

redux结构知识点

redux 源码结构

```
|-- utils
    |-- warnings.js
|-- applyMiddleware.js
|-- bindActionCreators.js
|-- combineReducers.js
|-- compose.js
|-- createStore.js
|-- index.js
```

index.js 中导出5个模块，即外部可用的：

```
export {
  createStore,
  combineReducers,
  bindActionCreators,
  applyMiddleware,
  compose
}
```

createStore、applyMiddleware 及 compose 这3个模块存在相互调用的关系

1. 每个模块的作用：

- createStore 函数作用是生成一个 store 对象，具有5个方法：

```
return {
  dispatch, // 传入 action, 调用 reducer 及触发 subscribe 绑定的监听函数
  subscribe,
  getState,
  replaceReducer, // 用新的 reducer 代替当前的 reducer, 使用不多
  [$$observable]: observable
```

```
}
```

- `applyMiddleware` 函数作用是对 `store.dispatch` 方法进行增强和改造，使得在发出 Action 和执行 Reducer 之间添加其他功能。
- `compose` 函数是 `applyMiddleware` 函数的核心，会形成串联的函数调用关系，用于增强 `dispatch` 方法。

2. 模块之间的调用关系