re-base

<https://github.com/tylermcginnis/re-base>

<https://www.youtube.com/watch?v=vmLaZafaw9E>

Приклад з авторизацією

<https://github.com/coderjourney/chords/blob/master/src/App.js>

Встановлення

npm install --save re-base firebase

|  |  |  |
| --- | --- | --- |
| Створюємо проект |  |  |
| Додаємо проект до веб-додатку |  |  |
| Створюємо конфігураційний файл |  | ------------------------ base.js ---------------------------- |
| Створюємо компонент і синхронізуємо state з базою даних використвоуючи методи | componentWillMount(){  this. посилання\_на\_базу=base.syncState( ‘назва\_у\_базі’ {  context: де\_знаходиться\_state,  state : ‘назва\_колекції\_у\_state’,  asArray:true //Якщо хочемо зберігати масив зі стейта  }) ;  }  componentWillUnmount(){  base.removeBinding(this.посилання на базу);  } |  |
|  |  |  |
|  |  |  |
|  |  |  |

Приклад

|  |  |
| --- | --- |
| Додаємо до проекта на firebase додаток |  |
| Реєструємо додаток  І копіюємо налаштування проекта |  |
| Створення  Base.js | import Rebase from 're-base'  import firebase from 'firebase'  var firebaseConfig = {  apiKey: "", 🡨 СЮДИ КОМПІЮЄМО ВАШІ НАЛАШТУВАННЯ (ЯКІ СКОПІЮВАЛИ З FIREBASE)  authDomain: "",  databaseURL: "",  projectId: "",  storageBucket: "",  messagingSenderId: "",  appId: ""  };  // Initialize Firebase  var app = firebase.initializeApp(firebaseConfig);  var base = Rebase.createClass(app.database());  export default base |
| Використання base | import React, { Component } from 'react';  import base from './base' 🡨 1. Підключаємо base.js  class Tasks extends Component {  constructor(props) {  super(props);  this.state = {  tasks:[]  }  }  componentDidMount(){ 🡨 2. Встановлюємо з’єднання бази зі state  this.db=base.syncState(`tasks`, {  context: this,  state: 'tasks',  asArray:true  });  }  componentWillUnmount(){ 🡨 3. Розриваємо з’єднання бази з state  base.removeBinding(this.db)  }  addTask=()=>{  this.setState({  tasks:[...this.state.tasks,{task:this.refs.inp.value,range:1}]  })  }  render() {  const {tasks}=this.state  {console.log(tasks)}  return (  <>  {tasks.length?(  <ul>  {tasks.map((el,i)=>(  <li key={i}> { el.task }- {el.range} </li>  ))}  </ul>):'no tasks'}  <div>  Add  <input ref='inp'/>  <button onClick={this.addTask}>  add  </button>  </div>  </>  );  }  }    export default Tasks; |
|  |  |
|  |  |