



Разработка приложения

Занятие 2

Дмитрий Бронских
Fullstack-developer в Телеком Нейва



Дмитрий Бронских

Fullstack-developer в Телеком Нейва

О спикере

- Занимался разработкой информационных систем для сети ресторанов быстрого питания
- Работал над автоматизированной системой расчетов для телеком-провайдера

План занятия

1. О работе с сервером
2. О vue.js
3. Практическая часть
4. Timeweb
5. Домашнее задание

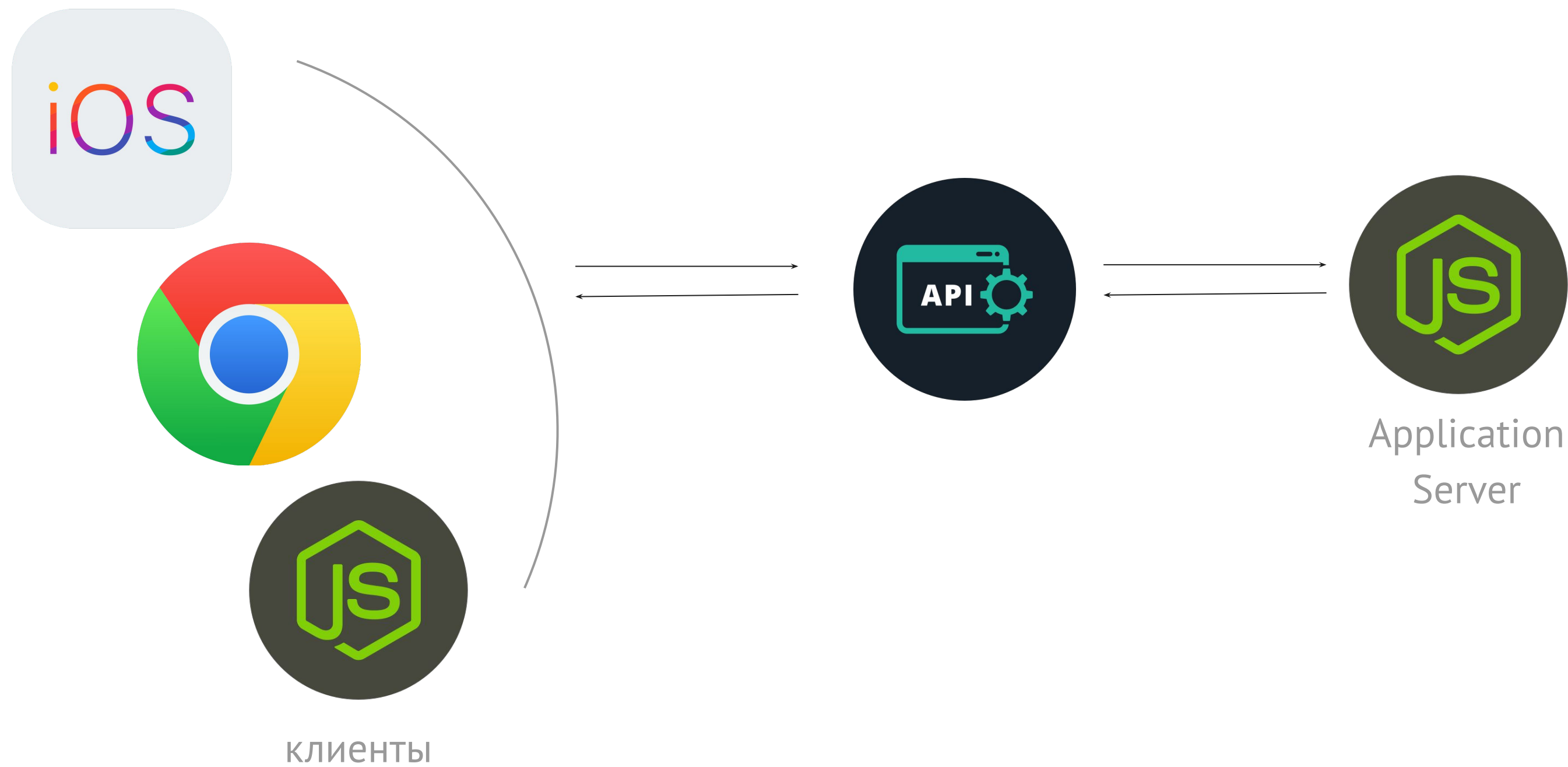
О работе с сервером

1



API

API - Application Programming Interface



API

Обзор [API](#) для приложения

baseURL: <https://netology-api-marvel.herokuapp.com>

метод	url	действие	пример
GET	/characters	получить всех персонажей	url
GET	/characters/:id	получить персонажа по id	url

JSON

JSON (JavaScript Object Notation) — простой формат обмена данными, удобный для чтения и написания как человеком, так и компьютером. Он основан на подмножестве языка программирования JavaScript, определенного в стандарте ECMA-262 3rd Edition - December 1999.



Структура объекта "character"

```
{
  id: 1234567,
  name: "string",
  description: "string",
  modified: "string",
  thumbnail: "string",
  comics:
  [
    {
      resourceURI: "string",
      name: "string"
    }
  ],
  urls:
  [
    {
      type: "string",
      url: "string"
    }
  ]
}
```



O vue.js

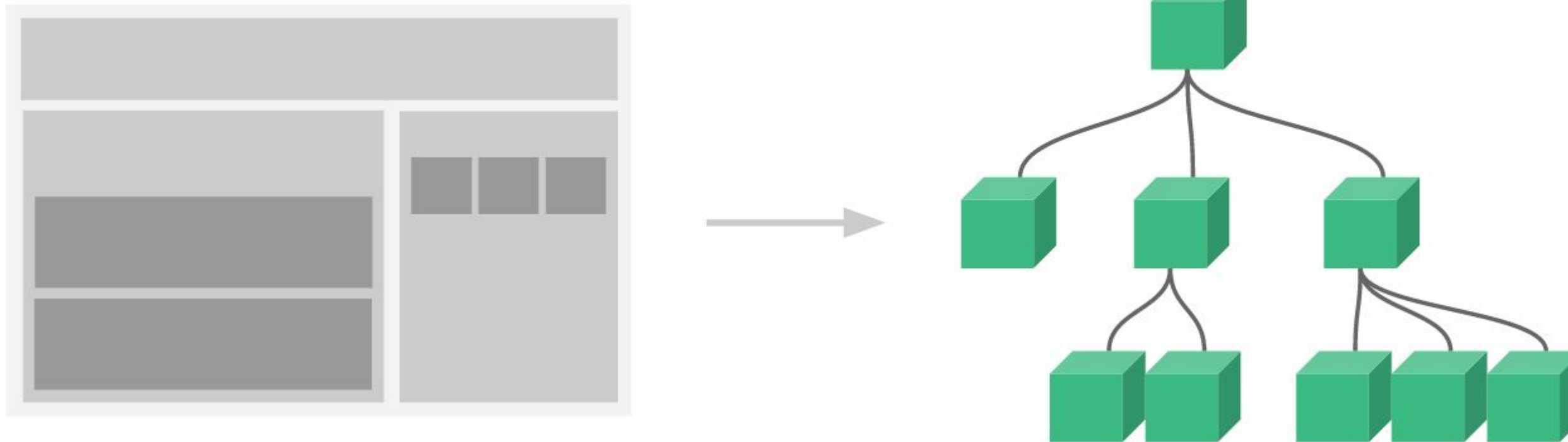
2



vue.js components

Важной концепцией Vue являются компоненты. Эта абстракция позволяет собирать большие приложения из маленьких «кусочков». Они представляют собой пригодные к повторному использованию объекты.

Если подумать, почти любой интерфейс можно представить как дерево компонентов:



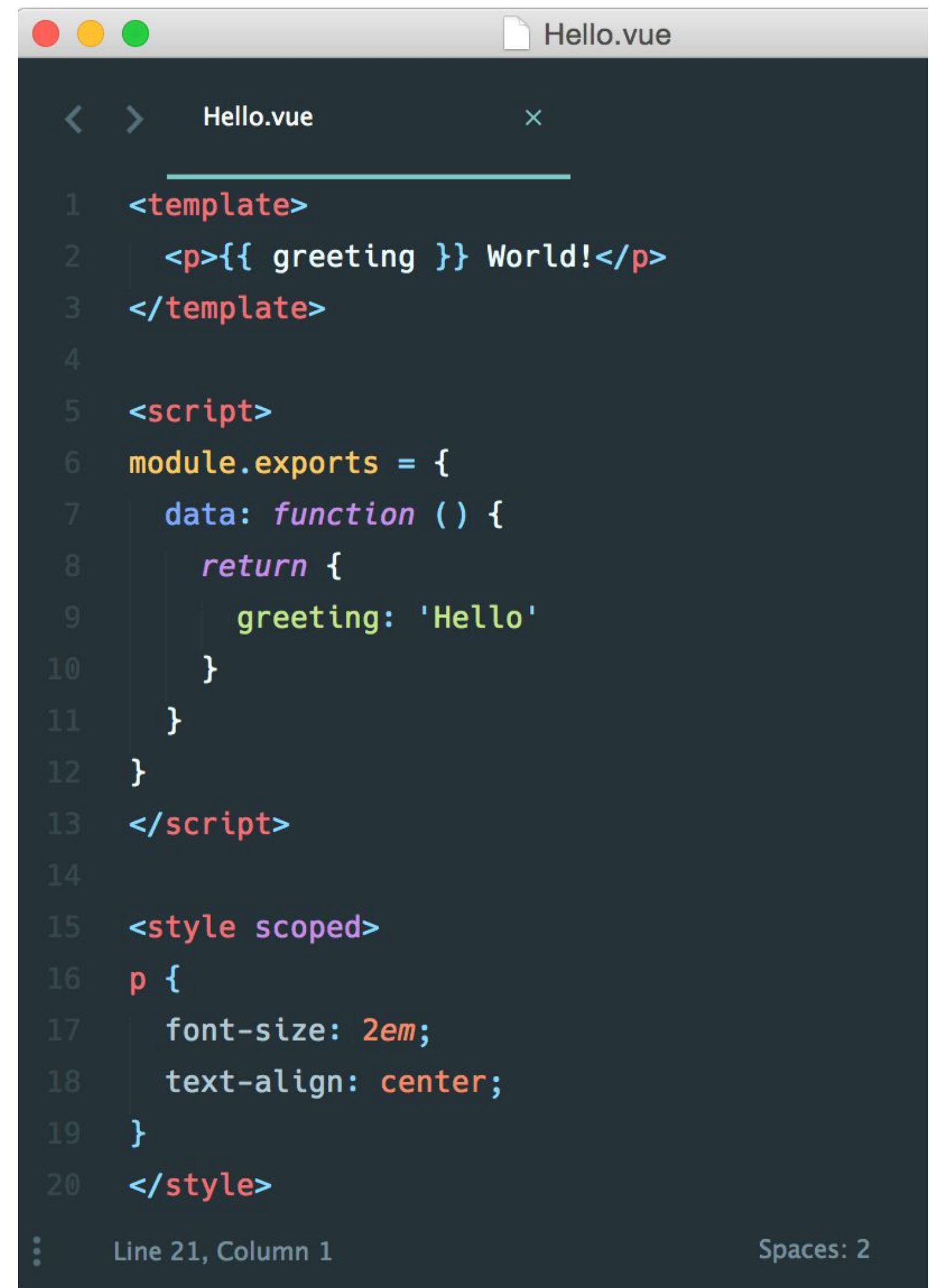
источник: ru.vuejs.org

vue.js однофайловые компоненты

Внутри компонента, его шаблон, логика и стили неразрывно связаны между собой, что позволяет сделать компонент более сплочённым и удобным в поддержке.

```
<!-- my-component.vue -->
<template>шаблон</template>
<script>логика</script>
<style>стили</style>
```

источник: ru.vuejs.org



The screenshot shows a code editor window titled 'Hello.vue'. The code is written in a dark theme and includes a template, script, and style section. The template contains a paragraph with a greeting. The script section defines a data object with a 'greeting' property set to 'Hello'. The style section uses a scoped CSS rule to set the font size to 2em and text-align to center for the paragraph.

```
1 <template>
2   <p>{{ greeting }} World!</p>
3 </template>
4
5 <script>
6   module.exports = {
7     data: function () {
8       return {
9         greeting: 'Hello'
10      }
11    }
12  }
13 </script>
14
15 <style scoped>
16   p {
17     font-size: 2em;
18     text-align: center;
19   }
20 </style>
```

Line 21, Column 1 Spaces: 2

Практика

Переходим на github.

1. развернем базовый шаблон
2. настроим компоненты
3. получим данные с API
4. свяжем компоненты с данными

3

MacBook Pro

Timeweb

4



Timeweb

Timeweb – это российский хостинг, отличающийся стабильным функционированием, безопасностью, быстродействием и множеством предложений для размещений сайтов.



Firebase Hosting

[Firebase Hosting](#) – это полностью управляемая служба хостинга для статического и динамического контента, а также микросервисов. Сервис поддерживается SSD-хранилищем и глобальной CDN (сетью доставки контента). SSL в базовой конфигурации, что гарантирует безопасность доставки контента.



Домашнее задание

5



Домашнее задание

Давайте посмотрим ваше [домашнее задание](#).

- Первые 5 работ будут оценены экспертом - в личном кабинете будет дана развернутая обратная связь
- Любые вопросы по решению задач задавайте в Telegram
- В поле для сдачи работы прикрепите архив в формате **.zip**

**Задавайте вопросы
и пишите отзыв
о лекции!**



[Телеграм чат курса](#)