



Феникс

разработка сайтов  
для застройщиков

# Неочевидные возможности SVG в современном фронтэнде



Константин Остров  
Креативная группа «Феникс»



# Константин Остров

Занимаюсь разработкой с 2014г



# Константин Остров

Занимаюсь разработкой с 2014г

Руководитель фронтэнд разработки в  
“Фениксе”



# Константин Остров

Занимаюсь разработкой с 2014г

Руководитель фронтэнд разработки в  
“Фениксе”

Люблю SVG

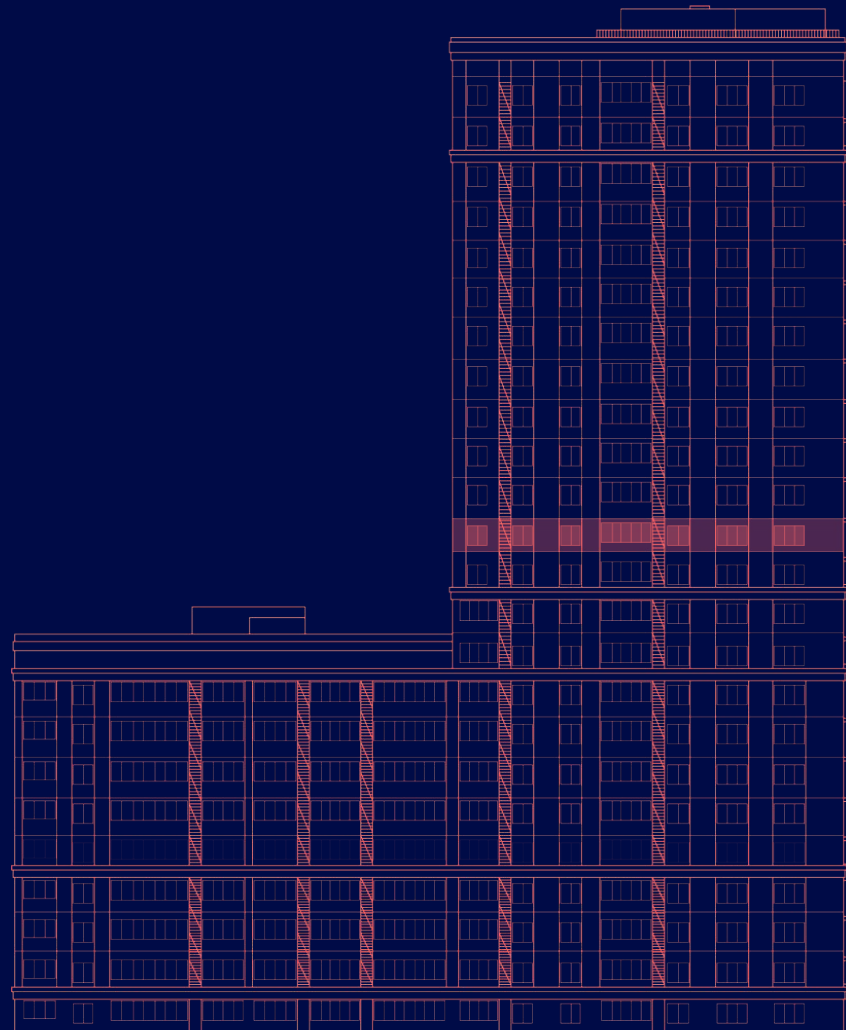


**Феникс**

разработка сайтов  
для застройщиков

# Мы двигаем, недвижимое!

- Разработка и прототипирования сайтов для застройщиков
- Разработка ERP-систем
- Аудит
- Реклама
- Маркетинговые исследования



# Подбор по параметрам

Комнат

Студия

1-к

2-к

3-к

Площадь

от

28

м²

до

81

м²

Цена

от

3 244 150

Р

до

7 495 589

Р

Особенности

Не важно

Найдено 382 квартиры

✕ Сбросить фильтры

Сортировка

По умолчанию

ул. Ново-садовая

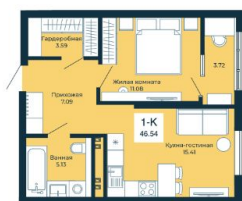


**3-к квартира № 1.2.1 • 80.05 м²**

Секция № 1 • Этаж 2 из 25

6 988 365 Р

ул. Ново-садовая



**1-к квартира № 1.2.2 • 46.54 м²**

Секция № 1 • Этаж 2 из 25

4 649 811 Р

ул. Ново-садовая

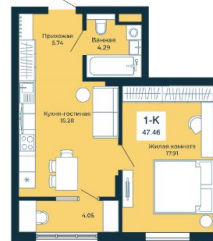


**1-к квартира № 1.2.3 • 58.77 м²**

Секция № 1 • Этаж 2 из 25

5 289 300 Р

ул. Ново-садовая



**1-к квартира № 1.2.4 • 47.46 м²**

Секция № 1 • Этаж 2 из 25

4 603 620 Р

ул. Ново-садовая



**1-к квартира № 1.2.3 ■ 58.77 м²**

Секция № 1 ■ Этаж 2 из 25

5 289 300 ₽

ул. Ново-садовая



**1-к квартира № 1.2.4 ■ 47.46 м²**

Секция № 1 ■ Этаж 2 из 25

4 603 620 ₽



**3-к квартира № 1.3.1 ■ 79.06 м<sup>2</sup>**

Секция № 1 ■ Этаж 3 из 25

7 115 400 Р





| Секция 1   | Секция 2 | Секция 3 |
|--|----------|----------|
|  |          |          |
| 25 этаж ▾  |          |          |
| 24 этаж ▾  |          |          |
| 21 этаж ▾<br>Ст. от 3,2 млн<br>1-к от 4,6 млн<br>2-к от 5,2 млн<br>3-к от 6,0 млн  |          |          |
| Далее →  |          |          |
| 20 этаж ▾  |          |          |
| 19 этаж ▾  |          |          |
|  |          |          |
| A 3D architectural rendering of a modern building complex. The central tower is highlighted with a yellow band around the 21st floor. The building has a mix of brown, white, and yellow facades. The surrounding area includes a parking lot with cars and some greenery. |          |          |

# Что здесь происходит



# Что здесь происходит



- SVC

# Что здесь происходит



## 15 этаж

8 квартир в продаже

Ст. — от 9 999 999 Р

1-к — от 9 999 999 Р

2-к — от 9 999 999 Р

3-к — от 9 999 999 Р

- SVG
- Area

# Что здесь происходит



- SVG
- Area
- Canvas

# О чем поговорим

- позиционирование объектов внутри SVG и о том, как им овладеть;

# О чем поговорим

- позиционирование объектов внутри SVG и о том, как им овладеть;
- внедрение динамических элементов внутрь SVG;



# О чем поговорим

- позиционирование объектов внутри SVG и о том, как им овладеть;
- внедрение динамических элементов внутрь SVG;
- расчет координат объектов вне box модели;



# О чем поговорим

- позиционирование объектов внутри SVG и о том, как им овладеть;
- внедрение динамических элементов внутрь SVG;
- расчет координат объектов вне box модели;
- возможности влиять на содержимое SVG средствами CSS и JS, ограничения и способы решения;

# О чем поговорим

- позиционирование объектов внутри SVG и о том, как им овладеть;
- внедрение динамических элементов внутрь SVG;
- расчет координат объектов вне box модели;
- возможности влиять на содержимое SVG средствами CSS и JS, ограничения и способы решения;
- производительность SVG в вебе и о том, как ее можно улучшить.



Феникс

разработка сайтов  
для застройщиков

# Что такое SVG



# Что такое SVG

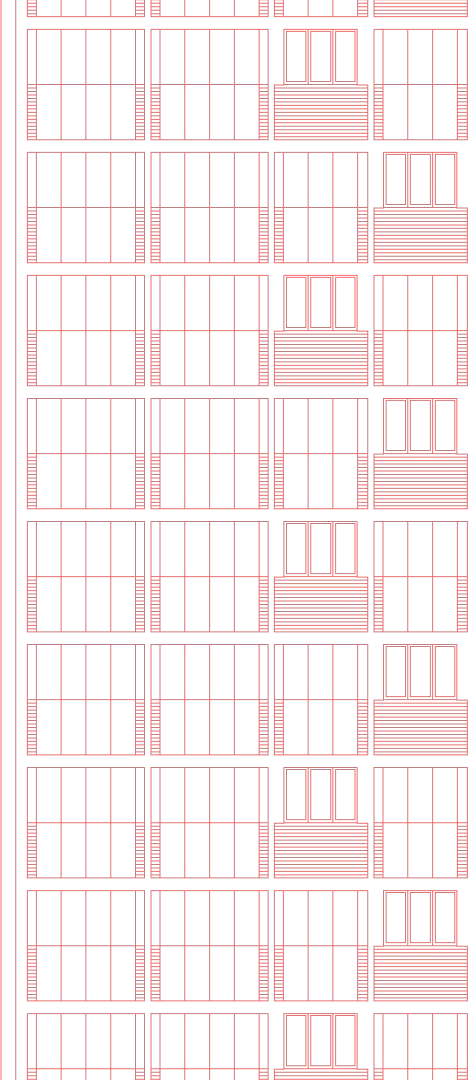
**This page contains the following errors:**

error on line 1 at column 21: Specification mandates value for attribute heie

**Below is a rendering of the page up to the first error.**

# Что такое SVG

- Не теряет качество при ресайзе



# Что такое SVG

- Не теряет качество при ресайзе
- Можно создавать и редактировать в IDE



# Что такое SVG

- Не теряет качество при ресайзе
- Можно создавать и редактировать в IDE

```
<svg preserveAspectRatio="xMidYMid" viewBox="0 0 244 218" fill="none" xmlns="http://www.w3.org/2000/svg">
  <g clip-path="url(#clip0_2_7)">
    <rect width="244" height="218" transform="matrix(-1 0 0 1 244 0)" fill="#00624F"/>
    <path fill-rule="evenodd" clip-rule="evenodd" d="M0.646835 95.7551C0.48613 ..." fill="#4F8475"/>
  </g>
</svg>
```

# Что такое SVG

- Не теряет качество при ресайзе
- Можно создавать и редактировать в IDE
- Доступность для скринридеров

```
<svg role="img" aria-describedby="fox description" ...>  
  <title id="fox">What does the fox say?</title>  
  <desc id="description">Will we ever know?</desc>  
  <path ... />  
</svg>
```



# Что такое SVG

- Не теряет качество при ресайзе
- Можно создавать и редактировать в IDE
- Доступность для скринридеров
- Возможность анимирования

```
<svg viewBox="0 0 200 200" width="200" height="200">  
  <rect class="block" width="200" height="200" x="0" y="0">  
    <animate attributeName="fill" dur="5000ms" from="#262335" to="#ff7edb" fill="freeze" />  
  </rect>  
</svg>
```

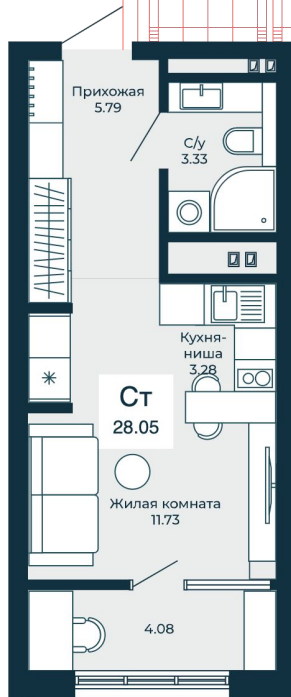
# Что такое SVG

- Не теряет качество при ресайзе
- Можно создавать и редактировать в IDE
- Доступность для скринридеров
- Возможность анимирования
- Меньший размер по сравнению с растровыми изображениями

SVG: 20kb

Webp (500x1147): 21kb

PNG (500x1147): 70kb



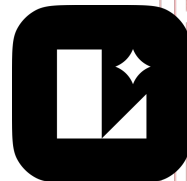
# Что такое SVG

- Не теряет качество при ресайзе
- Можно создавать и редактировать в IDE
- Доступность для скринридеров
- Возможность анимирования
- Меньший размер по сравнению с растровыми изображениями
- Широкая поддержка браузеров и инструментов



# Что такое SVG

- Не теряет качество при ресайзе
- Можно создавать и редактировать в IDE
- Доступность для скринридеров
- Возможность анимирования
- Меньший размер по сравнению с растровыми изображениями
- Широкая поддержка браузеров и инструментов





Феникс

разработка сайтов  
для застройщиков



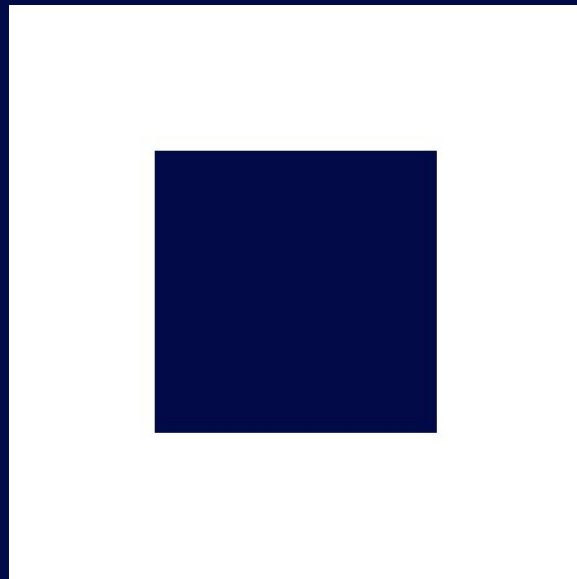
# Что внутри SVG

Что внутри SVG

# Фигуры

## Rectangle

```
<rect  
  x="62"  
  y="62"  
  width="120"  
  height="120"  
  fill="#020B47"  
>
```

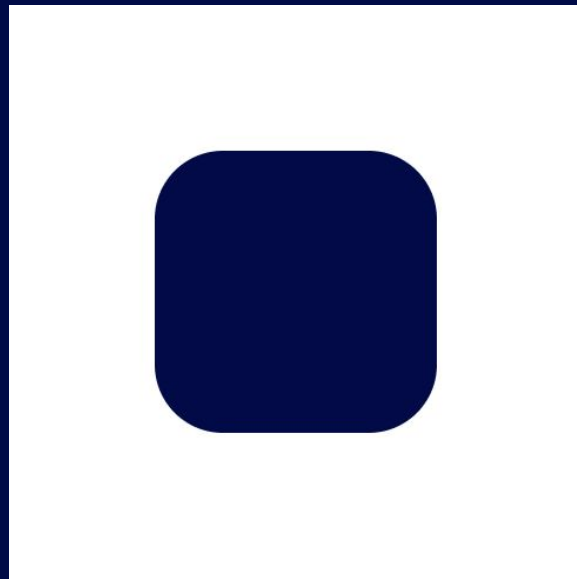


Что внутри SVG

# Фигуры

## Rectangle

```
<rect  
  x="62"  
  y="62"  
  rx="29"  
  width="120"  
  height="120"  
  fill="#020B47"  
>
```

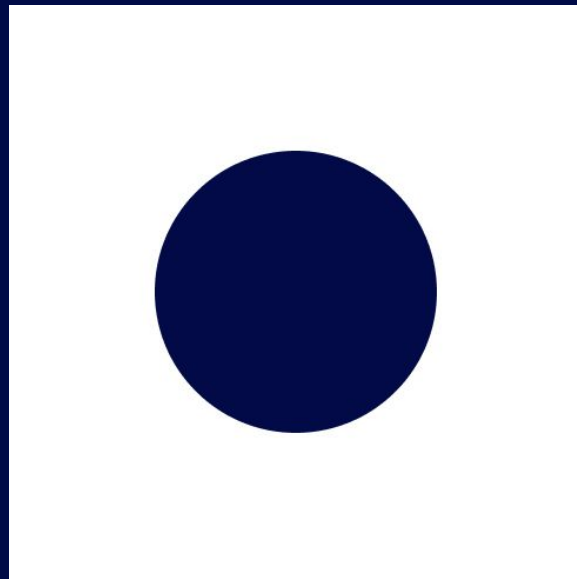


Что внутри SVG

# Фигуры

## Circle

```
<circle  
  cx="122"  
  cy="122"  
  r="60"  
  fill="#020B47"  
>
```



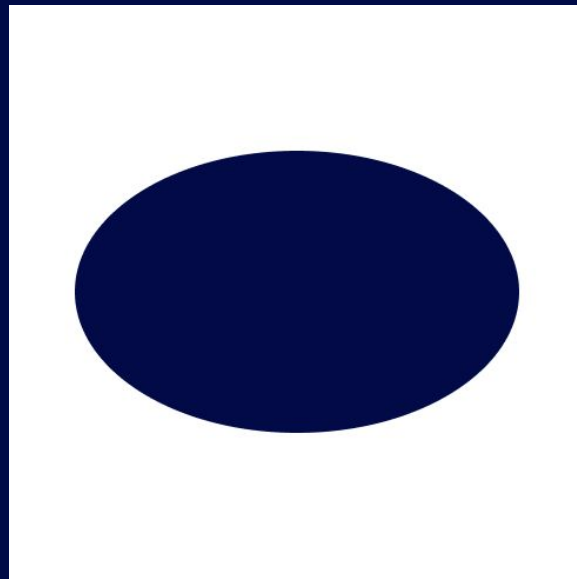


Что внутри SVG

# Фигуры

## Ellipse

```
<ellipse  
  cx="122.5"  
  cy="122"  
  rx="94.5"  
  ry="60"  
  fill="#020B47"  
>
```

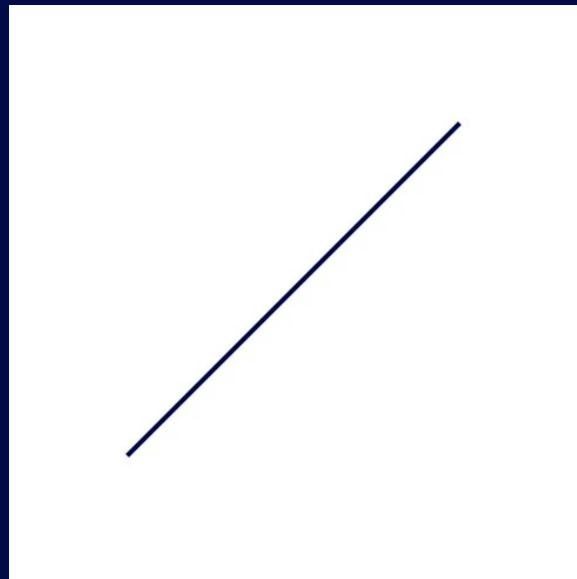


# Что внутри SVG

## Фигуры

### Line

```
<line  
  x1="50.2929"  
  y1="191.714"  
  x2="191.714"  
  y2="50.2929"  
  stroke="#020B47"  
  stroke-width="2"  
>
```

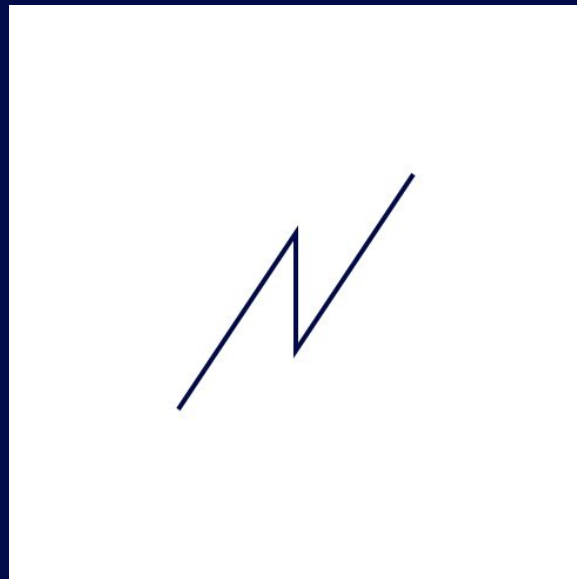


Что внутри SVG

# Фигуры

## Polyline

```
<polyline  
  points="0,100 50,25 50,75 100,0"  
  fill="none"  
  stroke="#020B47"  
  stroke-width="2"  
></polyline>
```

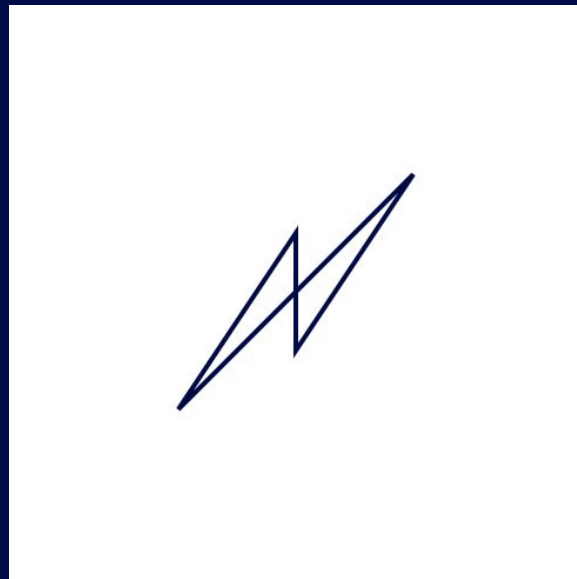


Что внутри SVG

# Фигуры

## Polygon

```
<polygon  
  points="100,100 150,25 150,75 200,0"  
  fill="none"  
  stroke="#020B47"  
  stroke-width="2"  
></polyline>
```



Что внутри SVG

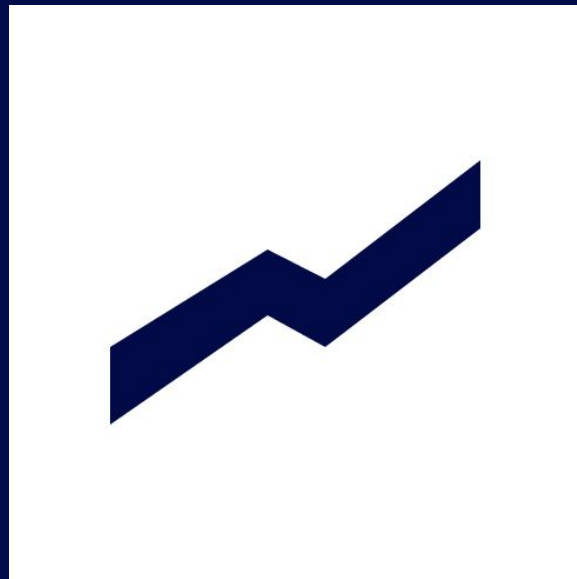
# Фигуры

## Path

```
<path
```

```
  d="M43 145.5V178.5L110  
    132L134.5 145.5L200.5  
    95V66L134.5 116.5L110  
    104L43 145.5Z"  
  fill="#020B47"
```

```
/>
```



Что внутри SVG

# Path

Атрибут d. Команды

- MoveTo: M, m

```
<path
  d="M43 145.5V178.5L110
    132L134.5 145.5L200.5
    95V66L134.5 116.5L110
    104L43 145.5Z"
  fill="#020B47"
/>
```

# Path

## Атрибут d. Команды

- **MoveTo:** M, m
- **LineTo:** L, l, H, h, V, v

```
<path
  d="M43 145.5V178.5L110
    132L134.5 145.5L200.5
    95V66L134.5 116.5L110
    104L43 145.5Z"
  fill="#020B47"
/>
```

# Path

## Атрибут d. Команды

- **MoveTo:** M, m
- **LineTo:** L, l, H, h, V, v
- **Cubic Bézier Curve:** C, c, S, s

```
<path
  d="M43 145.5V178.5L110
    132L134.5 145.5L200.5
    95V66L134.5 116.5L110
    104L43 145.5Z"
  fill="#020B47"
/>
```



# Path

## Атрибут d. Команды

- MoveTo: M, m
- LineTo: L, l, H, h, V, v
- Cubic Bézier Curve: C, c, S, s
- Quadratic Bézier Curve: Q, q, T, t

```
<path
  d="M43 145.5V178.5L110
    132L134.5 145.5L200.5
    95V66L134.5 116.5L110
    104L43 145.5Z"
  fill="#020B47"
/>
```

Что внутри SVG

# Path

Атрибут d. Команды

- MoveTo: M, m
- LineTo: L, l, H, h, V, v
- Cubic Bézier Curve: C, c, S, s
- Quadratic Bézier Curve: Q, q, T, t
- Elliptical Arc Curve: A, a

```
<path
  d="M43 145.5V178.5L110
    132L134.5 145.5L200.5
    95V66L134.5 116.5L110
    104L43 145.5Z"
  fill="#020B47"
/>
```

Что внутри SVG

# Path

Атрибут d. Команды

- MoveTo: M, m
- LineTo: L, l, H, h, V, v
- Cubic Bézier Curve: C, c, S, s
- Quadratic Bézier Curve: Q, q, T, t
- Elliptical Arc Curve: A, a
- ClosePath: Z, z

```
<path
  d="M43 145.5V178.5L110
    132L134.5 145.5L200.5
    95V66L134.5 116.5L110
    104L43 145.5Z"
  fill="#020B47"
/>
```

Что внутри SVG

# Fill

```
<path  
  d="..."  
  fill="#FFFFFF"  
>
```



Что внутри SVG

# Stroke

```
<path  
  d="..."  
  fill="#FFFFFF"  
  stroke="#FB6566"  
  stroke-opacity="0.8"  
  stroke-width="14"  
>
```

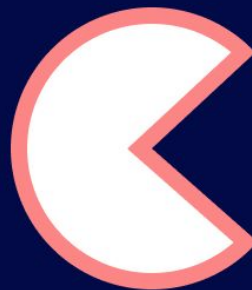


Что внутри SVG

# Mask

```
<mask id="mask">  
  <path d="..." fill="white"/>  
</mask>
```

```
<path  
  d="..."  
  fill="#FFFFFF"  
  stroke="#FB6566"  
  stroke-opacity="0.8"  
  stroke-width="14"  
  mask="url(#mask)"  
>
```

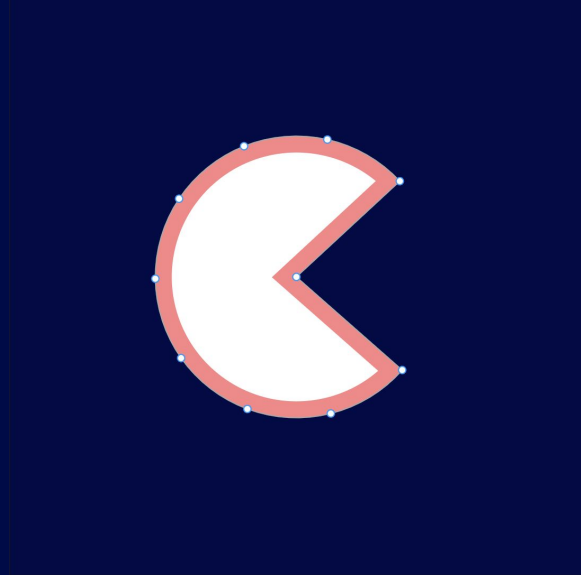


Что внутри SVG

# Mask

```
<mask id="mask">  
  <path d="..." fill="white"/>  
</mask>
```

```
<path  
  d="..."  
  fill="#FFFFFF"  
  stroke="#FB6566"  
  stroke-opacity="0.8"  
  stroke-width="14"  
  mask="url(#mask)"  
>
```

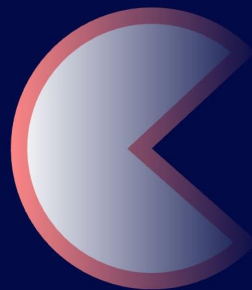
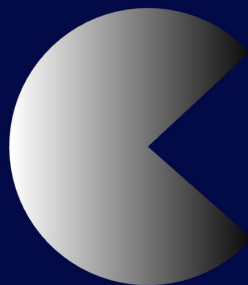


Что внутри SVG

# Mask

```
<mask id="mask">  
  <path d="..." fill="white"/>  
</mask>
```

```
<path  
  d="..."  
  fill="#FFFFFF"  
  stroke="#FB6566"  
  stroke-opacity="0.8"  
  stroke-width="14"  
  mask="url(#mask)"  
>
```





Что внутри SVG

# TextPath

```
<path
  id="path"
  d="..."
  fill="#FFFFFF"
  stroke="#FB6566"
  stroke-opacity="0.8"
  stroke-width="14"
  mask="url(#mask)"
/>
<text fill="white">
  <textPath href="#path" startOffset="8">...</textPath>
</text>
```



Что внутри SVG

# Filter

```
<defs>
  <filter
    id="filter"
    width="150%"
    height="160%"
    x="-25%"
    y="-25%"
  >
    ...
  </filter>
</defs>
<text filter="url(#filter)">
  HolyJS
</text>
```

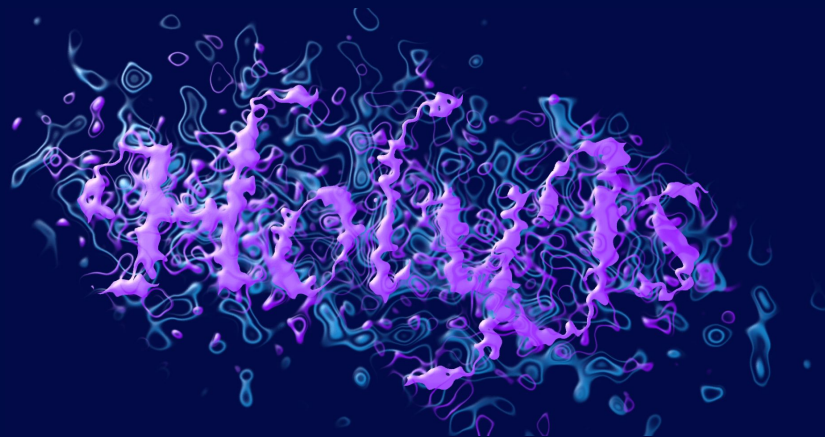
*HolyJS*

Что внутри SVG

# Filter

```
<defs>
  <filter
    id="filter"
    width="150%"
    height="160%"
    x="-25%"
    y="-25%"
  >
    ...
  </filter>
</defs>
<text filter="url(#filter)">
  HolyJS
</text>
```

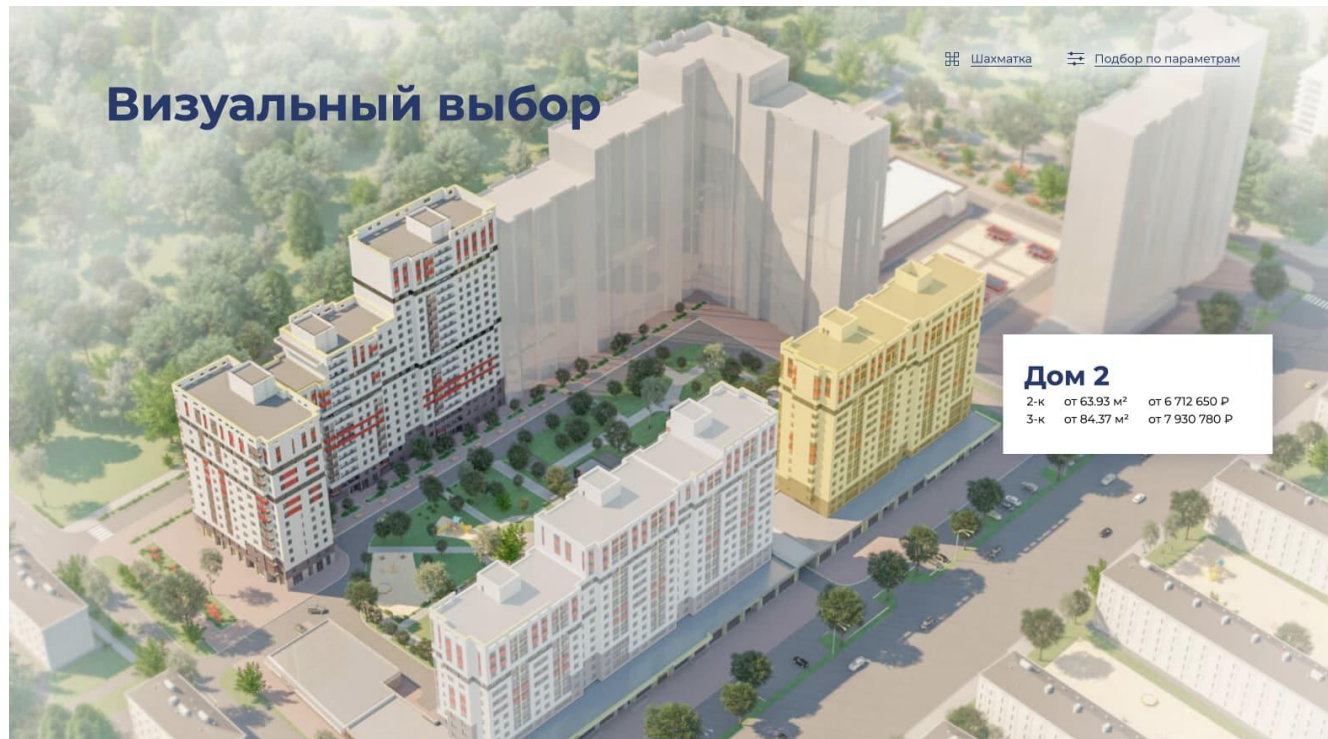
HolyJS



# Задачи

# Позиционирование внутри SVG

# Позиционирование внутри SVG



Количество комнат

Ст. 1 2 3

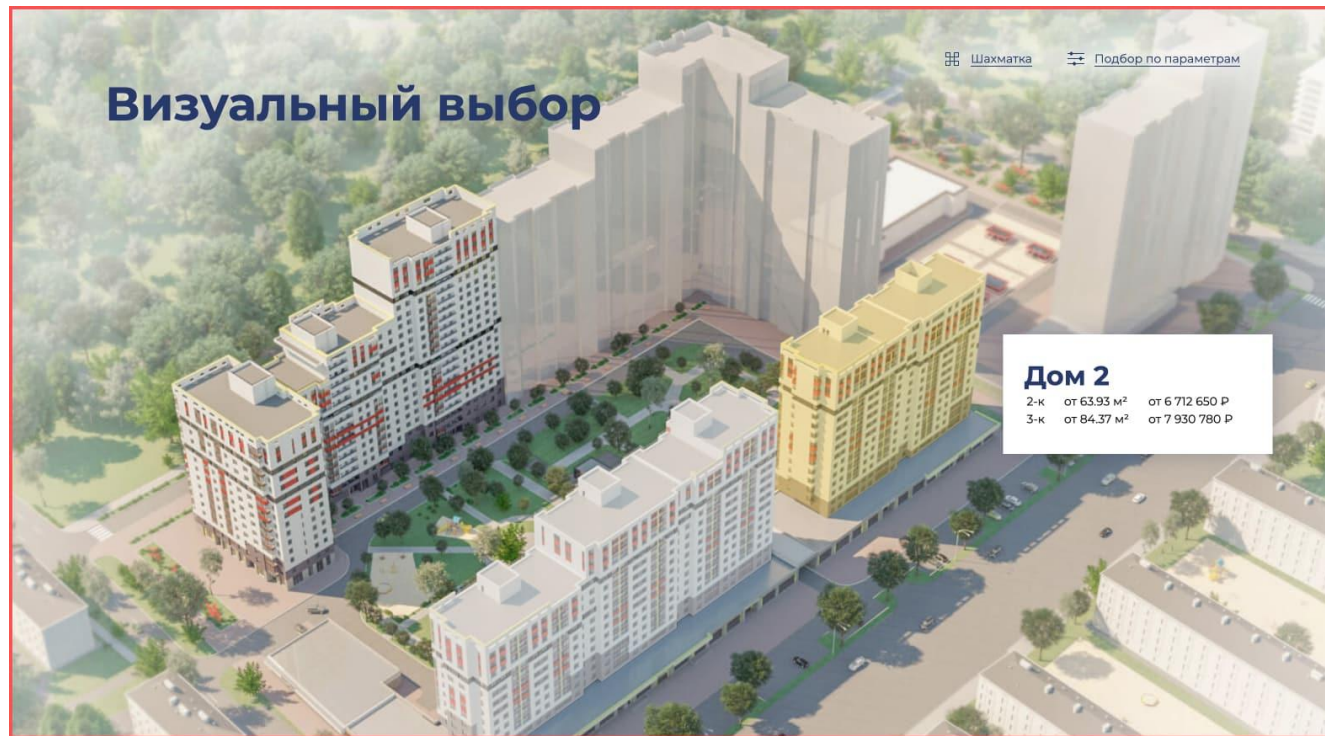
Площадь, м2

23,7 — 91,1

Цена

3 246 900 — 8 820 000

# Позиционирование внутри SVG



Количество комнат

Ст. 1 2 3

Площадь, м2

23,7 — 91,1

Цена

3 246 900 — 8 820 000



# Позиционирование внутри SVG



## 21 этаж

Секция 1 (32 этажа)  
В продаже : 6 квартир

|         |                |
|---------|----------------|
| 1-комн. | от 4 500 000 Р |
| 2-комн. | от 5 700 000 Р |
| 3-комн. | от 6 750 000 Р |



ТРЦ «Гринвич»





# Позиционирование внутри SVG



## 21 этаж

Секция 1 (32 этажа)

В продаже : 6 квартир

|         |                |
|---------|----------------|
| 1-комн. | от 4 500 000 Р |
| 2-комн. | от 5 700 000 Р |
| 3-комн. | от 6 750 000 Р |

ул. Хохрякова



ТРЦ «Гринвич»



# Позиционирование внутри SVG



## 21 этаж

Секция 1 (32 этажа)  
В продаже : 6 квартир

|         |                |
|---------|----------------|
| 1-комн. | от 4 500 000 Р |
| 2-комн. | от 5 700 000 Р |
| 3-комн. | от 6 750 000 Р |



ТРЦ «Гринвич»



# Позиционирование внутри SVG

`<img>`: `object-fit` + `object-position`

`<any>`: `background-image` + `background-size` + `background-position`

# Позиционирование внутри SVG

`<img>`: object-fit + object-position

`<any>`: background-image + background-size + background-position

`<svg>`:



# Позиционирование внутри SVG

```
const ratio = 0.5395
let finalWidth = 0
let finalHeight = 0
let topPosition = 0
let leftPosition = 0

const areaWidth = this.$el.offsetWidth
const areaHeight = this.$el.offsetHeight

const areaRatio = areaHeight / areaWidth

if (areaRatio > ratio) {
  finalWidth = areaHeight / ratio
  finalHeight = areaHeight
  leftPosition = (areaWidth - finalWidth) / 2
} else {
  finalWidth = areaWidth
  finalHeight = areaWidth * ratio
  topPosition = (areaHeight - finalHeight) / 2
}

let style = {
  width: `${finalWidth}px`,
  height: `${finalHeight}px`,
  top: `${topPosition}px`,
  left: `${leftPosition}px`
}
```



Феникс

разработка сайтов  
для застройщиков

# preserveAspectRatio

<align> <meetOrSlice>

Align: x[Mid|Min|Max]y[Mid|Min|Max]



Default (xMidYMid meet)



xMidYMid slice



Default (xMidYMid meet)



object-fit: contain

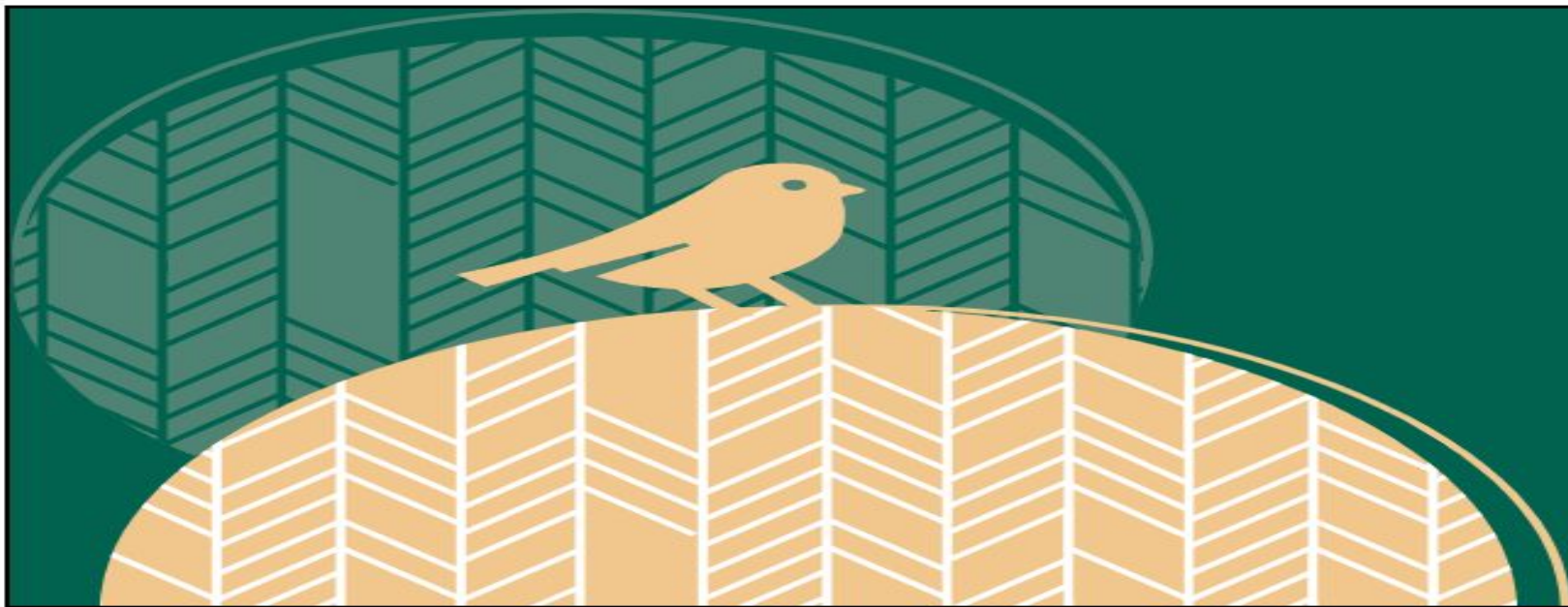
xMidYMid slice



object-fit: cover



None



# Позиционирование внутри SVG

```
const ratio = 0.5395
let finalWidth = 0
let finalHeight = 0
let topPosition = 0
let leftPosition = 0

const areaWidth = this.$el.offsetWidth
const areaHeight = this.$el.offsetHeight

const areaRatio = areaHeight / areaWidth

if (areaRatio > ratio) {
  finalWidth = areaHeight / ratio
  finalHeight = areaHeight
  leftPosition = (areaWidth - finalWidth) / 2
} else {
  finalWidth = areaWidth
  finalHeight = areaWidth * ratio
  topPosition = (areaHeight - finalHeight) / 2
}

let style = {
  width: `${finalWidth}px`,
  height: `${finalHeight}px`,
  top: `${topPosition}px`,
  left: `${leftPosition}px`
}
```

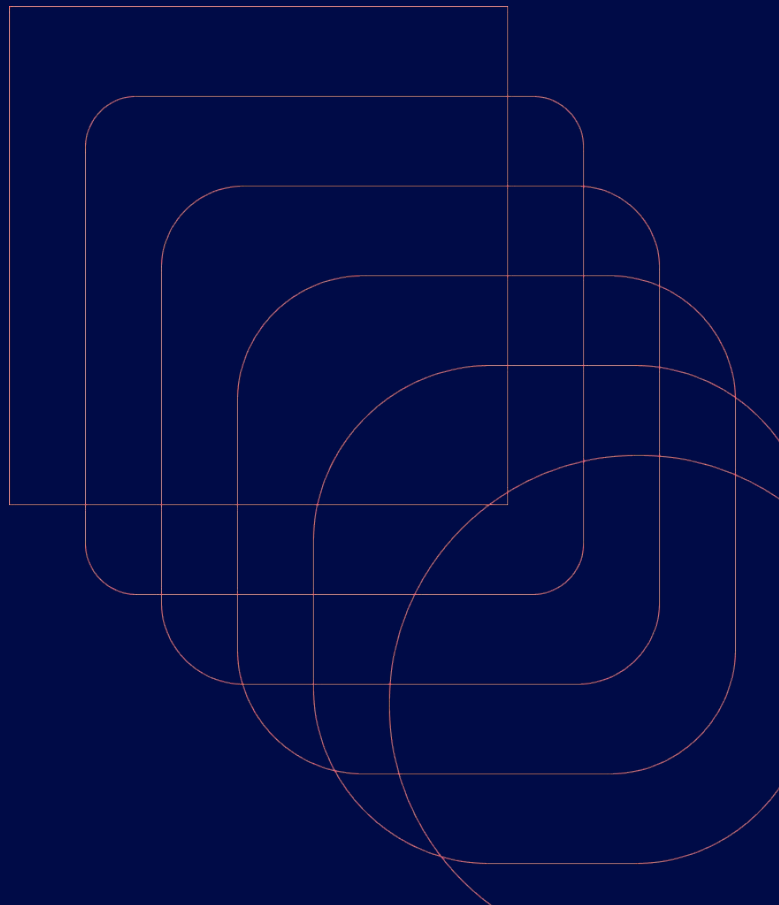
*preserveAspectRatio="xMidYMid slice"*



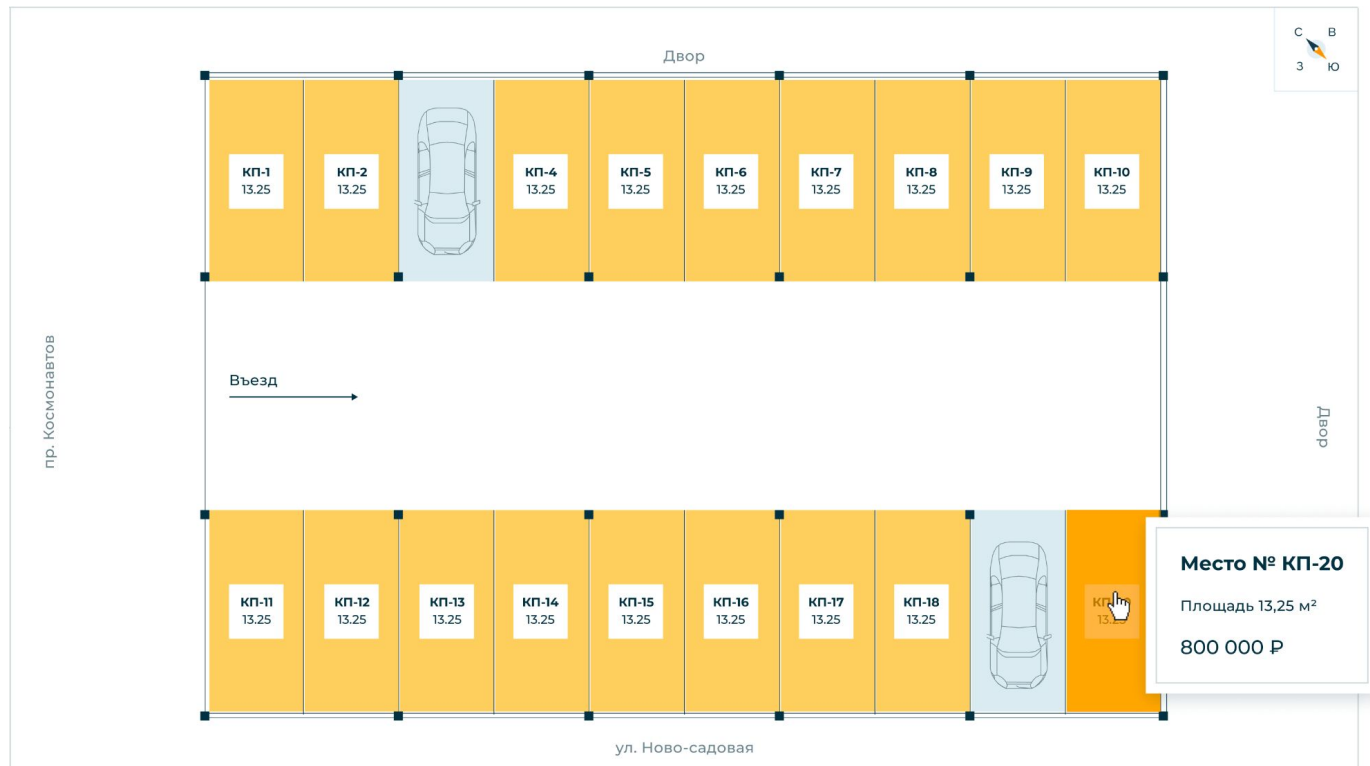
Феникс

разработка сайтов  
для застройщиков

# Внедрение динамических элементов внутрь SVG

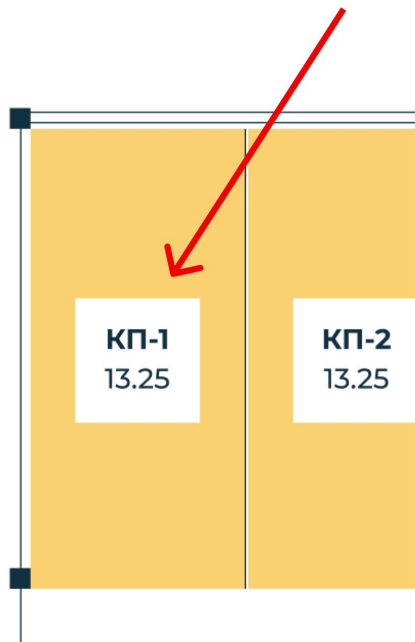


# Внедрение динамических элементов внутрь SVG



# Внедрение динамических элементов внутрь SVG

---





# Чужеродный объект

<foreignObject />

# Внедрение динамических элементов внутри SVG

```
<div class="floor">  
    
  <svg class="floor_svg" viewBox="0 0 900 600">  
    <path ... />  
    <foreignObject x="20" y="50" class="floor_parking-info">  
      <div>  
        Text  
      </div>  
    </foreignObject>  
  </svg>  
</div>
```

# Внедрение динамических элементов внутри SVG

```
<div class="floor">
  
  <svg class="floor_svg" viewBox="0 0 900 600">
    <path ... />
    <foreignObject x="20" y="50" class="floor_parking-info">
      <div>
        Text
      </div>
    </foreignObject>
  </svg>
</div>
```



# Внедрение динамических элементов внутри SVG

```
<div class="floor">  
    
  <svg class="floor_svg" viewBox="0 0 900 600">  
    <path ... />  
    <foreignObject x="20" y="50" class="floor_parking-info">  
      <div>  
        Text  
      </div>  
    </foreignObject>  
  </svg>  
</div>
```



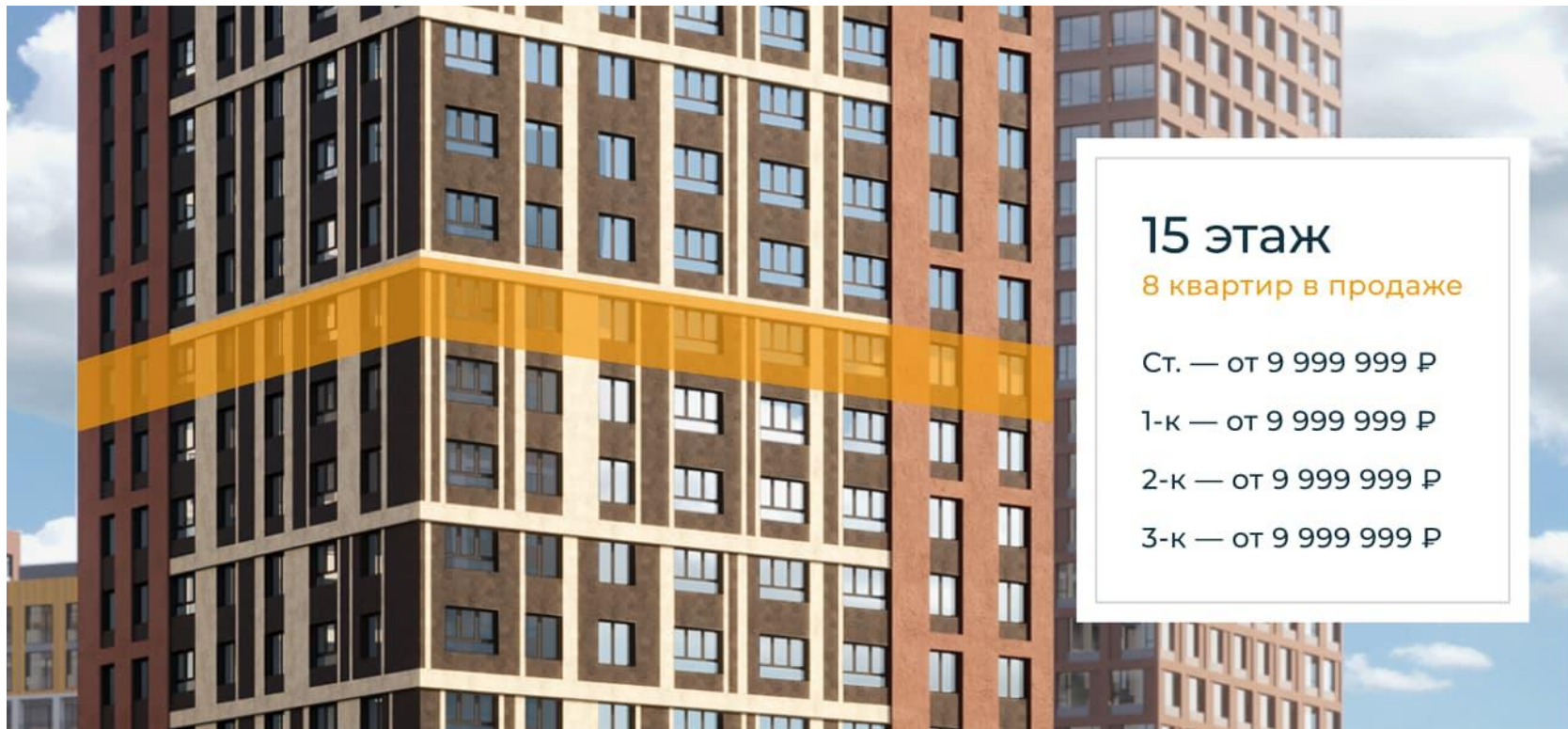
Феникс

разработка сайтов  
для застройщиков

# Расчет координат вне BOX модели



# Расчет координат вне BOX модели



**15 этаж**

8 квартир в продаже

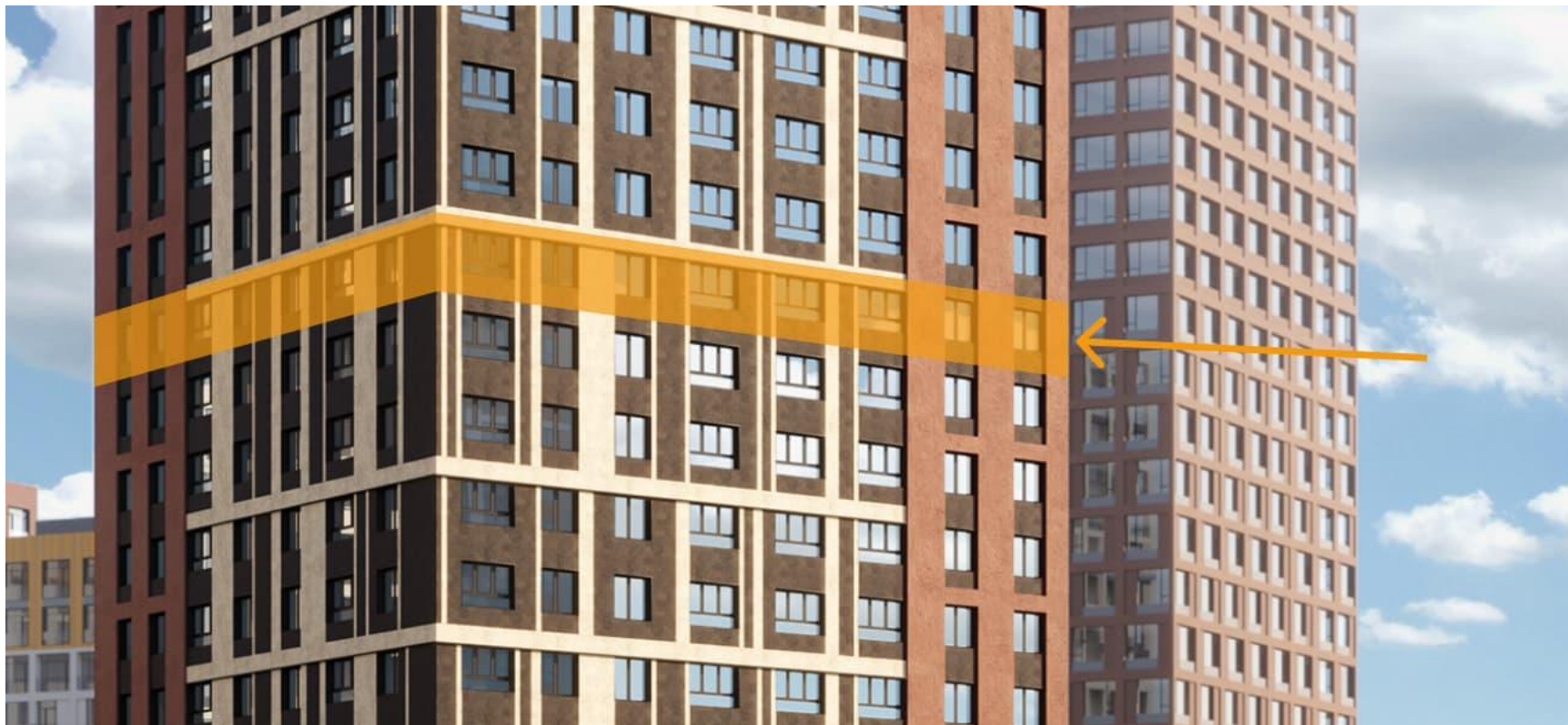
Ст. — от 9 999 999 ₽

1-к — от 9 999 999 ₽

2-к — от 9 999 999 ₽

3-к — от 9 999 999 ₽

## Расчет координат вне BOX модели



# Расчет координат вне BOX модели





# Точки на пути

points-on-path

[npmjs.com/package/points-on-path](https://npmjs.com/package/points-on-path)



# Расчет координат вне BOX модели

```
<path xmlns="http://www.w3.org/2000/svg" d="M693.75 386.016L584.062  
421.641V444.141L693.75 413.438L895.312 438.984V415.312L693.75 386.016Z"  
class="st1"></path>
```

# Расчет координат вне BOX модели

```
<path xmlns="http://www.w3.org/2000/svg" d="M693.75 386.016L584.062  
421.641V444.141L693.75 413.438L895.312 438.984V415.312L693.75 386.016Z"  
class="st1"></path>
```

```
const [pathPoints] = pointsOnPath(path.getAttribute('d'))  
  
const sortedXPoints = pathPoints.sort((a, b) => a[0] - b[0])  
const [firstEdgePoint, secondEdgePoint] = sortedXPoints  
sortedXPoints.reverse()  
const [lastEdgePoint, beforeLastEdgePoint] = sortedXPoints
```

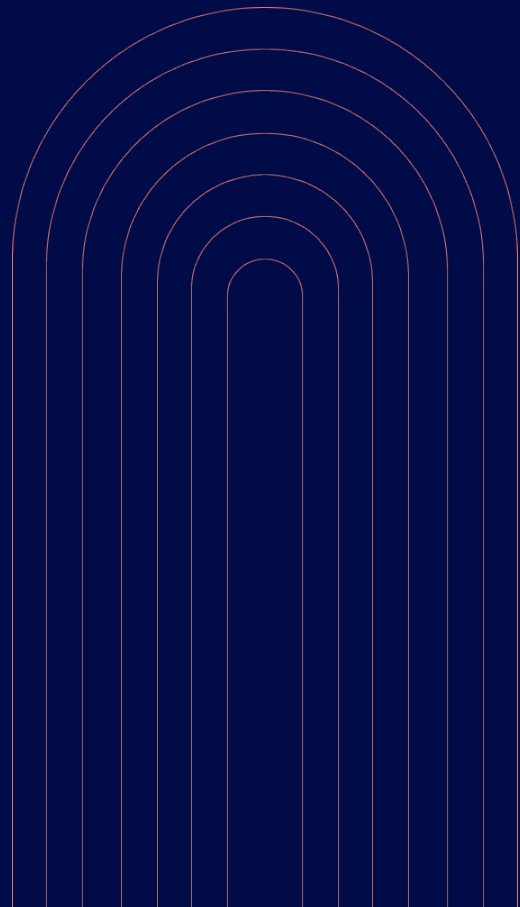




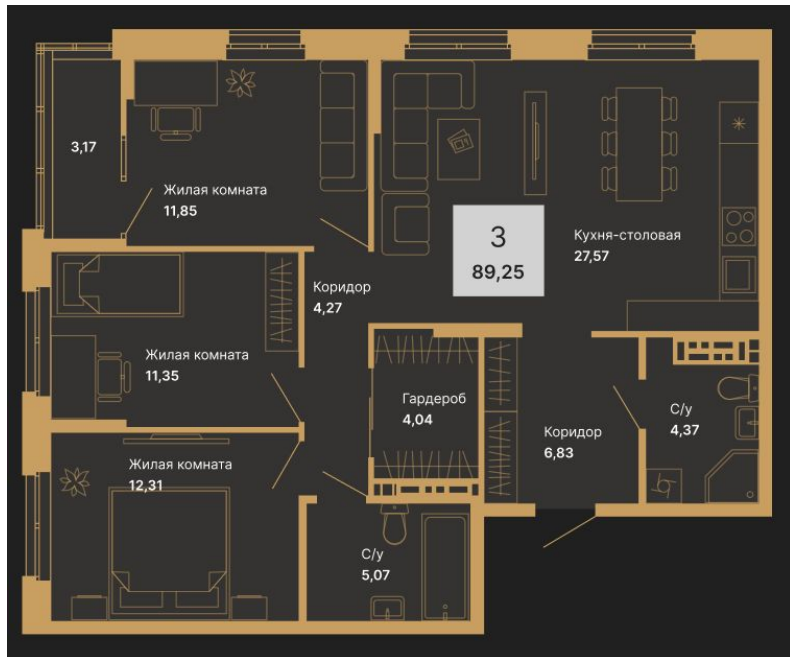
Феникс

разработка сайтов  
для застройщиков

# Влияем на SVG с помощью CSS и JS



# Влияем на SVG с помощью CSS и JS



# Способы встраивания

Img

CSS background

Object

Sprite

Inline SVG

Способы встраивания

# Img и CSS background

```

```

```
.bird {  
  background-image: url(/bird.svg);  
}
```

Способы встраивания

# Img и CSS background

```

```

```
.bird {  
  background-image: url(/bird.svg);  
}
```

```
.bird {  
  filter: grayscale(1);  
}
```

```
.bird {  
  filter: url(filters.svg#grayscale);  
}
```

Способы встраивания

# Img и CSS background



плюсы

Простота

Способы встраивания

# Img и CSS background



плюсы

Простота



минусы

Только SVG и CSS фильтры

Способы встраивания

# Object

```
<object data="/bird.svg">
```



Способы встраивания

# Object

JS

```
const bird = document.querySelector('#bird')  
const wing = bird.contentDocument.querySelector('#wing')
```

# Object

## JS

```
const bird = document.querySelector('#bird')  
const wing = bird.contentDocument.querySelector('#wing')
```

## CSS внутри SVG

```
<svg viewBox="0 0 200 200" width="200"  
height="200">  
  <style>  
    .block {  
      fill: #262335;  
    }  
    .block:hover {  
      fill: #ff7edb;  
    }  
  </style>  
  <rect class="block" width="200"  
height="200" x="0" y="0">  
</svg>
```

# Object

## JS

```
const bird = document.querySelector('#bird')
const wing = bird.contentDocument.querySelector('#wing')
```

## SMIL анимация

```
<svg viewBox="0 0 200 200" width="200" height="200">
  <rect class="block" width="200" height="200" x="0"
y="0">
    <animate attributeName="fill" dur="5000ms"
from="#262335" to="#ff7edb" fill="freeze" />
  </rect>
```

## CSS внутри SVG

```
<svg viewBox="0 0 200 200" width="200"
height="200">
  <style>
    .block {
      fill: #262335;
    }
    .block:hover {
      fill: #ff7edb;
    }
  </style>
  <rect class="block" width="200"
height="200" x="0" y="0">
</svg>
```

Способы встраивания

# Object



плюсы

JS через `contentDocument`

CSS внутри SVG

SMIL анимации

Способы встраивания

# Object



плюсы

JS через `contentDocument`

CSS внутри SVG

SMIL анимации



минусы

Вне SVG только SVG и CSS фильтры

Чувствителен к CORS

Способы встраивания

# Symbol sprite

## SVG

```
<svg xmlns="http://www.w3.org/2000/svg">
  <symbol id="icon1" viewBox="0 0 16 16">
    <rect width="16" height="16" x="0" y="0" fill="currentColor" />
  </symbol>
</svg>
```

## HTML

```
<svg class="icon">
  <use xlink:href="/sprite.svg#icon1" />
</svg>
```

Способы встраивания

# Symbol sprite

## SVG

```
<svg xmlns="http://www.w3.org/2000/svg">
  <symbol id="icon1" viewBox="0 0 16 16">
    <rect width="16" height="16" x="0" y="0" fill="currentColor" />
  </symbol>
</svg>
```

## CSS

```
.icon {
  color: #262335;
}
.icon:hover {
  color: #ff7edb;
}
```

## HTML

```
<svg class="icon">
  <use xlink:href="/sprite.svg#icon1" />
</svg>
```

Способы встраивания

# Symbol sprite



плюсы

Удобно применять

Можно красить иконки с помощью  
`currentColor`



Способы встраивания

# Symbol sprite



плюсы

Удобно применять

Можно красить иконки с помощью  
`currentColor`



минусы

Символьные спрайты  
плохо совместимы со сложной  
графикой

Способы встраивания

# Symbol sprite



Способы встраивания

# Symbol sprite

Способы встраивания

# Symbol sprite



Способы встраивания

# Inline SVG

Способы встраивания

# Inline SVG



плюсы

Доступен для CSS

Доступен для JS

Отсутствие дополнительных  
запросов

# Inline SVG



## плюсы

Доступен для CSS

Доступен для JS

Отсутствие дополнительных  
запросов



## минусы

Нет кэширования

Увеличенный размер  
страницы

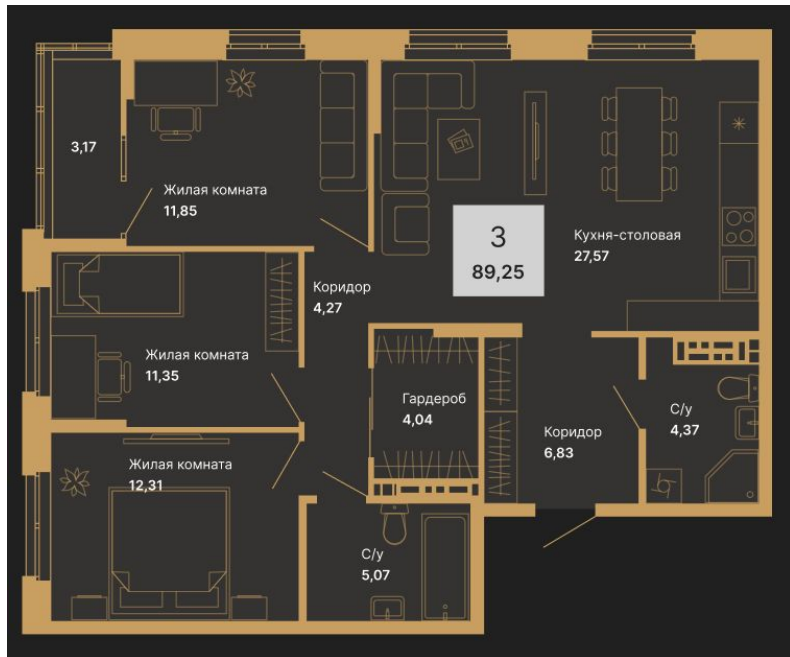
Могут возникать конфликты  
id и class

# Способы встраивания

|                  | Image | CSS Background | Object | Symbol sprite | Inline SVG |
|------------------|-------|----------------|--------|---------------|------------|
| Доступен для CSS | —     | —              | —      | —             | +          |
| Доступен для JS  | —     | —              | +      | —             | +          |



# Влияем на SVG с помощью CSS и JS





**Феникс**

разработка сайтов  
для застройщиков

# Производительность





**Феникс**

разработка сайтов  
для застройщиков

# Производительность

## Что влияет:

Реализация рендера

Оптимизация

Способ встраивания



# Производительность

## Оптимизация

- svgo ([github.com/svg/svgo](https://github.com/svg/svgo))
- Упрощение на уровне отрисовки

# Производительность. Оптимизация

Использование простых фигур, где возможно вместо <path>

```
<circle  
  fill="#FFFFFF"  
  stroke="#000"  
  cx="28.1"  
  cy="28.1"  
  r="27.6"  
>
```

```
<path  
  fill="#FFFFFF"  
  stroke="#000"  
  d="M55.7,28.1  
    c0,15.2-12.4,27.6-27.6,27.6  
    S0.5,43.3,0.5,28.1  
    S12.9,0.5,28.1,0.5  
    S55.7,12.9,55.7,28.1z"  
>
```

- Circle
- Rect
- Ellipse
- Line
- Polyline
- Polygon

# Производительность. Оптимизация

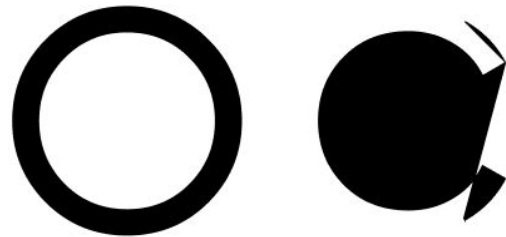
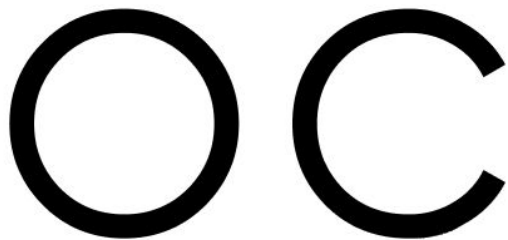
Преобразование текста в кривые

**Text**



# Производительность. Оптимизация

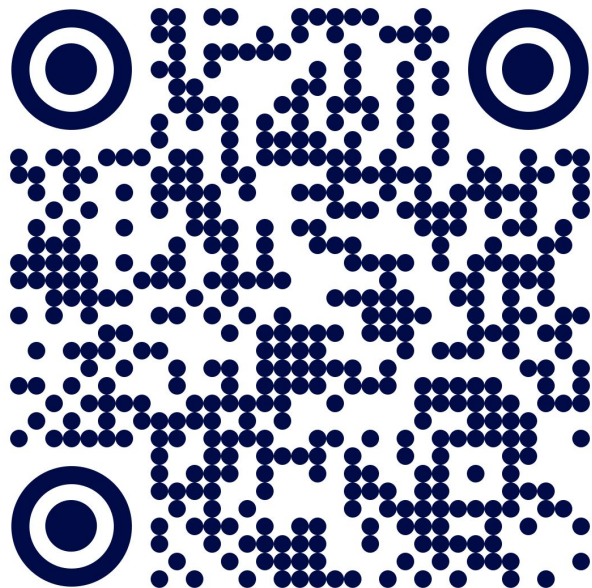
Преобразование текста в кривые



# Производительность. Оптимизация

Sara Soueidan: Tips for Creating and Exporting Better SVGs for the Web

Оригинал



[Ссылка](#)

Архив



[Ссылка](#)



# Производительность. Оптимизация

## Экспорт

```
<path d="M16.5717 13.9999C15.8574 13.9999 15.1717  
13.8856 14.5317 13.6742C14.4746 13.6571 14.4117  
13.6456 ... />
```

# Производительность. Оптимизация

## Экспорт

SVG Options

Styling: Internal CSS

Font: Convert To Outlines


Images: Preserve

Object IDs: Layer Names

Decimal: 2

☒ Minify ☒ Responsive

Show Code



Cancel

OK

# Производительность

Способ встраивания



Optimized



Unoptimized

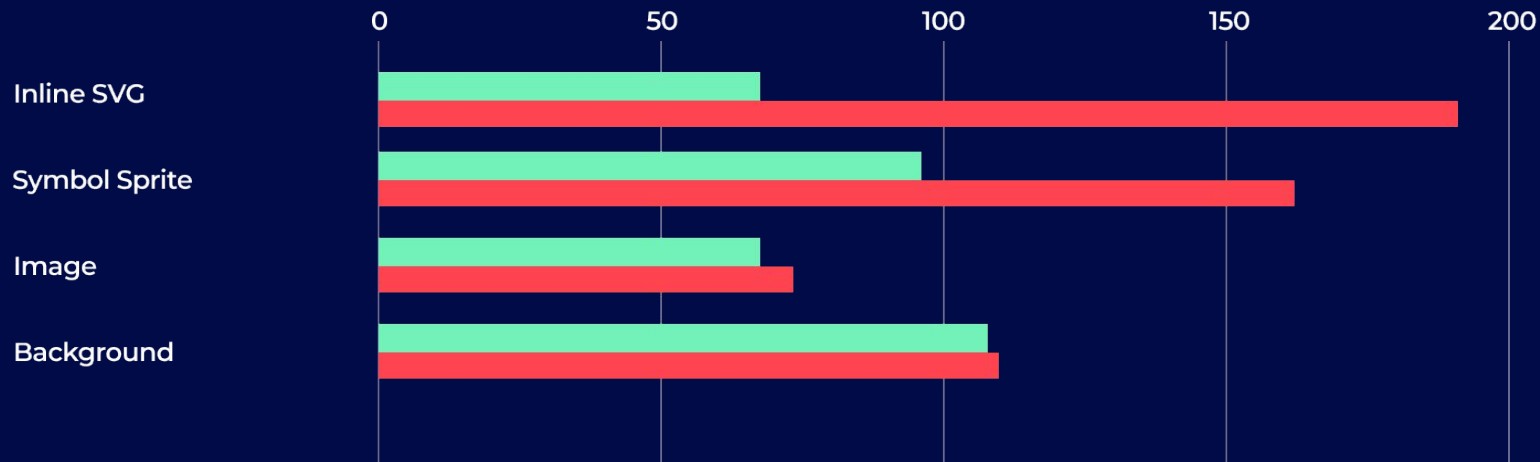


Chart: среднее время для рендера 1000 иконок (ms).

# Итоги

# Итоги

- Можно делать оптимизацию автоматически
- Оптимизировать со стороны дизайнера
- Правильно подбирать способ встраивания
- Исследуйте нативные свойства инструмента, который вы используете

# Итоги

Было:

- много кода, расчеты на js

# Итоги

Было:

- много кода, расчеты на js
- возможные баги с расположением объектов

# Итоги

Было:

- много кода, расчеты на js
- возможные баги с расположением объектов
- кривое положение всплывашек на сложных фигурах





# Итоги

Было:

- много кода, расчеты на js
- возможные баги с расположением объектов
- кривое положение всплывашек на сложных фигурах

Стало:

- решение в одну строку, нативно



# Итоги

Было:

- много кода, расчеты на js
- возможные баги с расположением объектов
- кривое положение всплывашек на сложных фигурах

Стало:

- решение в одну строку, нативно
- HTML элементы ведут себя как SVG



# Итоги

Было:

- много кода, расчеты на js
- возможные баги с расположением объектов
- кривое положение всплывашек на сложных фигурах



Стало:

- решение в одну строку, нативно
- HTML элементы ведут себя как SVG
- более точное и прогнозируемое положение всплывашек



# Вопросы

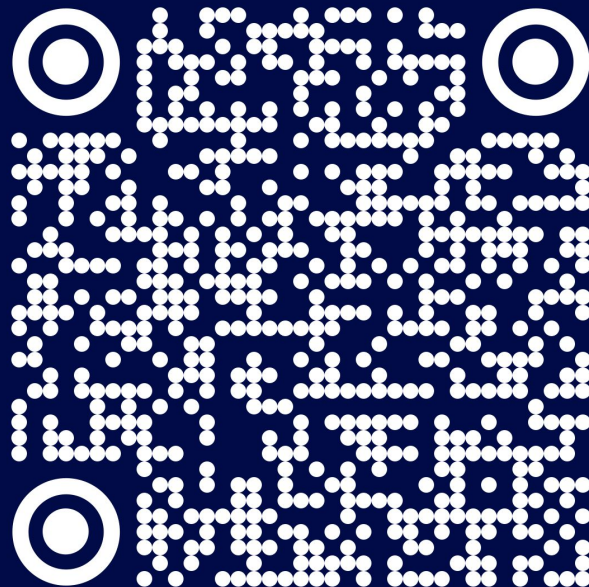


Константин Остров



@cos\_is

Материалы



[Ссылка](#)