

# Ментальное программирование II

Мокевнин Кирилл, hexlet.io



<http://www.devconf.ru>

# Ментальное программирование

Создание такого код, читая который  
можно воспроизвести задачу в исходных  
терминах

## Structure and Interpretation of Computer Programs

Second Edition



Harold Abelson and  
Gerald Jay Sussman  
with Julie Sussman

## HOW TO DESIGN PROGRAMS

An Introduction to Programming and Computing

Matthias  
Felleisen

Robert Bruce  
Findler

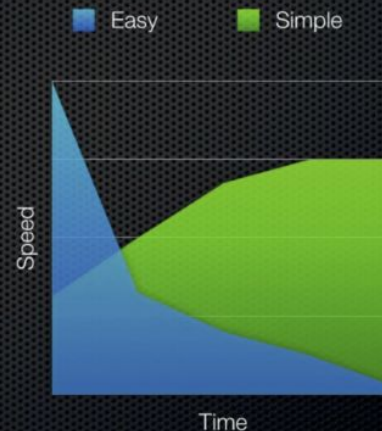
Matthew  
Flatt

Shriram  
Krishnamurthi

## Simple Made Easy

### Development Speed

- Emphasizing ease gives early speed
- Ignoring complexity will slow you down over the long haul
- On throwaway or trivial projects, nothing much matters



# Пример

```
const keys = Object.keys(obj1)
  .concat(Object.keys(obj2))
  .reduce((acc, el) => {
    if (!acc.find(e => e === el)) return [...acc, el];
    return acc;
  }, []);
```

# Пример

```
const keys = Object.keys({ ...obj1, ...obj2 });
```



# Пример

```
const keys = _.union(_.keys(obj1), _.keys(obj2));
```

# Выделение функции

- Повышение уровня абстракции
- Дублирование - побочный эффект
- Заморозка технического долга
- Отложенные оптимизации



```
const build = (key, obj1, obj2) => {  
  const val1 = obj1[key];  
  const val2 = obj2[key];  
  if (val1 && val2) {  
    if (val1 === val2 || (_.is0(val1) && _.is(val2))) {  
      const children = _.is0(val2) ? buildDiff(val1, val2) : [];  
      return { key, children, type: 'unchanged',  
        value: _.is(val1) ? null : val1,  
      };  
    }  
  }  
  // ...  
};
```

# Именование

Функция - Глагол (потому что действие)

`parser()` => `parse()`

# Декларативность

Naval Fate.

Usage:

```
naval_fate ship new <name>...
```

```
naval_fate ship <name> move <x> <y> [--speed=<kn>]
```

```
naval_fate ship shoot <x> <y>
```

Options:

```
--version    Show version.
```

```
--speed=<kn> Speed in knots [default: 10].
```

## Корреляция 96%



# Пожарники

Чем больше в городе пожарников, тем  
больше в городе пожаров

# Выбор языка

```
if I18n.locale === :ru
```

```
  # ...
```

```
else # en
```

```
  # ...
```

```
end
```

# Элементы массива

```
const arr = [3, 4, 9];  
if (arr.pop() % 2 === 0) {  
  console.log('Done');  
}
```



# Выбор языка

```
if I18n.locale === :ru  
  # ...  
else # en  
  # ...  
end
```

# Идеальная функция

???

# Структура кода

1. Побочные эффекты
2. Чистый код
3. Побочные эффекты

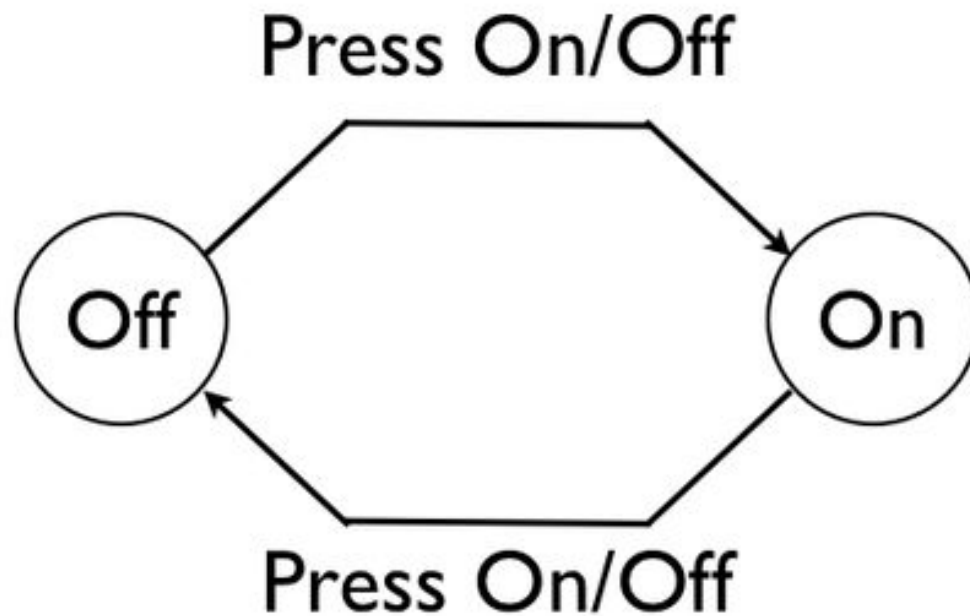
# Плохо

```
class Reporter
{
    private $data;
    public function __constructor($path)
    {
        $this->data = file_get_contents($path);
    }
}
```

# Хорошо

```
class Reporter
{
    private $data;
    public function __constructor($data)
    {
        $this->data = $data;
    }
}
```

# Конечный автомат



# Автоматное программирование

```
state_machine :email_state, initial: :active do
  state :active
  state :inactive

  event :activate { transition inactive: :active }
  event :deactivate { transition all => :inactive }
end
```



Спасибо!

mokevnin@gmail.com

<https://twitter.com/mokevnin>