## $ dva new react-news-dva

## $ yarn install

$ yarn start

# 组件的生命周期

# **路由 Router**

**import** { Switch, Route, routerRedux, Redirect } **from 'dva/router'**;

*RouterIndex1*({ history, app, location })

# DVA框架 model层即对redux进行了整合

**import** {routerRedux} **from 'dva/router'**

**结构：**

**export default** {  
 **namespace**: **'pageModel'**,  
 **state**: {},

**effects**: {},

**reducers**: {},

**subscriptions**: {},

}

## index.js

**import** *dva* **from 'dva'**;  
**import 'antd/dist/antd.css'**;  
*// 1. Initialize***const** app = *dva*();  
*// 2. Plugins  
// app.use({});  
// 3. Model*app.model(***require***(**'./models/store'**).**default**);  
*// 4. Router*app.router(***require***(**'./routes/index'**).**default**);  
*// 5. Start*app.start(**'#root'**);

## 获取state中的值用：

**const** *mapStateToProps* = (state) => {  
 **return** {  
 **tabName**: state.**store**.**tabName**,  
 };  
};

上面的state.pageModel中的pageModel

是空间名，任意名都可以。

## 改变state的状态：

**const** *mapDispatchToProps* = (dispatch) => {  
 **return** {  
 changeTab: (tabName) => {  
 dispatch({**type**: **'store/changeTab'**, **payload**: { **tabName**: tabName }})  
 },  
 currentTabClick: (tabName) => {  
 dispatch({**type**: pageModel**/currentTabClick'**, **payload**: { **tabName**: tabName }})  
 },  
 getTabName: () => {  
 dispatch({**type**: pageModel**/getTabName'**})  
 }  
 }  
};

## 3、每个model都有一个**namespace**: **'xxx'**

**Effects：异步操作 ，使用时需要加上空间名，如：**

\*getTabName({payload}, {select}){  
 **const** tabName = **yield** select(state => state.pageModel.**tabName**);  
 **return** tabName;  
},

**Reducers:同步操作，不必加空间名**

currentTabClick(state, action){ *//积累tab点击信息* state.**tabName** = action.**payload**.**tabName**;  
 **return** state;  
},