**React**

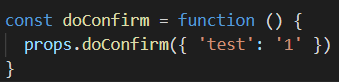
父子组件传值

子组件像父组件传值：

子组件：触发一个事件，事件上绑定方法，方法中使用 props.XXX（自己定义的名字，在父组件中使用 作为组件的属性名， 属性名=接受数据的方法(子组件传过来的值)）

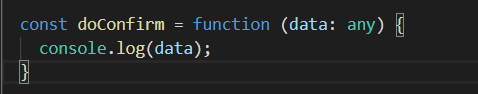
子组件





父组件





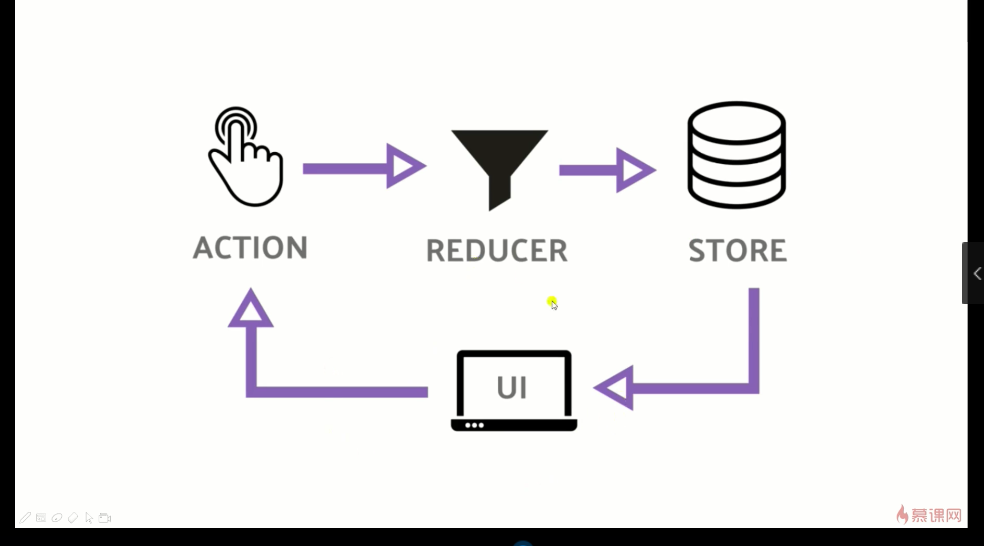
父组件向子组件传值：

Hooks

让函数组件具有类组件的能力

Redux 存取公用数据

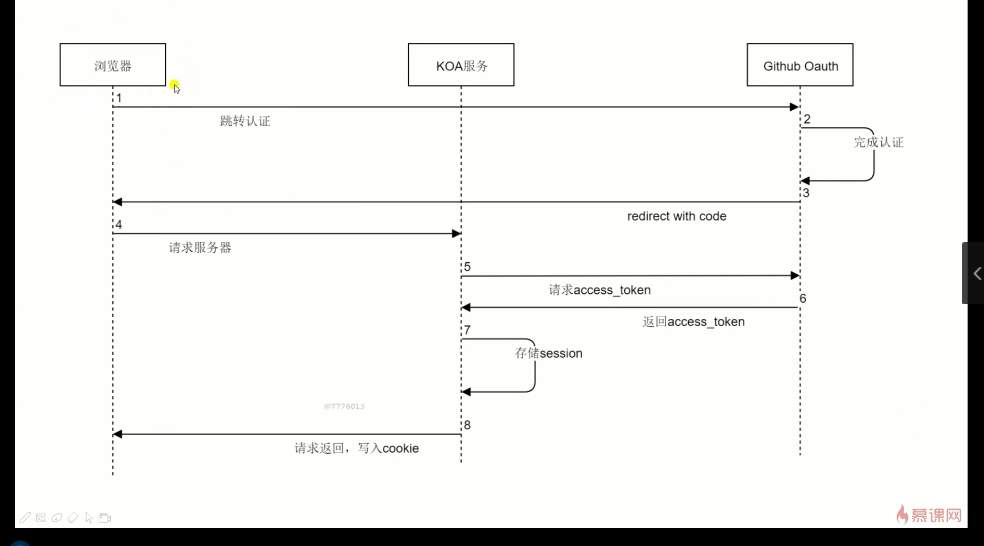
单向数据流状态管理工具



Hoc

接受组件作为参数并返回新的组件

OAuth



React-next.js-koa

同构

<https://segmentfault.com/a/1190000013609085>



Nextjs

Npm i -g Create-next-app 全局安装

Create-next-app XXX(项目名) 创建项目

Javascript 高级程序设计

共6种数据类型

5种简单数据类型

Undefined， Null，Boolean，Number，String

1种复杂数据类型

Object

typeof操作符 返回值一个字符串

“undefined”, “number”, “string”, “boolean”, “function” , “object”(对象或者null)

对于尚未声明的变量，只能执行一项操作，即typeof操作符检测其数据类型

typeof a // “undefined” a为未定义的变量

undefined 和 null 都是只有一个值

null值表示空对象指针 typeof null // “object”

Boolean(0) Boolean(NaN) // false

typeof NaN // “number”

NaN 表示一个本来要返回数值的操作 未返回值的情况

任何涉及NaN的操作都会返回NaN; NaN与任何值都不相等，包括NaN本身。

只有 0/0 返回NaN ，正数/0 返回Infinity，负数/0 返回 -Infinity

NaN == NaN // false

Number(undefined) // NaN

Number(不存在任何有意义的数字值) // NaN

parseInt() 如果第一个非空字符不是数字字符或者负号，parseInt() 返回 NaN

在基于对象调用isNaN()函数时，会首先调用对象的valueOf()方法，然后确定该方法的返回值是否可以转为数值。如果不能，则基于这个返回值再调用toString()方法，在测试返回值。

null 和 undefined 没有toString()方法

Object

一组数据和功能的集合

constructor: 保存着用于创建当前对象的函数

hasOwnProperty(propertyName): 用于检查给定的属性在当前对象实例中（而不是在实例的原型中）是否存在

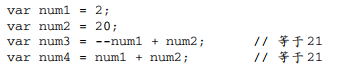
isPrototypeOf(object): 用于检查的对象是否是传入对象的原型

propertyIsEnumerable(propertyName): 用于检查给定的属性是否能够使用for-in来枚举

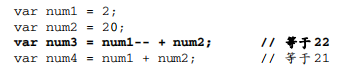
一元操作符 ++ 、--

只能操作一个值的操作符

前置递增和递减：与执行语句的优先级相等，整个语句从左至右被求值



后置递增和递减：在包含他们的语句被求值之后才执行



Object

当从一个变量向另一个变量复制应用类型的值时，同时也会将存储在变量对象中的值复制一份到为新变量分配的空间中。这个值的副本实际上是一个指针，而这个指针指向存储在堆中的一个对象。复制操作结束后，两个变量实际上将引用同一个对象。因此改变其中一个变量将影响另一个变量

