

课程介绍

► 课程目标

- 具备解决动画问题的专业能力
- 深度应用前端基础知识



使用anime.js



手撕anime.js



实践anime.js

► anime.js

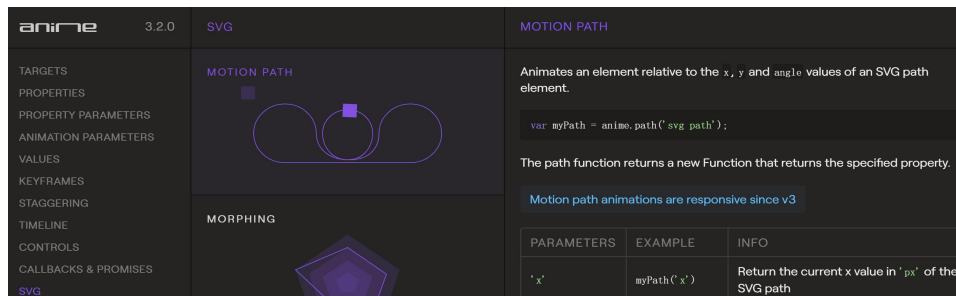
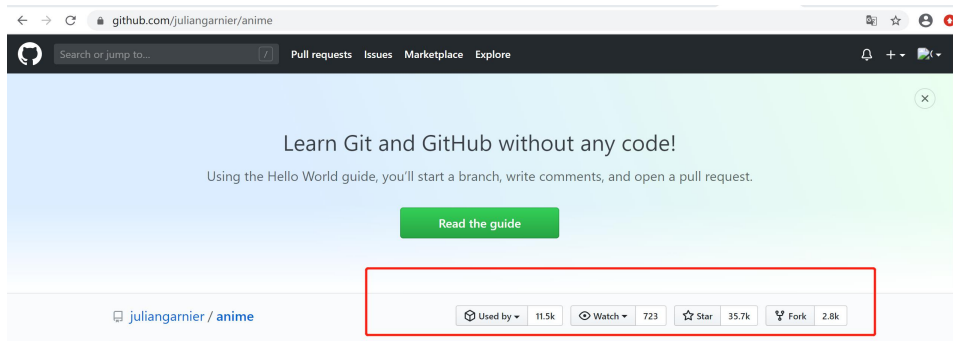
■ 简介

1. 一个最流行前端JS动画库
2. 超轻量级的JS动画库
3. 功能超强大的JS动画库

■ 资源链接

<https://github.com/juliangarnier/anime>

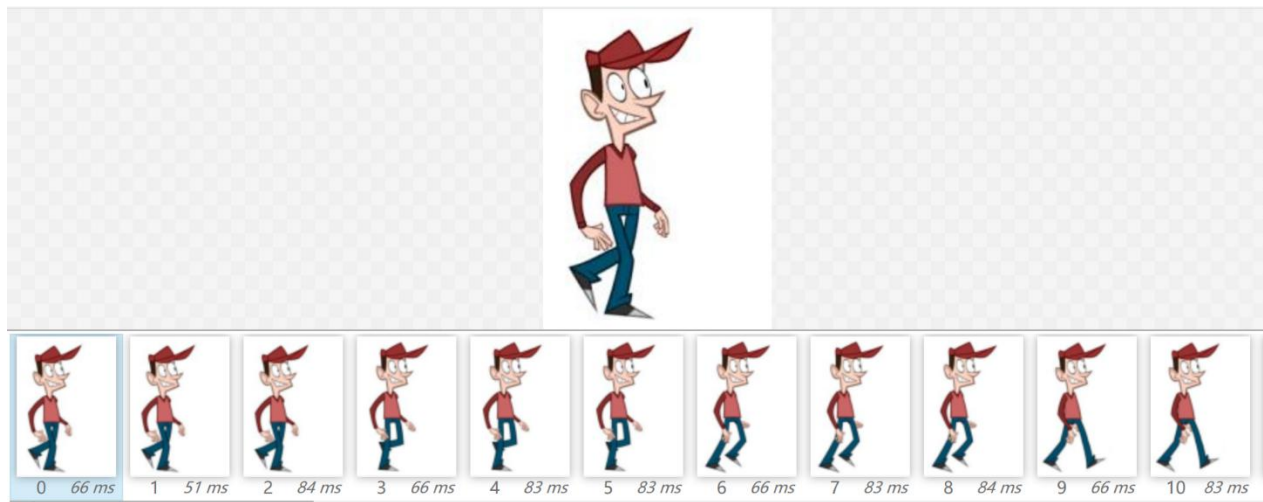
<https://animejs.com/documentation>



► 前端动画实现方式

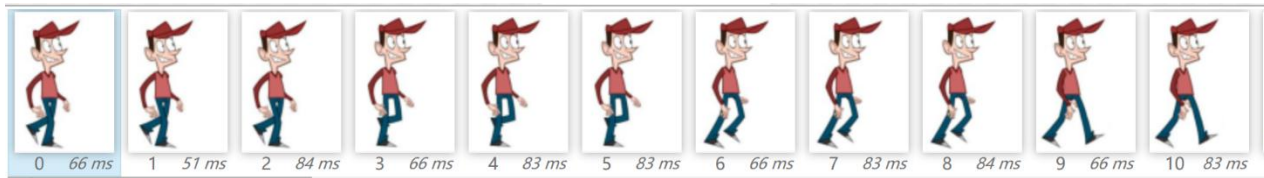
| | |
|--------------------------|------------------|
| CSS3-animation | 适合简单无需控制的动画效果 |
| JS-setInterval | 目前一般不使用这种方法制作动画 |
| JS-requestAnimationFrame | 适合需要控制，效果复杂度高的动画 |

► 动画的基本概念



► 动画的基本概念

- 帧：在动画过程中，每一幅静止画面即为一“帧”。
- 帧率：即每秒钟播放的静止画面的数量，单位是fps(Frame per second)。
- 帧时长：即每一幅静止画面的停留时间，单位一般是ms(毫秒)。
- 跳帧(掉帧/丢帧)：在帧率固定的动画中，某一帧的时长远高于平均帧时长，导致其后续数帧被挤压而丢失的现象。
- 10 FPS 达成基本视觉暂留 显示器刷新频率60+HZ



► CSS3动画与JS动画的对比

| | CSS3 | JS |
|-------|-----------|-----------|
| 性能 | 在某些特殊情况稍强 | 在某些特殊情况稍弱 |
| 动画流畅性 | 在某些特殊情况稍强 | 在某些特殊情况稍弱 |
| 兼容性 | 弱 | 强 |
| 可控性 | 弱 | 强 |
| 动画丰富性 | 弱 | 强 |
| 代码复杂度 | 相对不复杂 | 相对复杂 |
| 代码冗长性 | 笨重 | 简单 |

► setInterval 与 requestAnimationFrame 的对比

| | | setInterval | requestAnimationFrame |
|-----|--------|------------------------------------|-----------------------|
| 流畅度 | 刷新频率 | 动画刷新频率与系统屏幕刷新频率不匹配 | 动画刷新频率与系统屏幕刷新频率匹配 |
| 性能 | CPU 节能 | 页面切换或最小化动画仍在执行 | 页面切换或最小化动画停止 |
| | 函数节流 | 当实际帧时长超过设定帧时长，会造成回调堆积甚至浏览器内存耗尽页面崩溃 | 当实际帧时长超过设定帧时长，会跳过某些帧 |



Thanks

