

Homework #29

1

Что такое svg? В чем его преимущества?

SVG - формат векторной графики. Буквально его название значит «масштабируемая векторная графика» (Scalable Vector Graphics). SVG можно легко использовать в вебе, но сперва нужно во многом разобраться. SVG можно использовать для статических изображений, а также для анимаций и создания пользовательских интерфейсов.

Как и прочие языки, основанные на XML, SVG поддерживает использование таблиц стилей CSS, что позволяет отделить различные варианты визуального отображения от структуры данных.

Кроме того, таблицы стилей, которые вы используете в других языках разметки документов, могут содержать ссылку на SVG графику, в тех местах, где необходимо изображение. Например, в таблице стилей, для вашего HTML документа, можно указать ссылку (URL) на SVG графику в свойстве background.

Зачем вообще нужен SVG?

- Небольшие размеры файлов, отличное сжатие;
- Масштабирование до любого размера, без потери качества (разве что, при совсем маленьких размерах);
- Хорошо выглядит на ретине;
- Широкие возможности, которые предоставляют фильтры и интерактивность.

Опишите способы добавления svg на страницу. Какие преимущества дают те или иные способы?

I. Первый и самый простой — элемент `img` прямо в HTML-коде. Это в принципе самый эффективный способ загрузить любую картинку — браузеры заранее знают по HTML-коду, что она есть и начинают её подгружать.

```

```

Минус в том, что в таком SVG не будут работать скрипты и любые попытки взаимодействия с элементами внутри обречены. Файл будет как за стеклом: смотреть можно, а трогать нельзя. Хотя внутри всё остальное прекрасно работает, включая CSS-анимации.

Такой способ лучше всего подходит контентным изображениям, которым не нужно взаимодействие: логотипы, графики, схемы.

II. Второй способ — фоновая картинка в CSS. Причём неважно, зададите вы его элементу, псевдоэлементу или контентом вставьте — результат будет таким же, как с `img`: за стеклом, но внутри что-то работает.

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

Этот способ подходит для оформительской графики, которой не нужно взаимодействие: фоны, иконки и другая мелочь.

III. Третий способ, через `object`, наконец-то выбивает стекло между страницей и внутренностями SVG-файла. Работают скрипты, взаимодействие, анимация — если они описаны внутри SVG. Между тегами `object` можно вставить фолбэк, который покажется, если браузер не говорит на SVG.

```
<object type="image/svg+xml" data="picture.svg">  
    
</object>
```

На самом деле, вместо `object` можно даже использовать `iframe`, как если бы вы подключили другую страницу. Но `object` работает лучше и подстраивается под размеры картинки.

За гибкость приходится платить: из-за того, что это уже не просто графика и там можно скриптовать, к такому способу предъявляются другие требования безопасности. Например, картинку с другого домена просто так уже не вставить. Этот способ подходит, когда вам нужно вставить какую-то интерактивную графику: игрушки, графики и всякое сложное. Достаточно вспомнить, что когда-то через `object` вставлялись Flash-ролики. Спросите у родителей, что это такое.

IV. Четвёртый способ заработал, когда браузеры переписали свои HTML-парсеры по новому стандарту и содержимое SVG-файлов стало можно вставлять прямо на страницу, как любые другие теги.

```
<h1>Квадрат</h1>
<svg width="20" height="20">
  <rect fill="#fc0" width="20" height="20"/>
</svg>
```

С таким SVG можно делать то же, что и с обычными HTML-элементами: стили, скрипты — ну, вы сами знаете. Можно, например, менять цвет заливки при наведении и описывать всё в общих стилях.

```
<style>
  rect:hover { fill: #090; }
</style>
<svg>
  <rect fill="#fc0"/>
</svg>
```

Минус в том, что такие картинки не кэшируются отдельно от страницы