link: null
title: 珠峰架构师成长计划
description: null
keywords: null
author: null
date: null
publisher: 珠峰架构师成长计划
stats: paragraph=69 sentences=162, words=1077

1. React Hooks

- Hook 是 React 16.8 的新增特性。它可以让你在不编写 class 的情况下使用 state 以及其他的 React 特性

1.1 Hooks 优点

- 可以抽离公共方法和逻辑，提高代码的可复用性
- 函数式组件更简洁，开发效率更高

1.2 自定义 Hook

- 通过自定义 Hook，可以将组件逻辑提取到可重用的函数中
- 自定义 Hook 是一个函数，其名称以 `use` 开头，函数内部可以调用其他的 `Hook`

1.3 初始化项目

```
create-react-app zhufeng_custom_hooks
cd zhufeng_custom_hooks
npm i express cors morgan bootstrap@3 react-router-dom --save
```

2.useRequest

2.1 index.js

src\index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import 'bootstrap/dist/css/bootstrap.css'
import {BrowserRouter,Route,Link} from 'react-router-dom';
import Table from './Table';
import Drag from './Drag';
import Form from './Form';
import Circle from './Circle';
ReactDOM.render(
  <div className="container">
    <div className="row">
      <div className="col-md-12" style={{ padding: 10 }}>
        <BrowserRouter>
          <ul className="nav nav-tabs">
            <li><Link to="/table">TableLink>li>
            <li><Link to="/drag">DragLink>li>
            <li><Link to="/form">FormLink>li>
            <li><Link to="/circle">CircleLink>li>
          ul>
          <Route path="/table" component={Table}/>
          <Route path="/drag" component={Drag}/>
          <Route path="/form" component={Form}/>
          <Route path="/circle" component={Circle}/>
        BrowserRouter>
      div>
    div>
  div>,
  document.getElementById('root')
);
```

2.2 Table.js

src\Table.js

```
import React from 'react';
import useRequest from './hooks/useRequest';
const URL = 'http://localhost:8000/api/users';
export default function Table() {
    const [data, options,setOptions] = useRequest(URL);
    const { currentPage, totalPage, list } = data;
    return (
        <>
            <table className="table table-striped">
                <thead>
                    <tr><td>IDtd><td>姓名td>tr>
                thead>
                <tbody>
                    {
                        list.map(item => (<tr key={item.id}><td>{item.id}td><td>{item.name}td>tr>))
                    }
                tbody>
            table>
            <nav>
                <ul className="pagination">
                    {currentPage>1&&(
                        <li>
                            <button className="btn btn-default" href="#" onClick={() => setOptions({ ...options,currentPage: currentPage - 1 })}>
                                <span ><<span>
                            button>
                        li>
                    )}
                    {
                        new Array(totalPage).fill(0).map((item, index) => (
                            <li><button className={index+1===currentPage?'btn btn-success':'btn btn-default'} key={item.id} onClick={() => setOptions({
...options,currentPage: index + 1 })}>{index + 1}button>li>
                        ))
                    }
                    {
                        currentPage<totalPage&&(
                            <li>
                                <button className="btn btn-default"  onClick={() => setOptions({...options, currentPage: currentPage + 1 })}>
                                    <span>>>span>
                                button>
                            li>
                        )
                    }
                ul>
            nav>
        </>
    )
}
```

2.3 useRequest.js

src\hooks\useRequest.js

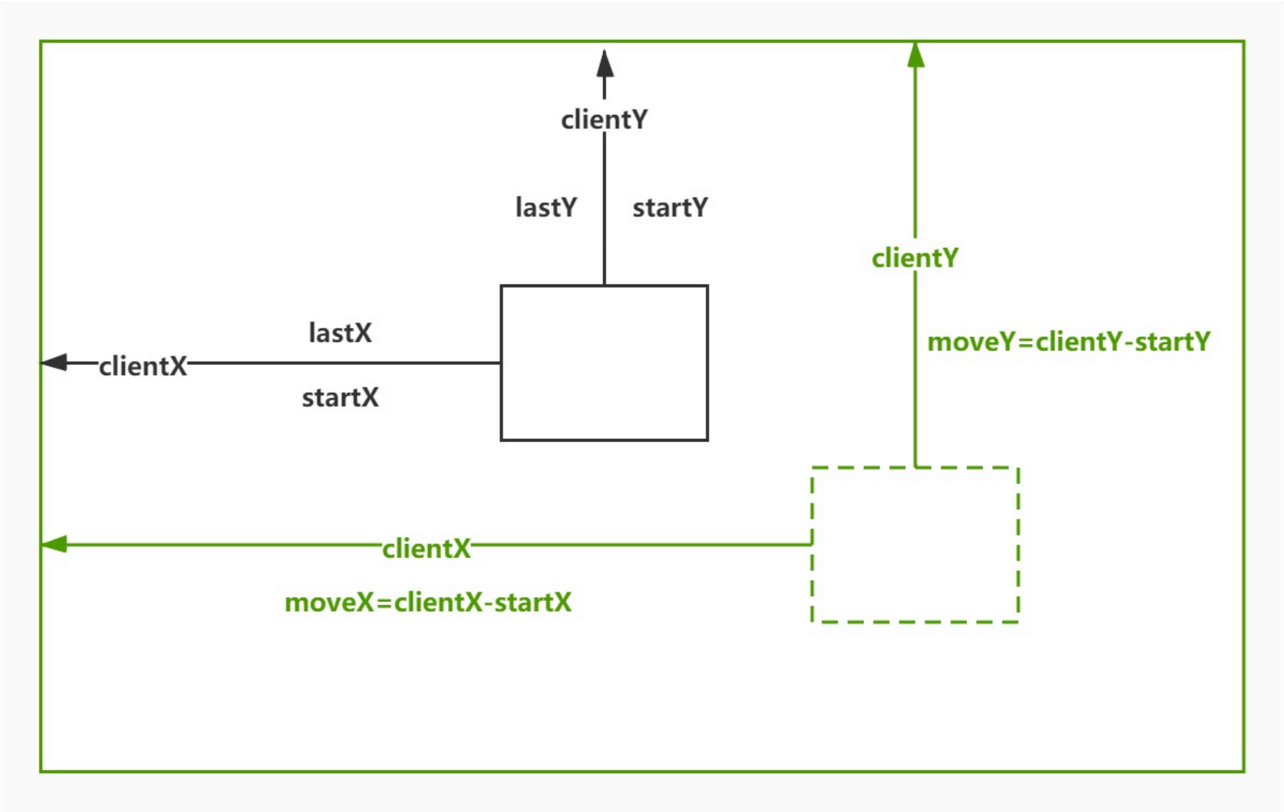
```
import { useState, useEffect } from 'react';
function useRequest(url) {
    let [options, setOptions] = useState({
        currentPage: 1,
        pageSize: 5
    });
    let [data, setData] = useState({
        totalPage: 0,
        list: []
    });
    function getData() {
        let { currentPage, pageSize } = options;
        fetch(`${url}?currentPage=${currentPage}&pageSize=${pageSize}`)
            .then(response => response.json())
            .then(result => {
                setData({...result});
            });
    }
    useEffect(getData, [options, url]);
    return [data,options, setOptions];
}

export default useRequest;
```

2.4 api.js

api.js

```
let express = require('express');
let cors = require('cors');
let logger = require('morgan');
let app = express();
app.use(logger('dev'));
app.use(cors());
app.get('/api/users', function (req, res) {
    let currentPage = parseInt(req.query.currentPage);
    let pageSize = parseInt(req.query.pageSize);
    let total=25;
    let list = [];
    let offset = (currentPage-1)*pageSize;
    for (let i = offset; i < offset + pageSize; i++) {
        list.push({ id: i + 1, name: 'name' + (i + 1) });
    }
    res.json({
        currentPage,
        pageSize,
        totalPage:Math.ceil(total/pageSize),
        list
    });
});
app.listen(8000,()=>{
    console.log('sever started at port 8000');
});
```

3.useDrag



3.1 基础

3.1.1 触摸事件

事件名称 描述 是否包含 touches 数组 touchstart 触摸开始发 是 touchmove 滑动时接触点改变 是 touchend 手指离开屏幕时触摸结束 是

3.1.2 触摸列表

参数 描述 touches 当前位于屏幕上的所有手指的列表 targetTouches 位于当前DOM元素上手指的列表

3.1.3 Touch对象

参数 描述 clientX 触摸目标在视口中的x坐标 clientY 触摸目标在视口中的y坐标 pageX 触摸目标在页面中的x坐标 pageY 触摸目标在页面中的y坐标

3.2 实现

3.2.1 index.js

src\index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import 'bootstrap/dist/css/bootstrap.css'
import Drag from './Drag';
ReactDOM.render(
  <div className="container">
    <div className="row">
      <div className="col-md-12">
        {<Drag />}
      div>
    div>
  div>,
  document.getElementById('root')
);
```

3.2.2 src\Drag.js

src\Drag.js

```
import React from 'react';
import useDrag from './hooks/useDrag';
let style = {width:'100px',height:'100px',borderRadius:'50%'};
export default function Drag() {
    const [style1, dragRef1] = useDrag()
    const [style2, dragRef2] = useDrag()
    return (
        <>
            <div
                ref={dragRef1}
                style={{...style,backgroundColor:'red',transform: `translate(${style1.x}px, ${style1.y}px)` }}
            >div>
             <div
                ref={dragRef2}
                style={{...style,backgroundColor:'green',transform: `translate(${style2.x}px, ${style2.y}px)` }}
            >div>
        </>
    )
}
```

3.2.3 useDrag.js

src\hooks\useDrag.js

```
import { useLayoutEffect, useState, useRef } from 'react';
function useDrag() {
    const positionRef = useRef({
        currentX: 0, currentY: 0,
        lastX: 0, lastY: 0
    })
    const moveElement = useRef(null);
    const [, forceUpdate] = useState({});
    useLayoutEffect(() => {
        let startX, startY;
        const start = function (event) {
            const { clientX, clientY } =  event.targetTouches[0];
            startX = clientX;
            startY = clientY;
            moveElement.current.addEventListener('touchmove', move);
            moveElement.current.addEventListener('touchend', end);

        }
        const move = function (event) {
            const { clientX, clientY } = event.targetTouches[0];
            positionRef.current.currentX = positionRef.current.lastX + (clientX - startX);
            positionRef.current.currentY = positionRef.current.lastY + (clientY - startY);
            forceUpdate({});
        }
        const end = (event) => {
            positionRef.current.lastX = positionRef.current.currentX;
            positionRef.current.lastY = positionRef.current.currentY;
            moveElement.current.removeEventListener('touchmove', move);
                moveElement.current.removeEventListener('touchend', end);
        }
        moveElement.current.addEventListener('touchstart', start);

    }, []);
    return [{ x: positionRef.current.currentX, y: positionRef.current.currentY }, moveElement]
}

export default useDrag;
```

4.useForm

4.1 src\index.js

src\index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import 'bootstrap/dist/css/bootstrap.css'
import Form from './Form';
ReactDOM.render(
  <div className="container">
    <div className="row">
      <div className="col-md-12">
        {<Form />}
      div>
    div>
  div>,
  document.getElementById('root')
);
```

4.2 src\Form.js

src\Form.js

```
import React from 'react';
import useForm from './hooks/useForm';
export default function Form() {
    const [formData, setFormValue, resetFormValues] = useForm({username:'',email:''});
    return (
        <div className="panel">
            <div className="panel-body">
                <form>
                    <div className="form-group">
                        <label >用户名label>
                        <input
                            className="form-control"
                            placeholder="用户名"
                            value={formData.username}
                            onChange={(event) => setFormValue('username', event.target.value)} />
                    div>
                    <div className="form-group">
                        <label >邮箱label>
                        <input
                            className="form-control"
                            placeholder="邮箱"
                            value={formData.email}
                            onChange={(event) => setFormValue('email', event.target.value)}
                        />
                    div>
                    <button type="button" className="btn btn-default" onClick={() => console.log(formData)}>提交button>
                    <button  type="button" className="btn btn-default" onClick={resetFormValues}>重置button>
                form>
            div>
        div>
    )
}
```

4.3 useForm.js

src\hooks\useForm.js

```
import { useState } from 'react';
function useForm(values) {
    const [formData, setFormData] = useState(values);
    const setFormValue = (key, value) => {
        setFormData({...formData,[key]:value});
    }
    const resetFormValues = () => {
        setFormData(values);
    }
    return [formData, setFormValue, resetFormValues];
}

export default useForm;
```

5.useAnimation

5.1 src\index.js

src\index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import 'bootstrap/dist/css/bootstrap.css'
import Circle from './Circle';
ReactDOM.render(
  <div className="container">
    <div className="row">
      <div className="col-md-12" style={{ padding: 10 }}>
        <Circle/>
      div>
    div>
  div>,
  document.getElementById('root')
);
```

5.2 src\Circle.js

src\Circle.js

```
import useAnimation from './hooks/useAnimation';
import './Circle.css';
function Circle() {
  const [className, start] = useAnimation('circle','active');
    return (
      <div className={className} onClick={start}>div>
    );
}
export default Circle;;
```

5.3 Circle.css

src\Circle.css

```
.circle {
    width : 200px;
    height : 200px;
    background-color : gray;
    transition: all 2s;
}
.circle.active {
    background-color : green;
}
```

5.4 useAnimation.js

src\hooks\useAnimation.js

```
import {useState} from 'react';
function useAnimation(initialClassName,activeClassName) {
    const [className, setClassName] = useState(initialClassName);
    function start() {
        if (className === initialClassName) {
            setClassName(`${initialClassName} ${activeClassName}`);
        }else{
            setClassName(`${initialClassName}`);
        }
    }
    return [className, start];
}
export default useAnimation;
```