

19. Учебный проект: большие переменные (Часть 2)

Рабочая ветка `module6-task2`

Задача

В этом задании мы подключим библиотеки, которые помогут нам эффективно работать с датой.

1. Установите из npm и подключите библиотеку `flatpickr`.
2. Импортируйте в компонент с формой редактирования стили для `flatpickr` из `node_modules`.
3. Установите из npm пакеты `style-loader` и `css-loader`, они потребуются для сборки стилей прямо из JavaScript. В файле конфигурации Webpack добавьте поле `module` с ключом `rules` и опишите в последнем объект с правилами:

```
module.exports = {  
  /* ... */  
  module: {  
    rules: [  
      {  
        test: /\.css$/i,  
        use: ['style-loader', 'css-  
loader']  
      }  
    ]  
  }  
}
```

```
    ]  
  }  
}
```

4. Настройте библиотеку `flatpickr` так, чтобы выбор даты в форме редактирования осуществлялся с её помощью. Формат даты указан в техническом задании.

5. Установите из `npm` и подключите библиотеку `moment`.

6. `moment` — тяжеловесная библиотека на все случаи жизни. Но нам всё не нужно, поэтому мы избавимся от лишнего. Для этого установите из `npm` пакет `moment-locales-webpack-plugin` (специальный плагин, который удалит при сборке ненужные локали) и подключите его в файле конфигурации Webpack:

```
const MomentLocalesPlugin = require(`moment-  
locales-webpack-plugin`);
```

7. Затем добавьте в объект настроек Webpack в самом конце ключ `plugins` вот так:

```
module.exports = {  
  /*...*/  
  plugins: [  
    new MomentLocalesPlugin()  
  ]  
}
```

8. Отобразите с помощью библиотеки `moment` дату выполнения в компоненте задачи в заданном формате. Формат указан в техническом задании.

21. Личный проект: большие переменны (Часть 2)

Задача

В этом задании мы подключим библиотеки, которые помогут нам эффективно работать с датой.

Киноман

1. Установите из npm и подключите библиотеку `moment`.
2. `moment` — тяжеловесная библиотека на все случаи жизни. Но нам всё не нужно, поэтому мы избавимся от лишнего. Для этого установите из npm пакет `moment-locales-webpack-plugin` (специальный плагин, который удалит при сборке ненужные локали) и подключите его в файле конфигурации Webpack:

```
const MomentLocalesPlugin = require(`moment-locales-webpack-plugin`);
```

3. Затем добавьте в объект настроек Webpack в самом конце ключ `plugins` вот так:

```
module.exports = {  
  /*...*/  
  plugins: [  
    new MomentLocalesPlugin()  
  ]  
}
```

4. Отобразите с помощью библиотеки `moment` продолжительность (будет приходит с сервера в минутах) и дату релиза фильма, а также дату у комментария в заданных форматах. Форматы указаны в техническом задании.

Обратите внимание, вы можете сразу выполнить дополнительную часть технического задания, и вывести дату комментария в человеческом формате на английском языке, используя возможности библиотеки `moment`.

Bigtrip

1. Установите из npm и подключите библиотеку `flatpickr`.
2. Импортируйте в компонент с формой редактирования стили для `flatpickr` из `node_modules`.
3. Установите из npm пакеты `style-loader` и `css-loader`, они потребуются для сборки стилей прямо из JavaScript. В файле конфигурации Webpack добавьте поле `module` с ключом `rules` и опишите в последнем объекте с правилами:

```
module.exports = {  
  /*...*/  
  module: {  
    rules: [  
      {  
        test: /\.css$/i,  
        use: ['style-loader', 'css-  
loader']  
      }  
    ]  
  }  
}
```

4. Настройте библиотеку `flatpickr` так, чтобы выбор даты в форме редактирования осуществлялся с её помощью. Формат даты указан в техническом задании.

5. Установите из npm и подключите библиотеку `moment`.

6. `moment` — тяжеловесная библиотека на все случаи жизни. Но нам всё не нужно, поэтому мы избавимся от лишнего. Для этого установите из npm пакет `moment-locales-webpack-plugin` (специальный плагин, который удалит при сборке ненужные локали) и подключите его в файле конфигурации Webpack:

```
const MomentLocalesPlugin = require(`moment-locales-webpack-plugin`);
```

7. Затем добавьте в объект настроек Webpack в самом конце ключ `plugins` вот так:

```
module.exports = {  
  /*...*/  
  plugins: [  
    new MomentLocalesPlugin()  
  ]  
}
```

8. Посчитайте и отобразите в заданном формате с помощью библиотеки `moment` продолжительность каждой точки (должна высчитываться по разнице дат начала и конца). Формат указан в техническом задании.