**componentWillMount() 相当于vue的created**

## componentDidMount() 相当于vue 的 mounted

## shouldCompoentUpdata(nextProps, nextState)更新触发

这个可以做优化 判断和之前一样就不用每次重复randen

shouldComponentUpdate(nextProps, prevState) {

  const { author, body } = this.props.data;

const { author: newAuthor, body: newBody } = nextProps.data;

  if (author === newAuthor && body === newBody) {

   return false;

  }

  return true;

}

## componentWillUnmount从 DOM 卸载组件后调用。

## getDerivedStateFromProps 新的生命周期 初始化和更新都会触发

// 新的生命周期 初始化更新都会触发

  static getDerivedStateFromProps(prevProps, prevState) {

    console.log(prevProps)

    prevState.number = prevProps.text;

    return null;

  };

// nextState可以让这两个变成同步 即两个都会执行 不会覆盖

    // this.setState(nextState => {

    //   return {

    //     number: nextState.number + 1

    //   }

    // });

    // this.setState(nextState => {

    //   return {

    //     number: nextState.number + 2

    //   }

    // });

    // console.log(this.state.number);

函数式编程

import React, { useState, useEffect, useRef } from "react";

const [count, setCount] = useState(0);

    let id;

    useEffect(() => {

      console.log(111)

      id = setInterval(() => {

        if (count < 5) {

          setCount(count + 1);

        }

        clearInterval(id);

      }, 1000);

      return () => {

        // clearInterval(id);

      }

    }, [count]);