JS実践

【スクロールに合わせて文字がアニメーションする】

IntersectionObserverを使用する。

.childクラスの要素が画面上に表示されているかをチェックし、表示されている場合はin view

表示されていない場合はout viewと表示する。

const child = document.querySelector('.child');

const cb = function(entries, observer){

  entries.forEach(entry => {

    if (entry.isIntersecting){

      console.log('in view');

    } else {

      console.log('out view');

    }

  });

}

const io = new IntersectionObserver(cb);

io.observe(child);

良く使用する記法

const cb = function(entries, observer){

  entries.forEach(entry => {

    if (entry.isIntersecting){

      // 画面内に要素が入っている場合はinviewクラスを付与。背景色緑

      entry.target.classList.add('inview');

      // 要素の追跡を終了する。

      observer.unobserve(entry.target);

    } else {

      // 画面内に要素が入っていない場合はinviewクラスを削除

      entry.target.classList.remove('inview');

    }

  });

スクロールAnimationを文字列に適用。（むずいからロジックだけ転記）

Domであるかどうかの判定

      if(el instanceof HTMLElement) {

        this.DOM.el = el;

      } else {

        this.DOM.el = document.querySelector(el);

      }

const els = document.querySelectorAll('.animate-title');

const cb = function(entries, observer){

    entries.forEach(entry => {

        if (entry.isIntersecting){

            const ta = new TextAnimation(entry.target);

            ta.animate();

            observer.unobserve(entry.target);

        } else {

    }

  });

}

class TextAnimation {

  constructor(el) {

      this.DOM = {};

      if(el instanceof HTMLElement) {

        this.DOM.el = el;

      } else {

        this.DOM.el = document.querySelector(el);

      }

      this.chars = this.DOM.el.innerHTML.trim().split("");

      this.DOM.el.innerHTML = this.\_splitText();

  }

  \_splitText() {

      return this.chars.reduce((acc, curr) => {

          curr = curr.replace(/\s+/, '&nbsp;');

          return `${acc}<span class="char">${curr}</span>`;

      }, "");

  }

  animate() {

      this.DOM.el.classList.toggle('inview');

  }

}

class TweenTextAnimation extends TextAnimation {

  constructor(el) {

      super(el);

      this.DOM.chars = this.DOM.el.querySelectorAll('.char');

  }

  animate() {

      this.DOM.el.classList.add('inview');

      this.DOM.chars.forEach((c, i) => {

          TweenMax.to(c, .6, {

              ease: Back.easeOut,

              delay: i \* .05,

              startAt: { y: '-50%', opacity: 0},

              y: '0%',

              opacity: 1

          });

      });

  }

}