

ГЕНЕРИРОВАНИЕ ФИКТИВНЫХ ДАННЫХ ПРИ ПОМОЩИ FAKER.JS

Для тестирования приложений, использующих определенные наборы данных, требуется формирование наборов реальных или фиктивных данных.

Фиктивные данные – это данные, которые содержат информацию, не соответствующую действительности.

Для удобства тестирования веб-интерфейса используется библиотека `Faker.js`.

`Faker` – это библиотека, которая позволяет генерировать случайные данные. С ее помощью можно заполнить таблицы в базе данных, построить корректные XML-документы, сформировать JSON-ответы для REST.

Особенности:

- наличие множества групп форматированных данных;
- поддержка возврата локализованных данных, отличных от `en_US`;
- возможность работы с готовыми наборами данных;
- возможность создавать собственные шаблоны данных
- возможность загрузки `faker.js` из CDN;
- наличие обертки для работы с CLI.

Подключить библиотеку возможно несколькими путями. Первое, используя «Node.js», что предоставляет возможность взаимодействия с консолью и браузером. Для работы через «Node» необходимо в консоль ввести следующие команды:

```
npm init
npm install faker -s
nodeapp
```

В файле `app.js`. (рисунок 1), приводится код инициализации библиотеки. После этого останется создать переменные, которым и присваиваются значения из библиотеки.



```
//npm init
//npm install faker -s
//node app
var faker = require('faker');
var randomName = faker.name.findName();

var http = require('http');
const PORT = 3000;
```

Рисунок 1– Работа с Node.js

Второе, подключение через сетевую инфраструктуру «CDN». Для этого необходимо подключить ссылку (рисунок 2) в скрипте JavaScript. Документация по работе с библиотекой Faker находится по адресу «<https://github.com/marak/Faker.js>». Библиотека располагает богатым набором методов для генерации случайных данных. Например, она содержит набор следующих полей: name, address, phone, internet, company, image, date, random и т.д. Каждый из пунктов содержит свои подпункты, которые и являются методами извлекаемых данных.

```
<!-- FAKER.JS -->  
<script src="https://cdnjs.cloudflare.com/ajax/libs/Faker/3.1.0/faker.js"></script>
```

Рисунок 2 – Подключение через CDN

Для демонстрации возможности «Faker.js» создается форма вида (рисунок 3). После чего необходимо написать код, обращение к элементам формы может происходить как при помощи JS, так и при помощи «jQuery».



Рисунок 3– Форма с использованием библиотеки «Faker.js»

На рисунке 4 представлена функция генерации псевдослучайных данных, для формы, где значениям блоков присваиваются данные библиотеки, после чего данная функция выполняется при нажатии на кнопку «GenerateNew».

Данная форма является тестовым объектом, содержащим несколько полей, как текстовых, так и графических, в блоке «locality» демонстрируется работа локализации библиотеки, в результате чего генерация данных меняется относительно выбранного языка (рисунок 5).

```
function generateName () {
  var email= "email: " + faker.internet.email();
  var jobTitle = "jobTitle: " + faker.name.jobTitle();
  var fullName = faker.name.findName();
  var phoneNumber = "phone: " + faker.phone.phoneNumber();

  var background = 'url("' + faker.image.nature() + '")';
  $('.cardheader').css({'background': background,
    'background-size': 'cover', 'height': '135px'});

  document.getElementById('name').innerHTML=email;
  document.getElementById('fullName').innerHTML = fullName;
  document.getElementById('jobTitle').innerHTML = jobTitle;
  document.getElementById('phoneNumber').innerHTML = phoneNumber;

  $('#imageAvatar').attr("src", faker.internet.avatar());
}
```

Рисунок 4 – Демонстрация функции генерации данных

Георгий Журавлев
 email: mr.human81@hotmail.com
 jobTitle: Corporate Security Representative
 phone: (958)244-05-61

Locality
 Russian ▼

Generate New

Рисунок 5 – Демонстрация функции локализации данных формы

Создание формы и вывод значений демонстрирует лишь малый аспект задач, решаемых при помощи faker.js. Например, существует возможность работы при помощи РНР. Данная библиотека подойдет для большинства задач, связанных с генерацией случайных данных, таких как формы, базы и т.д., а функционал библиотеки способен, при создании своих поставщиков ихэлперов. Данная библиотека отлично подходит для тестирования и разработки качественных интерфейсов или просто структур, включающих данные.

ЛИТЕРАТУРА

1. Faker.js [Электронный ресурс] / github.com. – 2019. – Режим доступа: <https://github.com/marak/Faker.js>. – Дата доступа: 11.04.2019.
2. Генерирование фейковых данных для вашего JavaScript-приложения [Электронный ресурс] / habr.com. – 2019. – Режим доступа: <https://habr.com/ru/post/248999/>. – Дата доступа: 11.04.2019.