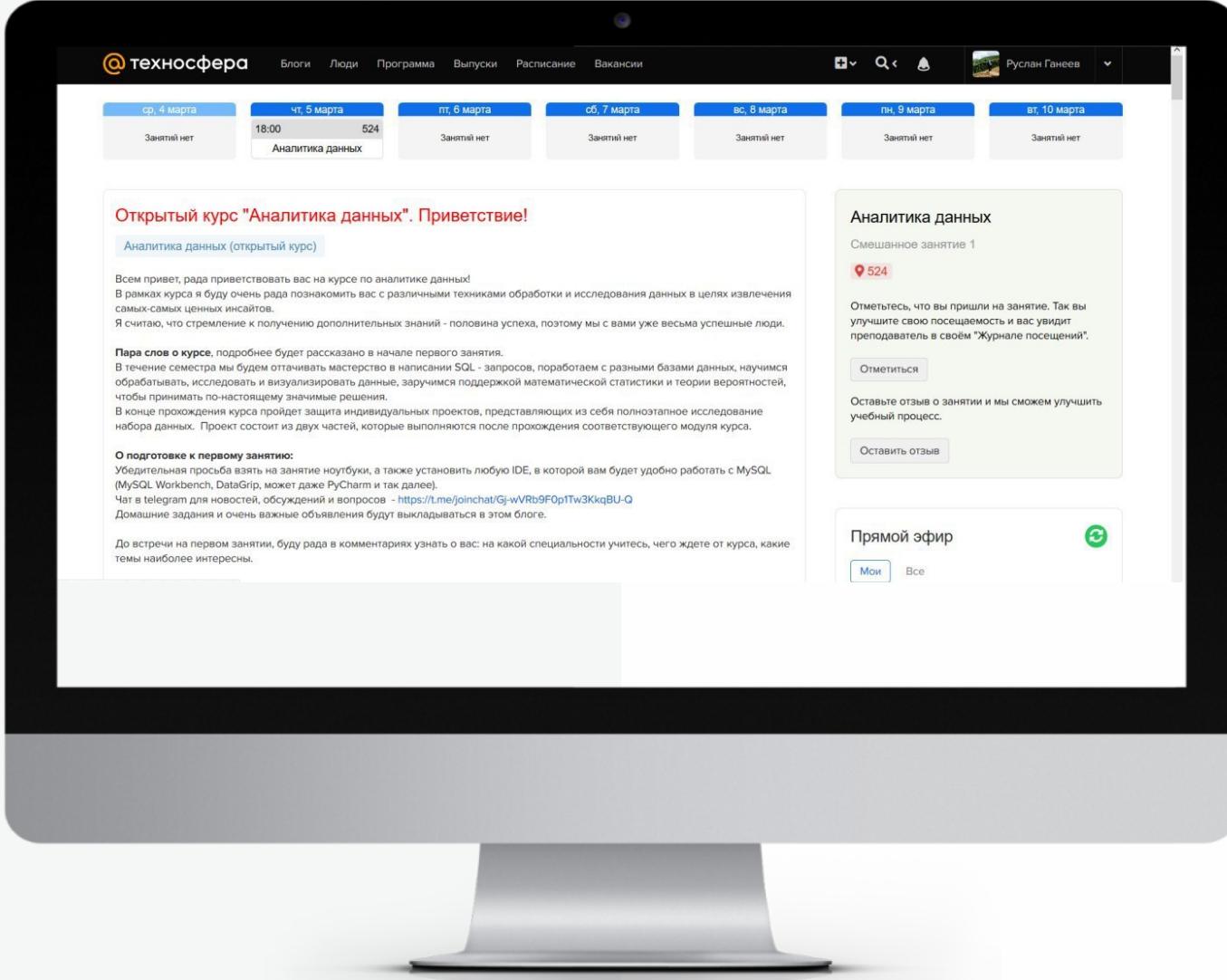


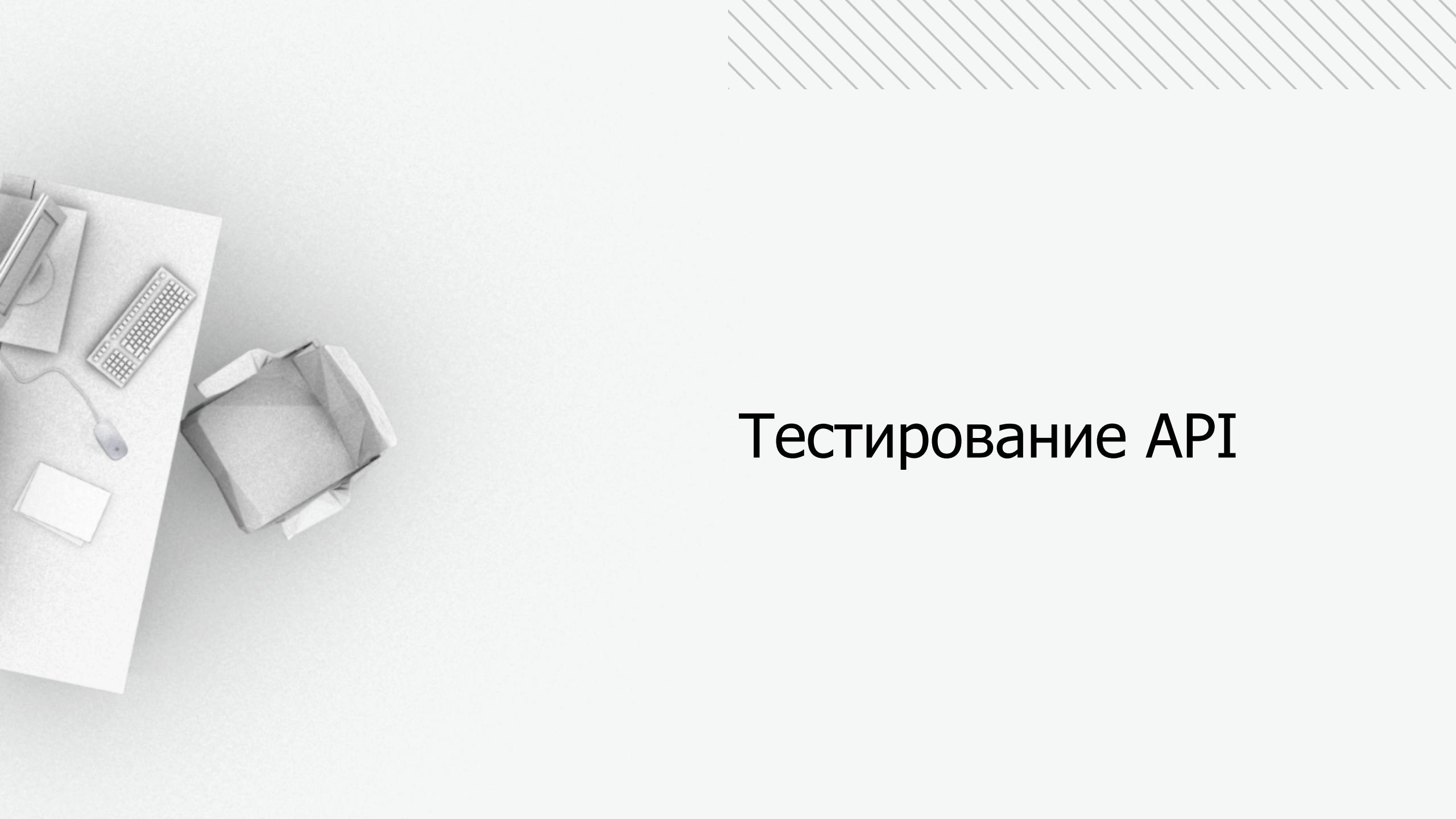
# Автоматизированное тестирование на Python

Кириллов Илья



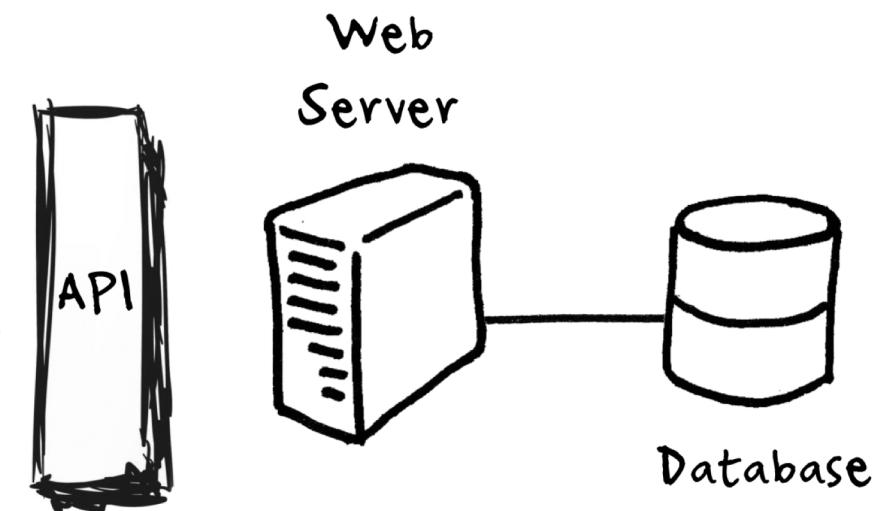
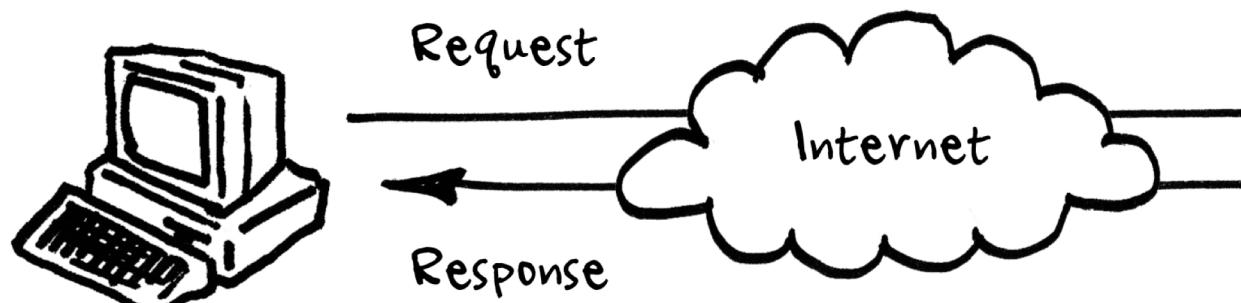


# Напоминание отметиться на портале



# Тестирование API

Web app  
in  
Browser



# Данные могут быть

- 1. Статические**
  - Это файлы на диске сервера
  - Как правило обладают постоянным адресом
- 2. Динамические**
  - Создаются на каждый запрос
  - Содержимое зависит от параметров (например время или имя пользователя)
  - Адрес может быть постоянным, а может меняться

# Типы данных (Mime Types)

- **text/html**
- **text/css**
- **application/javascript**
- **image/png**

- **video/mp4**
- **text/xml**
- **application/json**
- **image/jpeg**

# text / html

```
<html>
<body>
  <link rel="stylesheet" href="/css/style.css">
  <script src="http://code.jquery.com/jquery-2.1.4.js">
  </script>
  <p>Some text with 
    and <a href="#yes">hyperlinks</a>
  </p>
</body>
</html>
```

## text / css

```
.hljs-subst,  
.hljs-title,  
.json .hljs-value {  
    font-weight: normal;  
    color: #000;  
}
```

## text / xml

```
<response status="ok">
  <friends>
    <friend id="1" name="v.pupkin"/>
    <friend id="2" name="a.pushkin"/>
    <friend id="3" name="n.tesla"/>
  </friends>
</response>
```

# application / json

```
{  
    "status": "ok",  
    "friends": [  
        { "id": 1, "name": "v.pupkin" },  
        { "id": 2, "name": "a.pushkin" },  
        { "id": 3, "name": "n.tesla" }  
    ]  
}
```

# URL – Unified Resource Locator

http://server.org:8080/test/test.html?param1=a&param2=b

http:// – протокол

server.org – доменное имя сервера

8080 – TCP порт

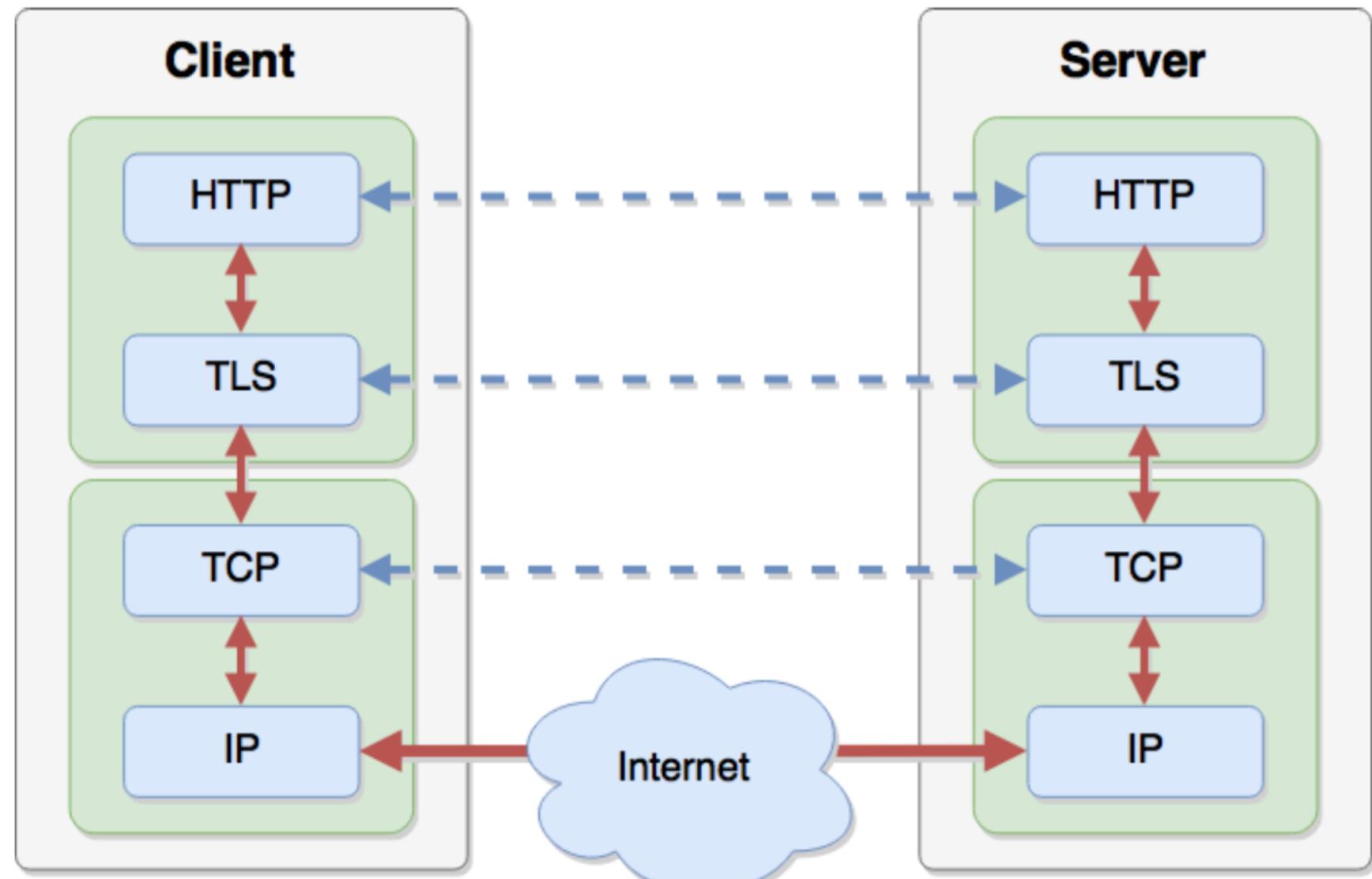
/test/test.html – путь к запрашиваему ресурсу

param1=a param2=b – параметры запроса

# HTTP запрос

http://server.org:8080/test/test.html?param1=a&param2=b

- Браузер анализирует введенный URL и извлекает доменное имя
- Браузер идет в DNS сервер и преобразовывает домен в ip-адрес
- Устанавливает TCP соединение по ip-адресу и порту
- Формирует HTTP запрос
- Отправляет его по установленному TCP соединению на сервер
- Ждет и получает HTTP ответ от сервера
- Закрывает TCP соединение
- Парсит ответ и отображает его на экране



#013

# HTTP/1.1 запрос

http://server.org:8080/test/test.html?param1=a&param2=b

```
GET /test/test.html?param1=a&param2=b HTTP/1.1
```

```
Host: server.org:8080
```

```
Accept: text/html,application/xhtml+xml
```

```
Accept-Encoding: gzip,deflate
```

```
Cache-Control: max-age=0
```

```
Connection: keep-alive
```

```
User-Agent: Mozilla/5.0 Firefox/39.0
```

- строка запроса

- метод

- URL

- документа

- версия HTTP

- заголовки

- тело запроса (\*)

# HTTP/1.1 ответ

http://server.org:8080/test/test.html?param1=a&param2=b

```
HTTP/1.1 404
Server: nginx/1.5.7
Date: Mon, 23 Mar 2020 15:00:00 GMT
Content-Type: text/html; charset=utf-8
Connection: close

<HTML><HEAD>....</HEAD>
....
<BODY>
<div>No such page: 404</div>
</BODY>
</HTML>
```

- строка ответа
  - версия HTTP
  - код ответа
- заголовки
- тело ответа (\*)

# HTTP методы

- **GET** – получение документа
- **HEAD** – получение только заголовков
- **POST** – отправка данных на сервер
- PUT – отправка данных на сервер(\*)
- DELETE – удаление документа(\*)
- PATCH, OPTIONS, TRACE....

# HTTP коды ответа

- **1xx** – информационные
- **2xx** – успешное выполнение
- **3xx** – перенаправления и редиректы
- **4xx** – ошибка на стороне клиента
- **5xx** – ошибка на стороне сервера

# HTTP коды ответа

- **200 OK** – запрос успешно выполнен
- **204 No Content** – запрос успешно выполнен, ресурс пуст
- **301 Moved Permanently** – ресурс сменил URL
- **302 Found** – повторить запрос к ресурсу по другому URL
- **304 Not Modified** – документ не изменился, использовать кэш

# HTTP коды ответа

- **400 Bad Request** – запрос неверно сформирован
- **401 Unauthorized** – требуется авторизация
- **403 Forbidden** – нет доступа (неверная авторизация)
- **404 Not Found** – ресурс не найден
- **500 Internal Server Error** – неожиданная ошибка сервера
- **504 Gateway Timeout** – проксируемый сервер не отвечает

# HTTP заголовки (общие)

- **Content-Type** – MIME тип ресурса
- **Content-Length** – длина(размер) ресурса
- **Content-Encoding** – кодирование ресурса (например gzip сжатие)
- **Connection** – управление соединением (например keep-alive)

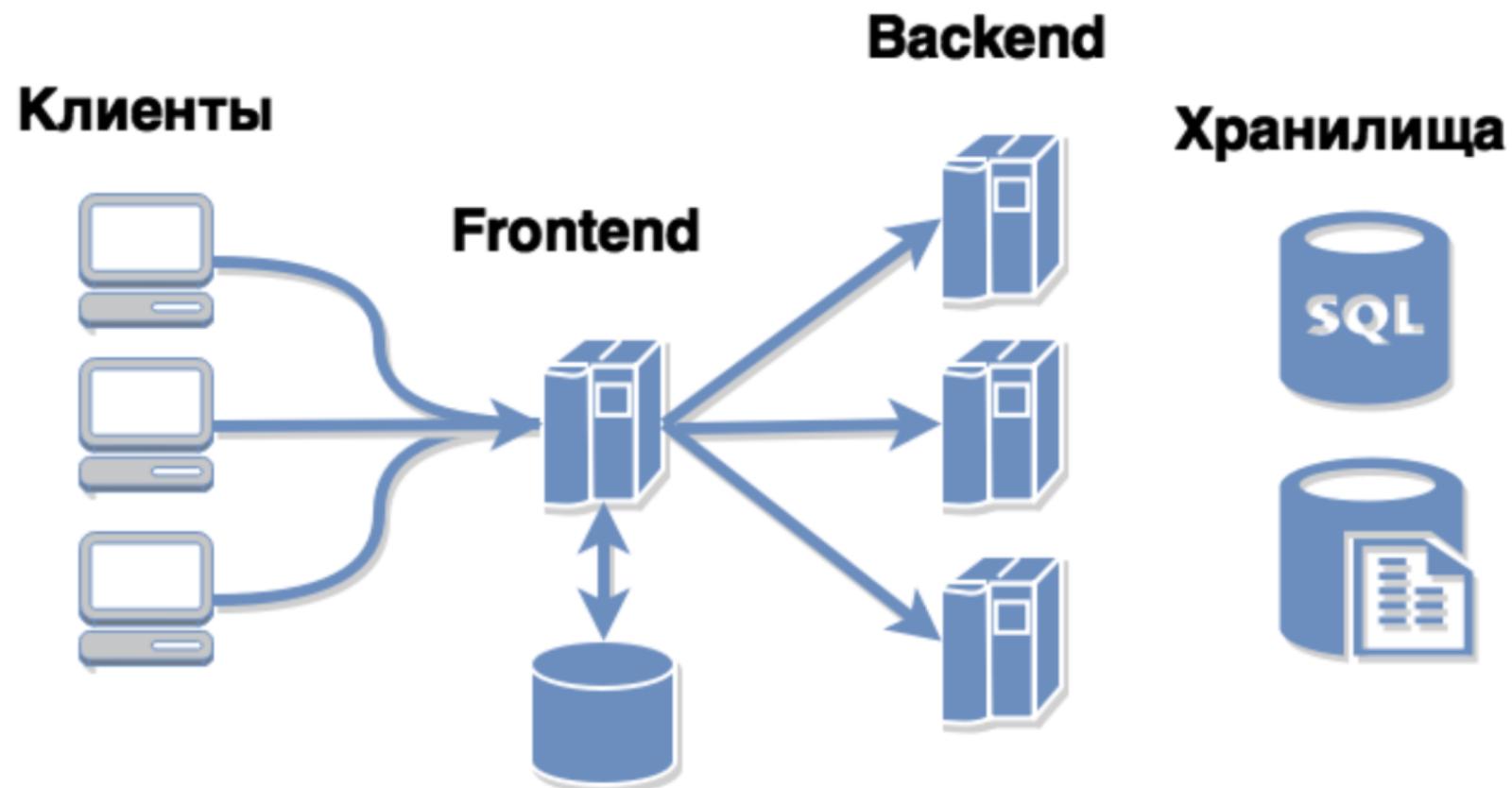
# HTTP заголовки запросов

- **Authorization** – авторизация (например логин/пароль)
- **Cookie** – передача текущей авторизационной сессии на сервер
- **Referrer** – URL предыдущего документа (обычно после 301, 302)
- **User-Agent** – описание веб клиента (например имя/версия браузера)
- **Accept** – согласование поддерживаемого содержимого (text/html)

# HTTP заголовки ответов

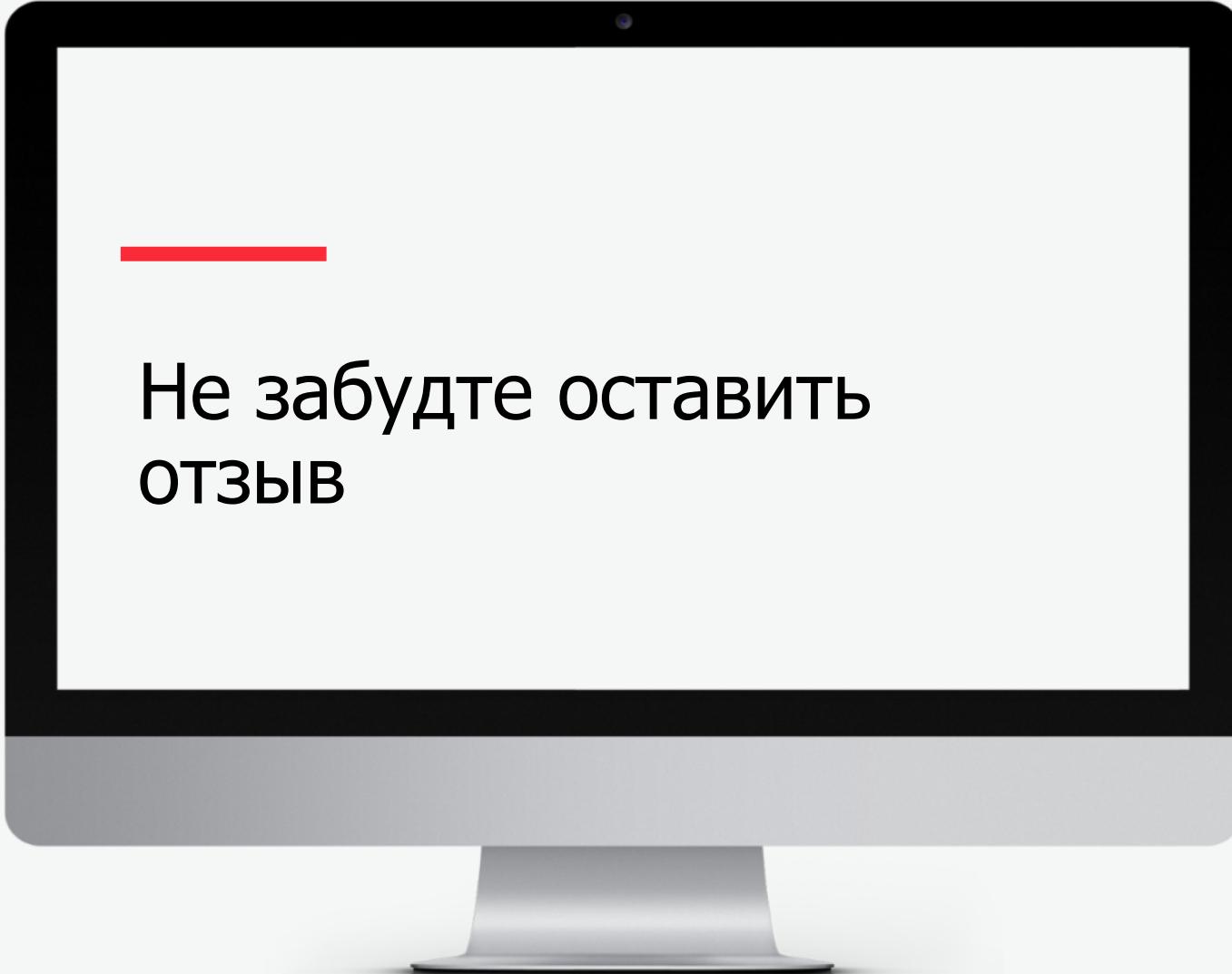
- **Location** – новый URL ресурса при перенаправлении (301, 302)
- **Set-Cookie** – установка сессии в клиент (в веб браузер)
- **Last-Modified** – дата последнего изменения ресурса
- **Date** – дата на сервере (в основном для согласования кеширования)
- **Server** – описание веб-сервера (обычно название/версия)

# Трехзвенная архитектура





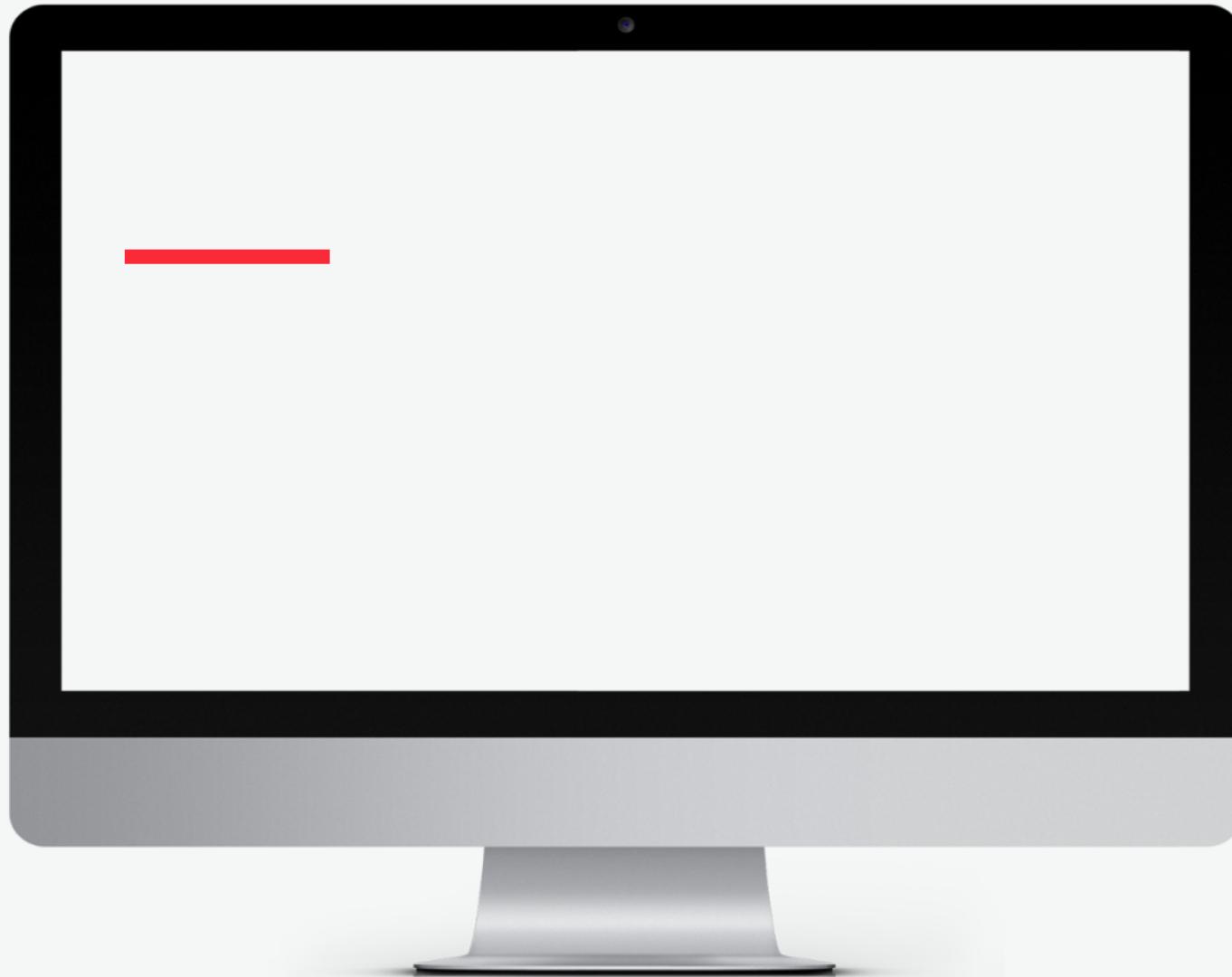
#



—

Не забудте оставить  
отзыв





**СПАСИБО  
ЗА ВНИМАНИЕ**

