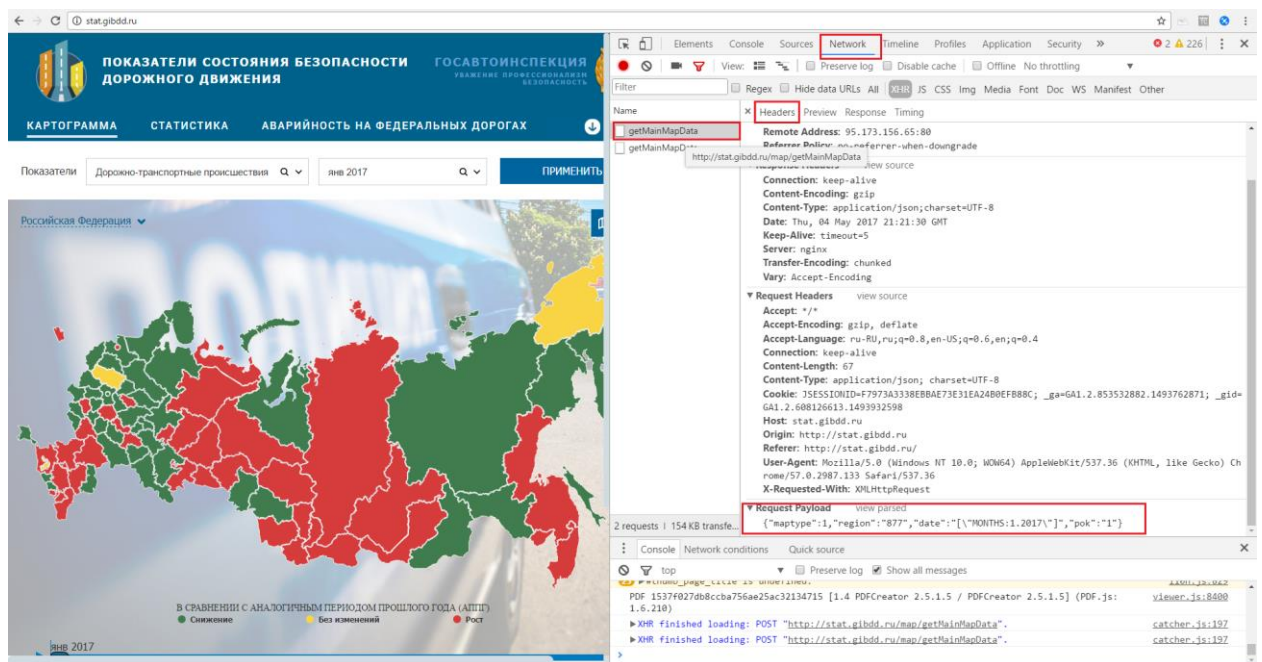
По правой кнопке на страничке выбрать «Просмотреть код». Откроется окно DevTools.

В нем в меню выбрать вкладку **Network** и перезагрузить веб-страничку. В окне DevTools появится ссылка на url запроса под названием `getMainMapData`. Нажать ее в окне справа выбрать **Headers**. Самый последний раздел в нем – **Request Payload**. Это данные, которые отправляются сайту при вызове url. Для того, чтобы получить payload в формате json, нажимаем **view source** рядом с надписью Request Payload.

Метод, которым отправляется запрос на сайт, и url смотрим в разделе **General** в **Headers**:



Для нашего случая url: <http://stat.gibdd.ru/map/getMainMapData>,
payload: `{"maptype":1,"region":"877","date":["MONTHS:1.2017"],"pok":"1"}`,
метод: POST.



При желании можно узнать User agent, cookies и другие параметры – они могут понадобиться при подключении к сайту из python. Их берем из раздела **Request Headers**:

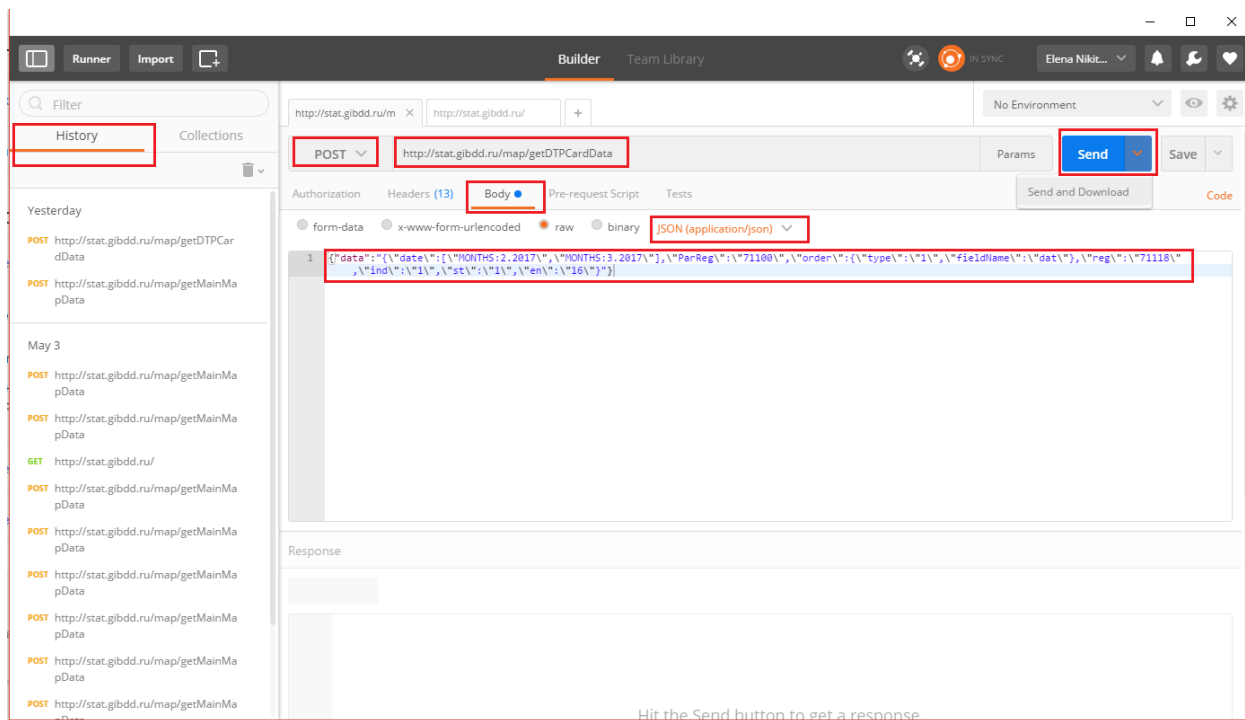
▼ Request Headers [view source](#)

Accept: */*
Accept-Encoding: gzip, deflate
Accept-Language: ru-RU,ru;q=0.8,en-US;q=0.6,en;q=0.4
Connection: keep-alive
Content-Length: 67
Content-Type: application/json; charset=UTF-8
Cookie: JSESSIONID=F7973A3338EBBAE73E31EA24B0EFB88C; _ga=GA1.2.853532882.1493762871; _gid=GA1.2.608126613.1493932598
Host: stat.gibdd.ru
Origin: http://stat.gibdd.ru
Referer: http://stat.gibdd.ru/
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/57.0.2987.133 Safari/537.36
X-Requested-With: XMLHttpRequest

Получаем данные при помощи Postman

Чтобы предварительно проверить получение данных без написания кода, используем приложение <https://chrome.google.com/webstore/detail/postman/fhbjgbfijnljbdggehccddcbncdddodomop> и к нему дополнение Postman Interceptor <https://chrome.google.com/webstore/detail/postman-interceptor/aicmkgpgakddgnaphhhpliifpcfhhicfo?hl=ru>.

Чтобы открыть Postman, наберите в адресной строке Chrome **chrome://apps**. Откроется страничка с установленными приложениями, среди которых будет Postman.



В Postman будем вставлять все данные, полученные из хрома: в адресную строку копируем наш url, выбираем метод POST.

Выбираем вкладку **Body** и в текстовое окно вставляем текст, который мы взяли из Request Payload. Формат ответа выбираем **JSON (application/json)**.

Справа вверху есть кнопка **Send**. Нажимаем на стрелку рядом с ней, выбираем **Send and Download**. После этого Postman выполнит запрос к сайту и, если все хорошо, появится диалог скачивания файла. Это тот самый json, которого мы ждали, - с данными ГИБДД.

Получаем данные при помощи python

Для работы с сайтом нужна библиотека requests.

Получение данных: `r = requests.post (url, json = payload)`. Здесь url и payload – в точности те значения, что мы получили через DevTools и добавляли в Postman.

Для работы с полученными данными нужна библиотека json. Преобразование полученного объекта в json: `json.loads (r.content)`

Дополнительные полезные инструменты

json часто приходит с экранированными управляющими символами. Чтобы убрать экранирование, есть онлайн-сервис <http://www.freeformatter.com/json-escape.html>

JSON String Escape / Unescape

Escapes or unescapes a JSON string removing traces of offending characters that could prevent parsing.

The following characters are reserved in JSON and must be properly escaped to be used in strings:

- **Backspace** is replaced with `\b`
- **Form feed** is replaced with `\f`
- **Newline** is replaced with `\n`
- **Carriage return** is replaced with `\r`
- **Tab** is replaced with `\t`
- **Double quote** is replaced with `\"`
- **Backslash** is replaced with `\\`

Copy-paste the string to escape or unescape here

```
{ "data": { "date": [ "MONTHS:2.2017", "MONTHS:3.2017" ], "ParReg": "71100", "order": { "type": "1", "fieldName": "dat", "reg": "71118", "ind": "1", "st": "1", "en": "16" } }
```

ESCAPE

UNESCAPE

Ad closed by Google

Report this ad

Ads by Google ⓘ

Unescaped string:

```
{ "data": { "date": [ "MONTHS:2.2017", "MONTHS:3.2017" ], "ParReg": "71100", "order": { "type": "1", "fieldName": "dat", "reg": "71118", "ind": "1", "st": "1", "en": "16" } }
```

Json Parser Online You like it? Support it! [Donate](#)

```

String parse
{
    "data": {}
    "RegName": "Ханты-Мансийский автономный округ - Югра,
Нефтегазский район",
    "tab": {}
    {
        "KartId": "204358068",
        "rowNum": 1,
        "date": "31.03.2017",
        "Time": "21:40",
        "District": "Нефтегазский район",
        "OTS_UV": "Наезд на препятствие",
        "POG": 0,
        "RAN": 1,
        "K_TS": 1,
        "K_UCH": 1,
        "infoDup": {}
        "ndu": {}
        "Не установлен"
    },
    "ador": {}
    "Перегон (нет объектов на месте ДТП)",
    "Мост, аэстакада, путепровод"
    },
    "ts_info": {}
    {
        "n_ts": "1",
        "ts_d": "Вос",
        "ts_ts": "Сезонные огачи",
        "marks_ts": "KMA3",
        "m_ts": "6426",
        "color": "Оранжевый",
        "f_rule": "Полосный",
        "n_uv": "2044",
        "m_prov": "Передний левый бок (передняя дверь
колески, в случае ТС с правым расположением
руля передняя дверь пассажира) | Передний
левый угол",
        "n_d": "Технические неисправности
аппаратуры",
        "f_sob": "Коллективная собственность

```