**Недостатки setTimeout, setInterval**

- При написании кода, мы вручную устанавливаем, как именно будет происходить анимация покадрово. Если у пользователя компьютер перегружен. В таких случаях можно наблюдать, как какие-то кадры замирают. Вдобавок частота кадров стремится к какому-то значению. Вот тогда и происходит рассинхронизация: **setTimeout** принуждает перерисовывать кадры, не синхронно со сменой кадров у компьютера.

- Однако самая большая проблема состоит в том, что если открыто несколько вкладок браузера, и на одной из них есть анимация, то она все равно работает, активна эта вкладка или нет. В результате, все это ведет к избыточной нагрузке на компьютер.

**Функция requestAnimationFrame**

Функция **requestAnimationFrame** была создана для решения всех вышеперечисленных проблем. У нее достаточно высокая поддержка среди браузеров. Функция **requestAnimationFrame** позволяет нам запускать какие-то другие функции в качестве анимаций. Она берет вашу анимацию и подстраивает ее под частоту обновления вашего браузера. Делая так, что ваша анимация будет происходить в тот момент, когда обновляется страничка. Добиваясь плавности анимации и при этом не нагружая компьютер. А если на страничке прописано несколько анимаций, то запускаться они будут одновременно. Звучит хорошо, не правда ли?

Синтаксис:

let requestId = requestAnimationFrame(callback)

Такой вызов планирует запуск функции callback на ближайшее время, когда браузер сочтёт возможным осуществить анимацию.

Если в callback происходит изменение элемента, тогда оно будет сгруппировано с другими requestAnimationFrame и CSS-анимациями. Таким образом браузер выполнит один геометрический пересчёт и отрисовку, вместо нескольких.

Значение requestId может быть использовано для отмены анимации:

// отмена запланированного запуска callback

cancelAnimationFrame(requestId);

Функция callback имеет один аргумент – время прошедшее с момента начала загрузки страницы в миллисекундах. Это значение может быть получено с помощью вызова [performance.now()](https://developer.mozilla.org/ru/docs/Web/API/Performance/now).

Как правило, callback запускается очень скоро, если только не перегружен CPU или не разряжена батарея ноутбука, или у браузера нет какой-то ещё причины замедлиться.

Код ниже показывает время между первыми 10 запусками requestAnimationFrame. Обычно оно 10-20 мс: