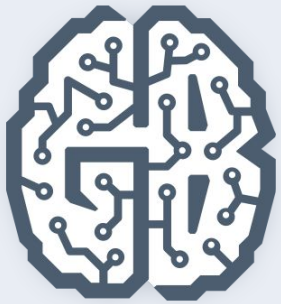
Методы сбора и обработки данных при помощи Python

Урок 1



Введение в методы сбора и обработки данных

Краткий обзор технологий для понимания сбора и обработки данных

Регламент

- 8 уроков по 2 часа
- Домашние задания
- Видеозапись будет
- Задавайте вопросы

Что мы будем изучать на курсе?

1. Основы компьютерных сетей.
2. Основы HTTP, веба и форматы данных (JSON, XML, CSV).
3. Принципы работы REST и SOAP.
4. Работа с MongoDB.

Каких результатов мы добьемся?

1. Узнаем, как работают сервисы и приложения в Интернете.
2. «Пообщаемся» при помощи Python с сервисами.
3. Узнаем о форматах данных.
4. Близко познакомимся с MongoDB.

По итогу курса

- Работа с RESTful-сервис.
- Работа с SOAP.
- Парсинг HTML-сайта с данными.
- Парсинг open data.
- Полученную БД MongoDB с данными мы в дальнейшем будем использовать для анализа.

Ваши ожидания от курса

- Какие уже есть вопросы?
- Ваши ожидания от курса?

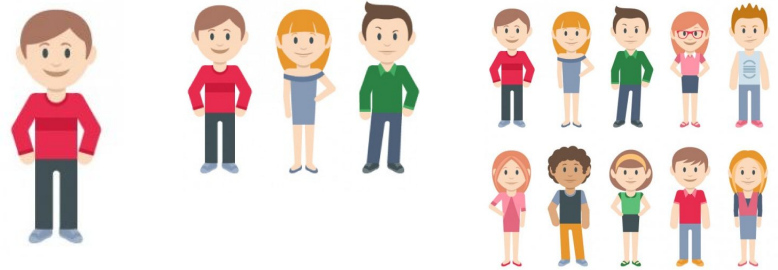
План урока

1. Введение в компьютерные сети.
2. Модель OSI.
3. Протоколы TCP и UDP.
4. Глобальные и частные IP-адреса, MAC-адреса, NAT
5. Основы HTTP
   HTTP и HTTPS.
   HTTP-заголовки, коды и cookies.
   Что такое API.

   К концу урока мы разберемся во всех базовых технологиях, на которых построена Всемирная паутина.

Зачем программисту знать, как работают сетевые технологии?

- Масштабирование приложения.



- Производительность приложения.

- Безопасность приложения.

Модель OSI

- 2 группы слоев – Host Layers и Media Layers.
- 7 слоев – 3 в Media и 4 в Host.
- Протокол модели OSI взаимодействует с протоколами своего уровня и уровнем выше/ниже.
- Протокол модели OSI может выполнять только функции своего уровня.

Модель OSI – слои

- Прикладной
- Уровень представления
- Сессионный
- Транспортный
- Сетевой
- Канальный
- Физический

Модель OSI – слои

NAT, IP- и MAC-адреса

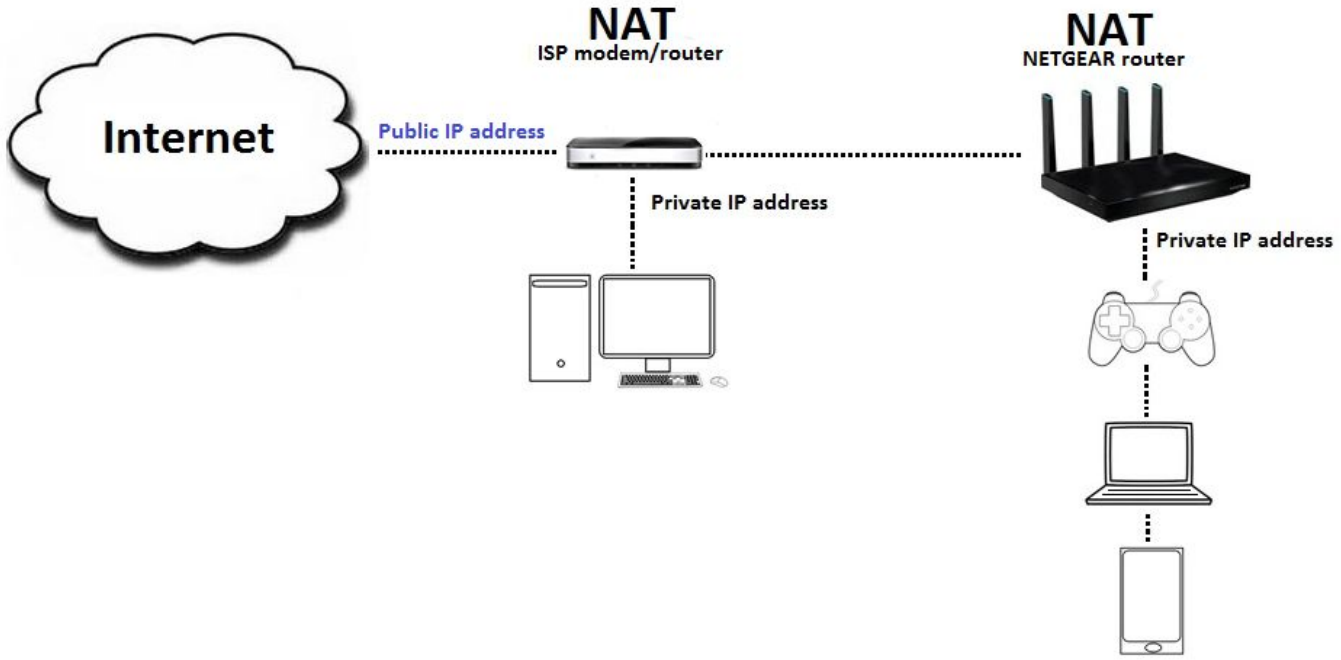
IP- адреса

- IPv4 – 32-битовое число. Привычная форма записи – четыре десятичных числа от 0 до 255, разделенных точками (192.168.80.19). Максимум 2^32 уникальных адресов.
- IPv6 – 128-битовое число. Записывается как восемь шестнадцатеричных чисел, разделенных двоеточиями (2001:cdba:0000:0000:0000:0000:3257:9652).

Маска

255.255.255.255	11111111	11111111	11111111	11111111	/32
255.255.255.254	11111111	11111111	11111111	11111110	/31
255.255.255.252	11111111	11111111	11111111	11111100	/30
255.255.255.248	11111111	11111111	11111111	11111000	/29
255.255.255.240	11111111	11111111	11111111	11110000	/28
255.255.255.224	11111111	11111111	11111111	11100000	/27
255.255.255.192	11111111	11111111	11111111	11000000	/26
255.255.255.128	11111111	11111111	11111111	10000000	/25
255.255.255.0	11111111	11111111	11111111	00000000	/24
255.255.254.0	11111111	11111111	11111110	00000000	/23
255.255.252.0	11111111	11111111	11111100	00000000	/22
255.255.248.0	11111111	11111111	11111000	00000000	/21
255.255.240.0	11111111	11111111	11110000	00000000	/20
255.255.224.0	11111111	11111111	11100000	00000000	/19
255.255.192.0	11111111	11111111	11000000	00000000	/18
255.255.128.0	11111111	11111111	10000000	00000000	/17

NAT - Network Address Translation



MAC адрес


Структура MAC адреса.
1
1
22 бита
24 бита
0/1
Глобальный/
Локальный
Уникальное значение
выбираемое производителем
0/1
Одиночный/
Групповой
Универсальный
идентификатор
производителя платы
Например 00-1A-4D-2D-C2-6A
00000000-00011010-01001101-00101101-11000010-01101010

NAT, IP- и MAC-адреса – MAC

```
Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Qualcomm Atheros QCA9377 Wireless
Network Adapter
   Physical Address. . . . . . . . . : 80-C5-F2-70-8F-A3
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . :
fe80::3ce2:11f9:7c8:4ec4%15(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.0.151(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : 25 июня 2019 г. 18:11:16
   Lease Expires . . . . . . . . . . : 25 июня 2019 г. 21:11:16
   Default Gateway . . . . . . . . . : 192.168.0.1
   DHCP Server . . . . . . . . . . . : 192.168.0.1
   DHCPv6 IAID . . . . . . . . . . . : 411092466
   DHCPv6 Client DUID. . . . . . . . :
00-01-00-01-22-FA-EB-94-2C-FD-A1-38-69-C2
   DNS Servers . . . . . . . . . . . : 192.168.0.1
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

HTTP/HTTPS, API

HTTP и HTTPS


Пароль:
admin12345
4OvZ5$n0kY
admin12345
Браузер
Отправляет
admin12345
Сервер
Получает
admin12345
Хакер
Перехватывает
4OvZ5$n0kY

HTTP и HTTPS – методы

- GET – получить ресурс
- PUT – обновить
- POST – создать
- DELETE – удалить
- PATCH – исправить
- HEAD
- OPTIONS
- TRACE
- CONNECT

HTTP и HTTPS – заголовки

- General
- Request
- Response
- Entity

HTTP и HTTPS – заголовки

```
> GET / HTTP/1.1
> Host: google.com
> User-Agent: curl/7.54.0
> Accept: */*
>
< HTTP/1.1 301 Moved Permanently
< Location: http://www.google.com/
< Content-Type: text/html; charset=UTF-8
< Date: Sun, 08 Jul 2018 11:45:03 GMT
< Expires: Tue, 07 Aug 2018 11:45:03 GMT
< Cache-Control: public, max-age=2592000
< Server: gws
< Content-Length: 219
< X-XSS-Protection: 1; mode=block
< X-Frame-Options: SAMEORIGIN
<
<HTML><HEAD><meta http-equiv="content-type"
content="text/html;charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="http://www.google.com/">here</A>.
</BODY></HTML>
```

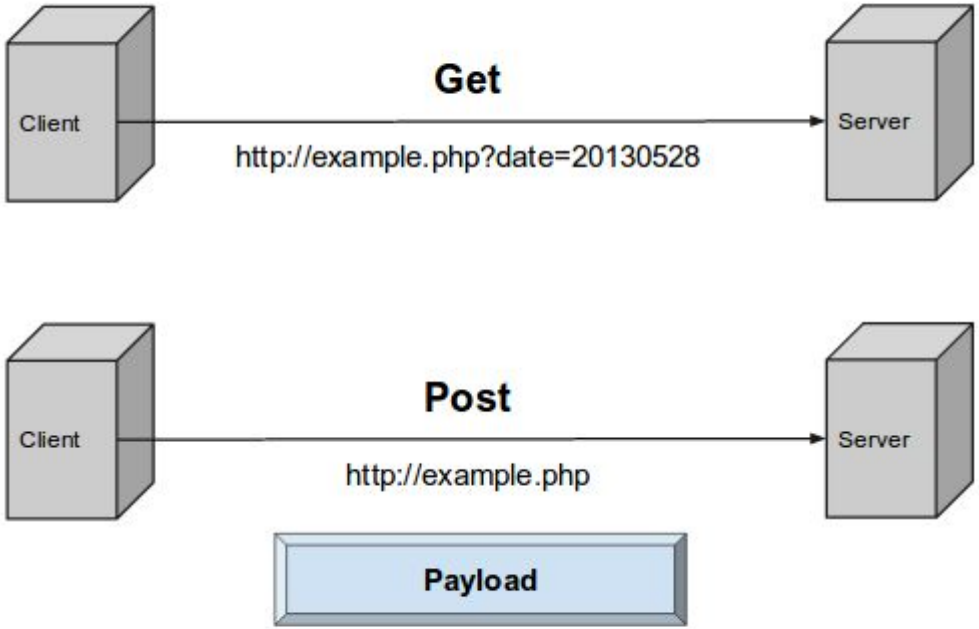
HTTP и HTTPS – заголовки

```
HTTP/1.1 401 Authorization Required
Date: Tue, 01 Mar 2005 11:30:10 GMT
Server: Apache/1.3.33 (Unix)
WWW-Authenticate: Basic realm="How about authorization?"
Connection: close
Content-Type: text/html; charset=iso-8859-1
```

HTTP и HTTPS – коды

- 1xx: Information;
- 2xx: Success;
- 3xx: Redirect;
- 4xx: Client Error;
- 5xx: Server Error.

GET vs. POST


Client
Get
Server
http://example.php?date=20130528
Client
Post
Server
http://example.php
Payload

Утилита Fiddler

https://www.telerik.com/fiddler

https://docs.telerik.com/fiddler/Configure-Fiddler/Tasks/FirefoxHTTPS

https://www.youtube.com/watch?v=Fd7tWOiKiMA – видео-инструкция

Что такое API

Что такое API

- Private API
- Public API
- Набор классов и библиотек

Организационные вопросы

- Пишите в комментарии к уроку. Я буду отвечать на них каждый день.
- Личные сообщения.
- Видео буду выкладывать в день урока (самое позднее – на следующий день).

Домашнее задание

- Установить Fiddler, настроить отображение HTTPS трафика

- Сделать POST-запрос для заполнения какой-либо формы на сайте, прислать скриншот POST-параметров из Fiddler’а

  Посмотреть документацию к API гитхаба, разобраться как вывести список репозиториев для конкретного пользователя, прислать JSON-вывод в текстовом файле.

- * Выбрать тематику данных: лучшего всего для этого посмотреть, какие сервисы есть в публичном доступе там, откуда вы сможете эти данные собрать. Нам нужны RESTful- и SOAP-сервисы.

Ваши вопросы?