



Программирование без стресса

/// ключевые принципы



ЧИСТЫЙ КОД

СОЗДАНИЕ, АНАЛИЗ
И РЕФАКТОРИНГ



РОБЕРТ МАРТИН



ЧИСТАЯ АРХИТЕКТУРА

ИСКУССТВО РАЗРАБОТКИ
ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ



РОБЕРТ МАРТИН

Алексей Табула

/// front-end разработчик



- 😊 Веду telegram-канал о front-end технологиях [**TabulaWeb**](#)
- 😊 Занимаюсь менторством [**Getmentor**](#)

ЭЙ, ПРОБЛЕМА, Я
СОБИРАЮСЬ ИДТИ
СПАТЬ.

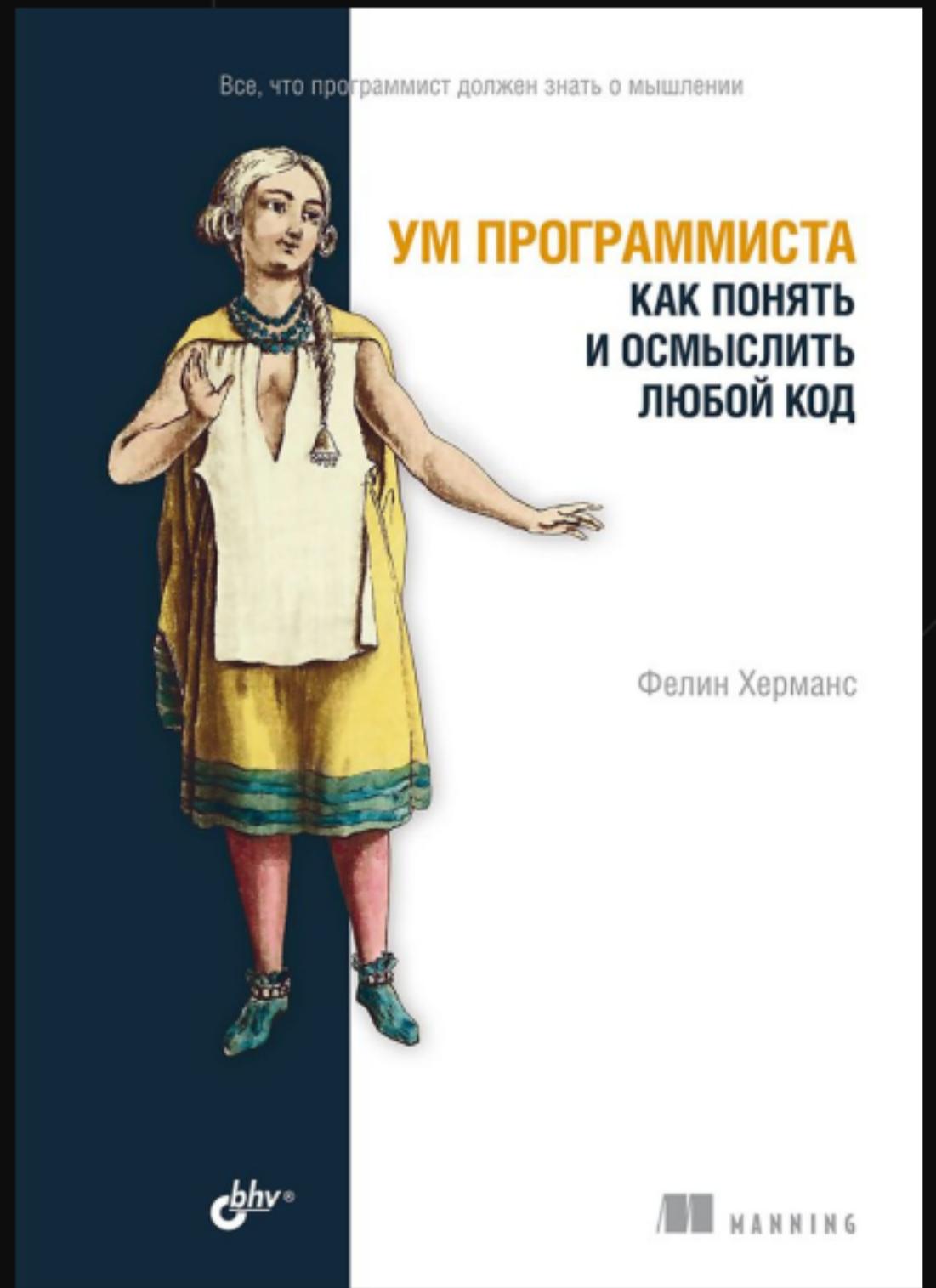
ОК.

ПРОБЛЕМА

```
> 📂 font  
> 📂 img  
> 📂 script  
> 📂 style  
  ⚒ bosses.html  
  ⚒ index.html  
  ⚒ license.html  
  ⚒ line-business.html  
  ⚒ map.html  
  ⚒ work.html
```

```
> 🐾 .husky  
> 📂 .storybook  
> 📂 cypress  
> 📂 public  
└─ 📂 src  
    > 📂 app  
    > 📂 entities  
    > 📂 pages  
    > 📂 shared  
      ⚒ .dependency-cruiser.cjs  
      ⚒ .env.example  
      ⚒ .eslintrc.cjs  
      ⚒ .gitignore  
      ⚒ .prettierignore  
      ⚒ .prettierrc  
      ⚒ .tool-versions  
      ⚒ cypress.config.js  
      ⚒ dependency-graph-preview.svg  
      ⚒ index.html  
      ⚒ jest.config.json  
      ⚒ package-lock.json  
      ⚒ package.json
```





Так, о чём пойдёт речь?

Из анонса про программирование без стресса:

Не разобрались в нюансах работы исходного кода? ”
Ошибка. И причиной многих этих ошибок являются
когнитивные проблемы

Это типа деменции...?

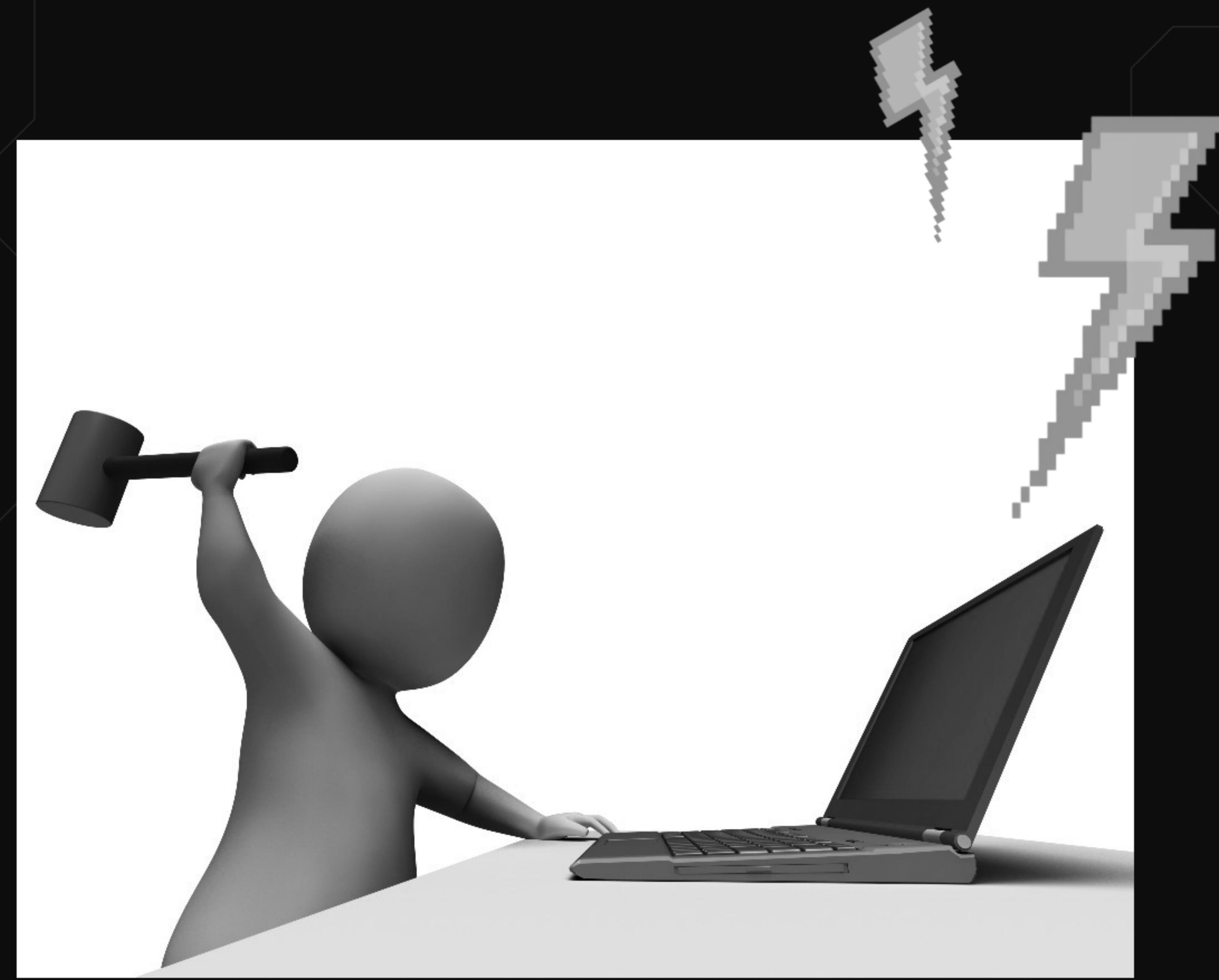


Сегодня поговорим

- ⑪ Что такое когнитивная нагрузка и какие проблемы с ней связаны?

Сегодня поговорим

- ❶ Что такое когнитивная нагрузка и какие проблемы с ней связаны?
- ❷ Как бороться с когнитивной нагрузкой?



Когнитивная нагрузка



Определённый объём информации, которая может уместиться
одновременно в рабочей памяти человека

ЧТО ВИДИТ

• • •



Backend

Frontend

```
&._loader {  
  pointer-events: none;  
}  
  
&._primary {  
  background-color: ${({ theme }) => theme.primary};  
  color: white;  
  
  &:hover {  
    background-color: ${({ theme }) => darken(-0.009, adjustHue(10, theme.primary))};  
    color: ${({ theme }) => darken(-0.009, adjustHue(10, theme.primary))};  
  }  
  
  &:active {  
    background-color: ${({ theme }) => darken(0.12, adjustHue(7, theme.primary))};  
  }  
  
  & svg {  
    fill: ${({ theme }) => theme.card};  
  }  
}  
  
&._secondary {  
  border: 1px solid ${({ theme }) => theme.primary};  
  background-color: initial;  
  
  &:hover {  
    background-color: ${({ theme }) => lighten(0.483, adjustHue(-15, theme.primary))};  
  }  
  
  &:active {  
    background-color: ${({ theme }) => lighten(0.44, adjustHue(-13, theme.primary))};  
  }  
  
  & svg {  
    fill: ${({ theme }) => theme.primary};  
  }  
}
```

Storage API мозга

- ❖ Долговременная память
- ❖ Кратковременная память
- ❖ Рабочая память



Долговременная память

/// Жесткий диск



Performance

```
const start = performance.now();

for (var i = 0; i < 5; i++) {
    setTimeout(() => {
        console.log(i, performance.now() - start);
    }, i * 1000);
}

console.log('finish', performance.now() - start);
```

Кратковременная память

/// Оперативная память

Отличий не ищи,
код тот-же

```
... Performance

const start = performance.now();

for (var i = 0; i < 5; i++) {
    setTimeout(() => {
        console.log(i, performance.now() - start);
    }, i * 1000);
}

console.log('finish', performance.now() - start);
```

Рабочая память

/// Процессор

Отличий не ищи,
код тот-же

```
... Performance

const start = performance.now();

for (var i = 0; i < 5; i++) {
    setTimeout(() => {
        console.log(i, performance.now() - start);
    }, i * 1000);
}

console.log('finish', performance.now() - start);
```

Какие проблемы?

❗ Снижение способности к обучению

Какие проблемы?

- ☒ Снижение способности к обучению
- ☒ Ослабление понимания



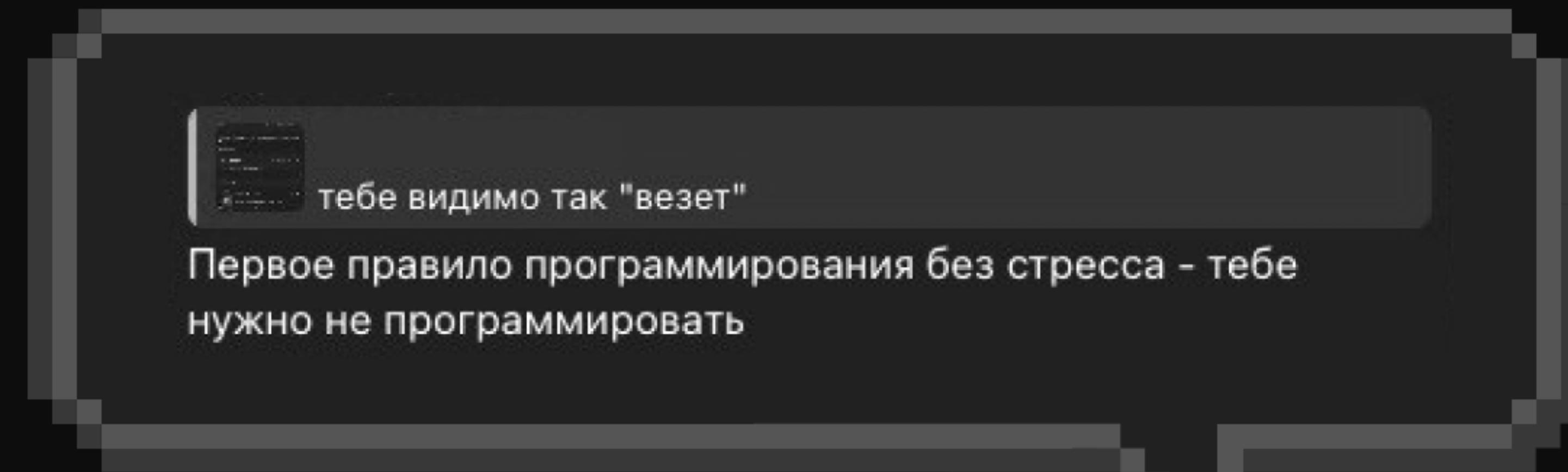
Какие проблемы?

- ❗ Снижение способности к обучению
- ❗ Ослабление понимания
- ❗ Увеличение вероятности ошибок при выполнении задач

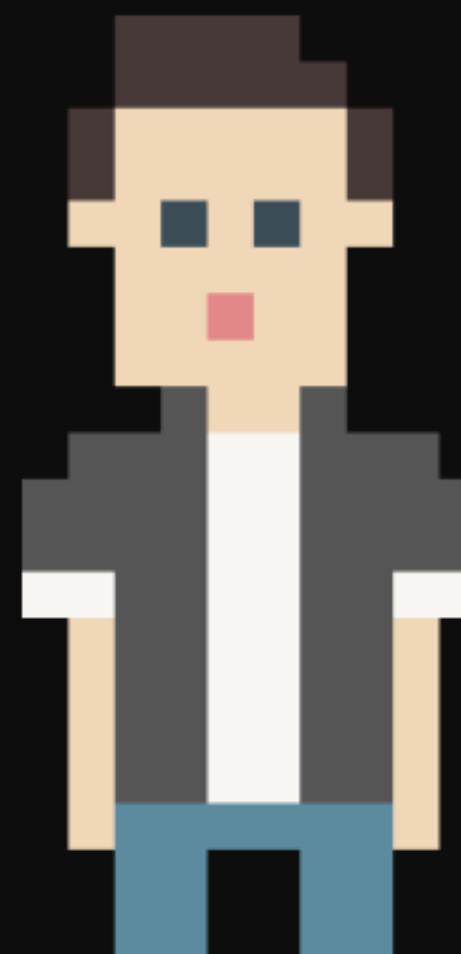




Как будем справляться?



Первое правило программирования без стресса - тебе
нужно не программировать



Как бороться с когнитивной нагрузкой

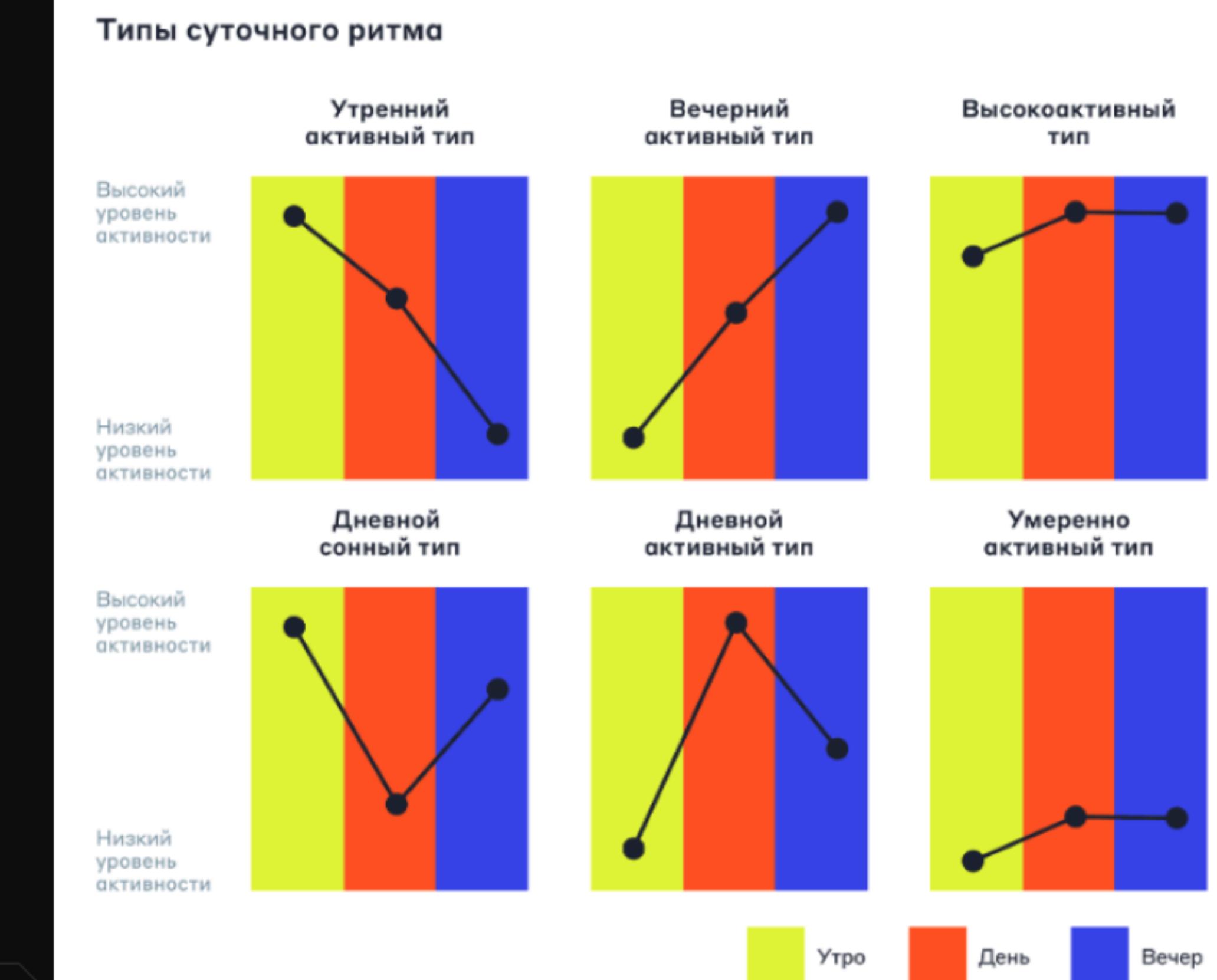
💡 Учитесь у других



Как бороться с когнитивной нагрузкой

💡 Учитесь у других

💡 Учитывайте свои
продуктивные часы



Как бороться с когнитивной нагрузкой

- 💡 Учитесь у других
- 💡 Учитывайте свои продуктивные часы
- 💡 Делайте перерывы



Как бороться с когнитивной нагрузкой

- 💡 Учитесь у других
- 💡 Учитывайте свои продуктивные часы
- 💡 Делайте перерывы
- 💡 Подружитесь со своей IDE



Как бороться с когнитивной нагрузкой

- 💡 Учитесь у других
- 💡 Учитывайте свои продуктивные часы
- 💡 Делайте перерывы
- 💡 Подружитесь со своей IDE
- 💡 Изучите код команды



Как мы понимаем код?

- ① Изучили синтаксис языка

```
export const Status = ({ status }) => {
```

```
}
```

Как мы понимаем код?

- ① Изучили синтаксис языка
- ② Изучили фреймворк

```
export const Status = ({ status }) => {
  const text = useMemo(() => {
    switch (status) {
      case 'duplicate':
        return 'Дубль'
      case 'inprogress':
        return 'В работе'
      case 'done':
        return 'Сделано'
      case 'error':
        return 'Ошибка'
      default:
        return 'статус'
    }
  }, [status])

  /**
   * 
  return (
    <Main className={`_${status}`}>
      <Text>{text}</Text>
    </Main>
  )
}
```

Эксперимент: на сколько ваш код когнитивно сложный



Попробуйте измерить количество путаницы, которую испытывает новый член команды. Если он находится в замешательстве более ~40 минут подряд – вам есть что улучшать.

Чистый код

```
import { s } from '@libs/axios'

export const crtG = async (b) => {
  try {
    const i = await s.post('/groups/add', b)

    return i.data
  } catch (e) {
    return e
  }
}
```





А зачем писать чистый код?

```
import { server } from '@/libs/axios'

export const createGroups = async (body) => {
  try {
    const response = await server.post('/groups/add', body)

    return response.data
  } catch (error) {
    return error
  }
}
```



Вкатываемся в программирование По версии frontend

Я смогу
HTML

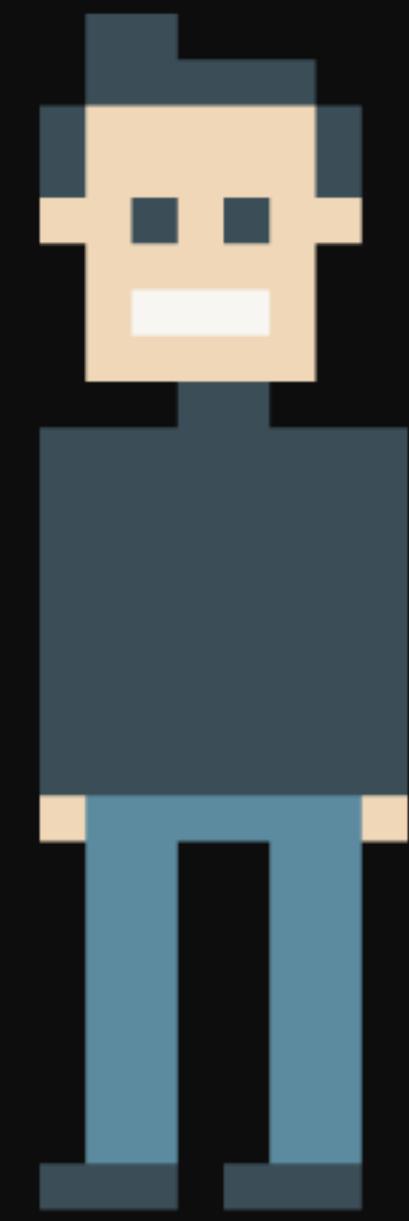
Ещё чуть-чуть
CSS

Уже почти!
JS

Твою ж...

REACT
ANGULAR 2
ANGULAR1
VUE
Enzyme
Jest
Sass
TDD
REST
GraphQL
SSR
Mocha
Karma
Chai
BDD
node
Gulp
TypeScript

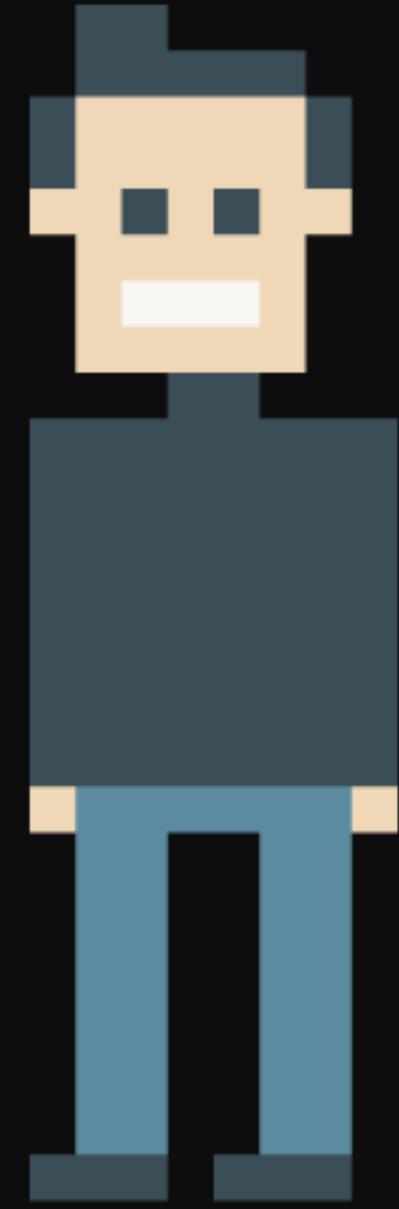
**Человек состоит на 80% из воды,
а из чего состоит программист?**



Человек состоит на 80% из воды, а из чего состоит программист?

 Синтаксис – сложно представить программиста который не знает синтаксиса языка на котором пишет

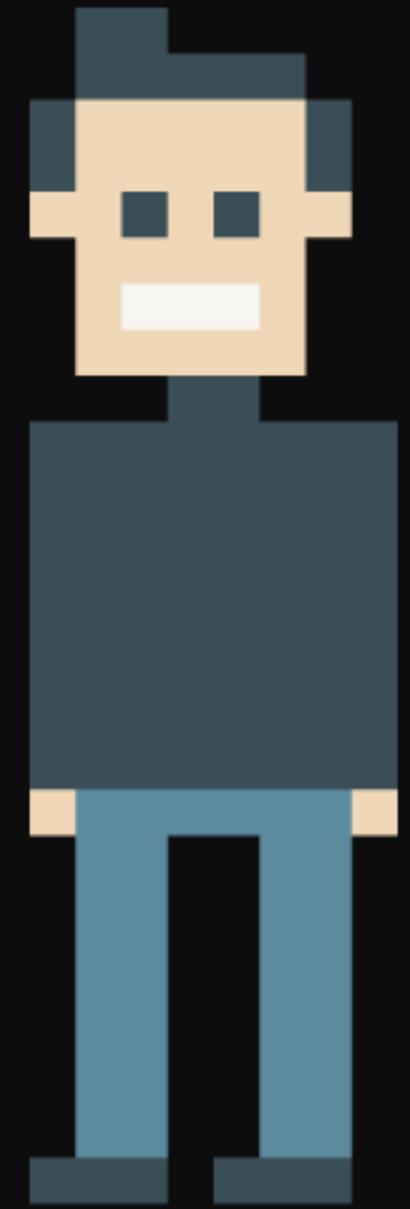
LEVEL
UP



Человек состоит на 80% из воды, а из чего состоит программист?

- 💡 Синтаксис – сложно представить программиста который не знает синтаксиса языка на котором пишет
- 💡 IDE

LEVEL
UP



Человек состоит на 80% из воды, а из чего состоит программист?

- ✍ Синтаксис – сложно представить программиста который не знает синтаксиса языка на котором пишет
- ✍ IDE
- ✍ Фреймворк



Человек состоит на 80% из воды, а из чего состоит программист?

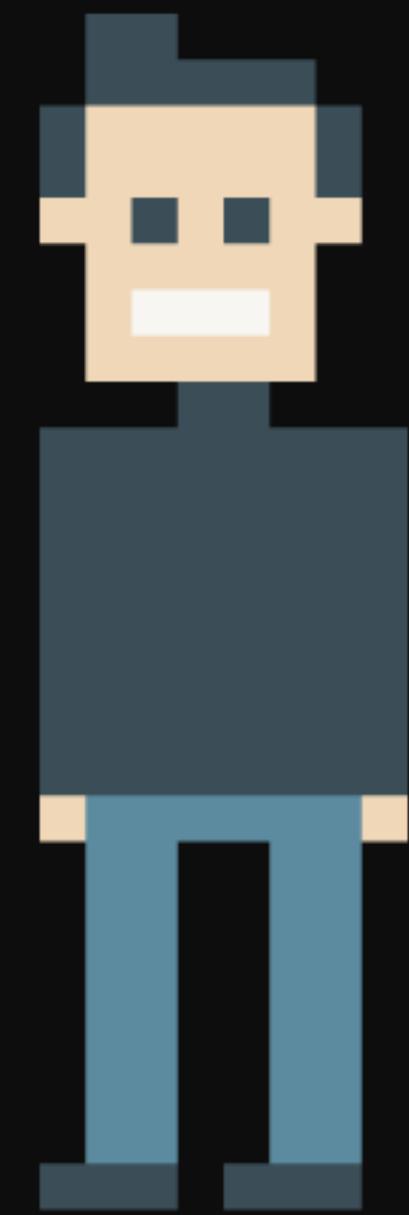
LEVEL
UP

- ✍ Синтаксис – сложно представить программиста который не знает синтаксиса языка на котором пишет
- ✍ IDE
- ✍ Фреймворк
- ✍ Библиотеки



Человек состоит на 80% из воды, а из чего состоит программист?

- ✍ Синтаксис – сложно представить программиста который не знает синтаксиса языка на котором пишет
- ✍ IDE
- ✍ Фреймворк
- ✍ Библиотеки
- ✍ Чистый код

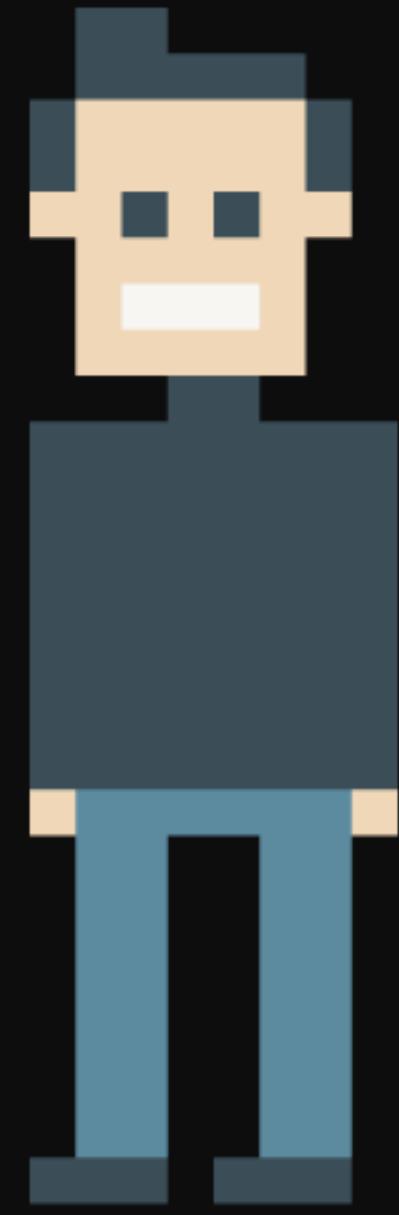


LEVEL
UP

Человек состоит на 80% из воды, а из чего состоит программист?

- ✍ Синтаксис – сложно представить программиста который не знает синтаксиса языка на котором пишет
- ✍ IDE
- ✍ Фреймворк
- ✍ Библиотеки
- ✍ Чистый код
- ✍ Паттерны

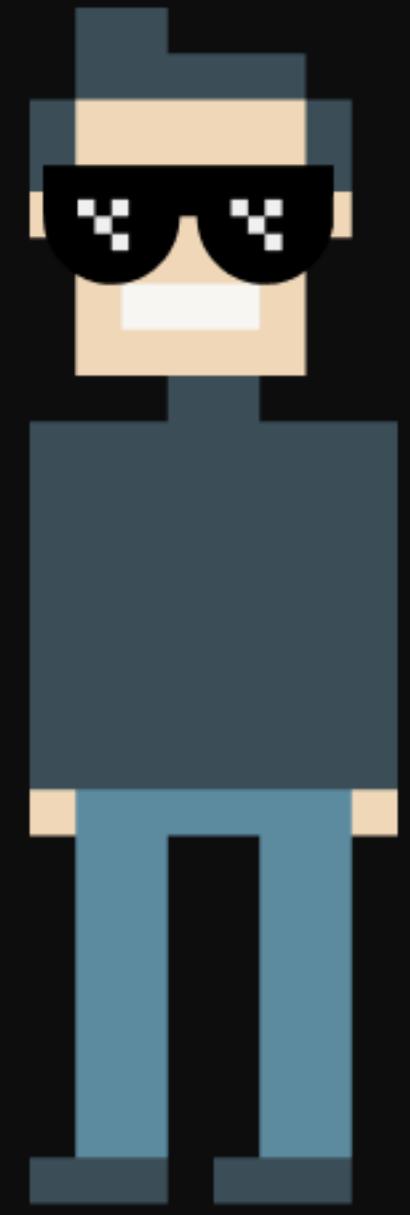
LEVEL
UP



Человек состоит на 80% из воды, а из чего состоит программист?

- ✍ Синтаксис – сложно представить программиста который не знает синтаксиса языка на котором пишет
- ✍ IDE
- ✍ Фреймворк
- ✍ Библиотеки
- ✍ Чистый код
- ✍ Паттерны
- ✍ Архитектура

LEVEL
UP

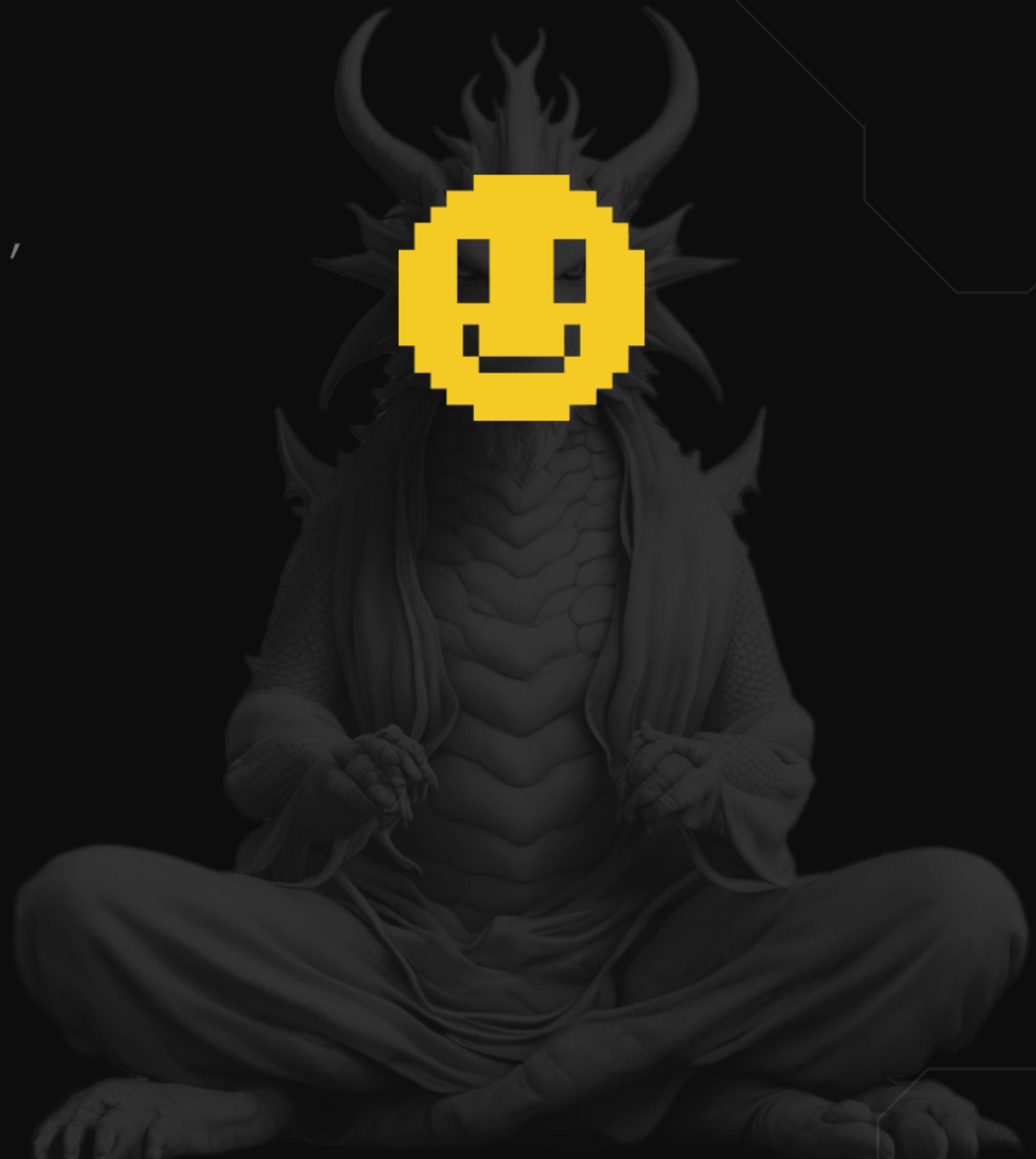


Выводы

Если чувствуете себя выжатым лимоном,
что не бойтесь делать перерывы
и анализировать, какие события
вас истощают больше всего

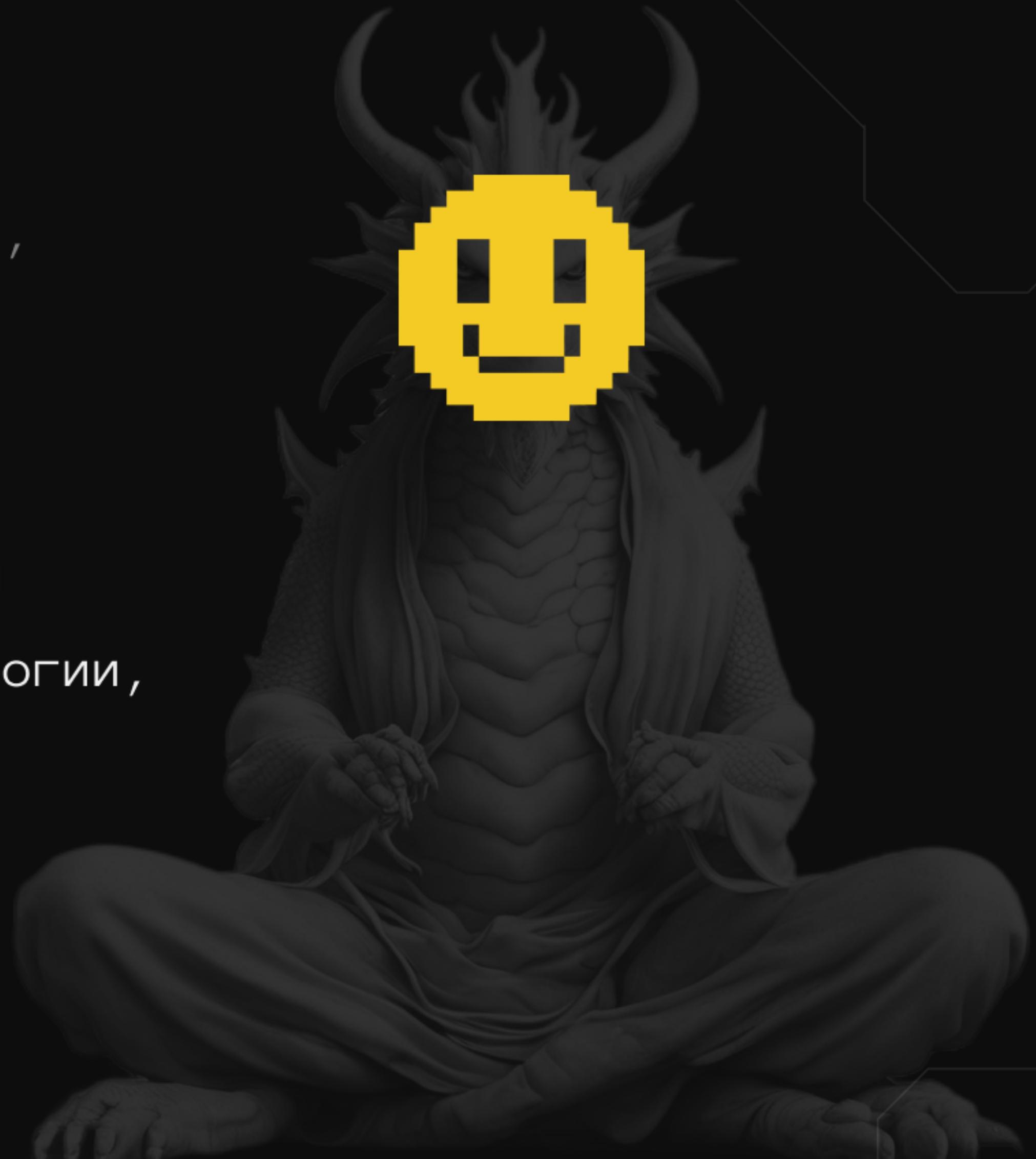
Выводы

- ④ Если чувствуете себя выжатым лимоном, что не бойтесь делать перерывы и анализировать, какие события вас истощают больше всего
- ④ Проще ориентироваться в коде когда в нем есть знакомые приемы и подходы



Выводы

- ④ Если чувствуете себя выжатым лимоном, что не бойтесь делать перерывы и анализировать, какие события вас истощают больше всего
- ④ Проще ориентироваться в коде когда в нем есть знакомые приемы и подходы
- ④ Практикуйтесь, изучайте новые технологии, паттерны





Сложная и многогранная природа когнитивной нагрузки в сфере понимания и стратегического подхода, чтобы ориентироваться в сложностях и оптимизировать распределение умственных способностей



Сложная и многогранная природа когнитивной нагрузки в сфере понимания и стратегического подхода, чтобы ориентироваться в сложностях и оптимизировать распределение умственных способностей

Мы только что создали ненужную
когнитивную нагрузку в вашей голове

