

PRETTIER

Инструмент для форматирования кода

ЧТО ЗА ЗВЕРЬ И С ЧЕМ ЕГО ЕДЯТ?

- Единый стиль
- Любовь к прекрасному
- Ускорение разработки

ОБЛАСТЬ ПРИМЕНЕНИЯ



ES2017
TypeScript
JSX
JSON
Flow
Vue



GraphQL
GraphQL Schemas



CommonMark
Github Flavored Markdown



CSS3+
SCSS
Less
styled-components 🔥



Work In Progress
Swift
Elm
Ruby
Python
PHP
Java
PostgreSQL

ПРИМЕР

eslint-disable-line max-len

```
test('should parse nds from `В Т.Ч. НДС 18%, 232` cases', () => {  
  expect(getNdsInfo('В Т.Ч. НДС 18%, 232')).toEqual(  
    // eslint-disable-next-line max-len  
    { prepend: '', ndsString: 'В Т.Ч. НДС 18%, 232', percentString: '18%', sumSt  
  });  
});
```

ПРИМЕР

eslint-disable-line max-len

```
test('should parse nds from `В Т.Ч. НДС 18%, 232` cases', () => {  
  expect(getNdsInfo('В Т.Ч. НДС 18%, 232')).toEqual({  
    prepend: '',  
    ndsString: 'В Т.Ч. НДС 18%, 232',  
    percentString: '18%',  
    sumString: '232',  
    append: '',  
    mode: 2  
  });  
});
```

НА ПРАКТИКЕ



10 сезон 2 серия

ARUI-FEATER

```
checkbox.jsx (Working Tree)

94 .....disabled: this.props.disabled,
95 .....checked: checked || this.props.indetermina
96 .....indeterminate: this.props.indeterminate,
97 .....focused: this.state.focused,
98 .....hovered: this.state.hovered,
99 .....width: this.props.type === 'button' ? this
100 - .....}}
101 - .....htmlFor={this.props.id}
102 - .....onFocus={this.handleFocus}
103 - .....onBlur={this.handleBlur}
104 - .....onMouseEnter={this.handleMouseEnter}
105 - .....onMouseLeave={this.handleMouseLeave}
106 - .....ref={ (root) => { this.root = root; } }

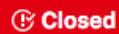
107 .....>
108 - .....{
109 - .....this.props.type === 'button'
110 - .....? this.renderButtonCheckbox(cn, checked)
111 - .....: this.renderNormalCheckbox(cn, checked)
112 - .....}
113 .....</label>
114 .....);
115 .....}
116 .....
117 .....renderNormalCheckbox(cn, checked) ·{
118 .....return [
119 - .....<span
120 - .....className={cn('box')}
121 - .....key='box'
122 - .....>
123 .....<input
124 - .....className={cn('control')}
125 - .....type='checkbox'

91 .....disabled: this.props.disabled,
92 .....checked: checked || this.props.indetermina
93 .....indeterminate: this.props.indeterminate,
94 .....focused: this.state.focused,
95 .....hovered: this.state.hovered,
96 .....width: this.props.type === 'button' ? this
97 + .....}}
98 + .....htmlFor={this.props.id}
99 + .....onFocus={this.handleFocus}
100 + .....onBlur={this.handleBlur}
101 + .....onMouseEnter={this.handleMouseEnter}
102 + .....onMouseLeave={this.handleMouseLeave}
103 + .....ref={root => {
104 + .....this.root = root;
105 + .....}}
106 .....>
107 + .....{this.props.type === 'button'
108 .....? this.renderButtonCheckbox(cn, checked,
109 + .....: this.renderNormalCheckbox(cn, checked)}
110 .....</label>
111 .....);
112 .....}
113 .....
114 .....renderNormalCheckbox(cn, checked) ·{
115 .....return [
116 + .....<span className={cn('box')} key='box'>
117 .....<input
118 + .....className={cn('control')}
119 + .....type='checkbox'
```

КОНФИГУРАЦИЯ .prettierrc.js

```
module.exports = {  
  printWidth: 120,  
  semi: true,  
  singleQuote: true,  
  bracketSpacing: true,  
  trailingComma: 'none'  
};
```


Resist adding configuration #40



Closed

gaearon opened this issue on 10 Jan 2017 · 38 comments



gaearon commented on 10 Jan 2017



You're going to be under a lot of pressure to make X, Y, and Z, configurable. Configuration is a perfect example of [tragedy of the commons](#).

Everybody wants to add just a tiny piece of configuration to satisfy their use case. As a result we end up with inconsistently designed tools, with unexpected combinations that don't work correctly, with bugs and regressions that, when fixed in one configuration, break something else with another configuration.

The requests for configuration are vocal. The requests to stay simple are often silent.

I just wanted to take a moment to say there are some of us who actually appreciate tools with limited scope that don't attempt to solve everybody's use case, and instead do a specific thing well.



280



6



2



25



40

ПОПЫТКА РЕШИТЬ ПРОБЛЕМУ

Code ➡ prettier ➡ eslint --fix ➡ Formatted Code ✨

НЕ ВСЁ ТАК ПРОСТО

```
function createPayment() { // My favorite comment style
  fetch(CREATE_PAYMENT_URL); // Second comment
}
```

```
function createPayment() {
  // My favorite comment style
  fetch(CREATE_PAYMENT_URL); // Second comment
}
```

НЕ ВСЁ ТАК ПРОСТО

```
function totalMoney() {  
  return ((income * tax) - credits + debits);  
}
```

```
function totalMoney() {  
  return income * tax - credits + debits;  
}
```

РЕШЕНИЕ

```
> npm install --save prettier eslint-plugin-prettier
```






```
// eslinttrc.json
{
  "plugins": ["prettier"],
  "rules": {
    "prettier/prettier": "error"
  }
}
```

PRETTIER ~~FOV...~~ ПЛОХО ФОРМАТИРУЕТ

```
let foo = a => a
  ? convert(resume(dispatch)) (favorites)
  : rollback(resume(dispatch))
```

```
let foo = a =>
  a ? convert(resume(dispatch)) (favorites) : rollback(resume(dispatch))
```

PRETTIER ХОРОШО ФОРМАТИРУЕТ 🦄








 knotri 19.10.17 в 13:16    +1 

Перешел на **prettier** с standardJs — Это просто божественно — да, в нем мало настроек, по началу это бесило (например нельзя в реакте сделать кавычки на пропсах одинарными) — но сейчас я понимаю что вообще больше руками не форматирую код — все делается автоматически за меня.

Я не представляю себе ситуацию чтоб я вернулся к ручному форматированию.

Но у преттиера есть минус — он немного сырой и кое-где ведет себя странно, с теми же промисами

[Ответить](#)

 Staltec 20.10.17 в 08:08      +2 

У нас практика показала, что если **prettier** что то форматирует странно, это значит изначально код написан сильно криво. Я своих ребят ещё какое-то время отучал от фраз типа «это претир так сформатировал». Писать надо так чтобы форматировалось нормально. Выносить вычисления в константы, не совмещать вычисления с возвратами, особенно тринарники. И всё сразу нормально форматироваться начинает.

[Ответить](#)

PRETTIER-IGNORE

```
// prettier-ignore
const matrix = [
  1, 2, 3,      1,      4, 9, 1, 1, 6, 7,
  4,      5,      0, 2,      5,      2,
  6,      8,      9,      3, 6, 3, 2, 3,
  6,      8,      9,      3, 6,      4,
  6, 7, 8, 9,      3, 6, 7, 8, 5,
];

// without-ignore
const matrix = [
  1,
  2,
  3,
  ...
  5
];
```


Prettier - форматирование, которое мы заслужили

- *Jason Statham*

СПАСИБО ЗА ВНИМАНИЕ

twitter.com/luchanso

hr.alfabank.ru