



리액트로 구현하는 이더리움 ERC-721



1. reducer

- 사용자의 reducer 추가

```
<DrizzleProvider options={drizzleOptions} store={store}>
```

```
  <LoadingContainer>
```

```
    <MainContainer />
```

```
  </LoadingContainer>
```

```
</DrizzleProvider>
```

store를 새로 생성하여
DrizzleProvider에 전달해야 한다.

```
const store = createStore(  
  reducer, -----  
  initialState,  
  compose(  
    applyMiddleware(  
      thunkMiddleware,  
      routingMiddleware,  
      sagaMiddleware  
    )  
  )  
)
```

```
const reducer = combineReducers({  
  customReducer: customReducer,  
  routing: routerReducer,  
  ...drizzleReducers  
})  
  
export default reducer
```

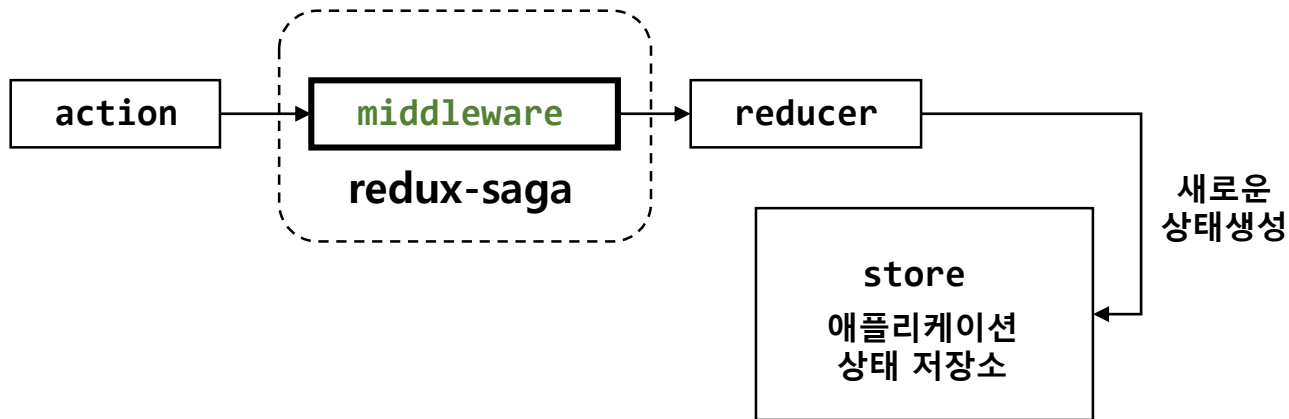
2. redux-saga

- 비동기적인 처리를 위해 사용하는 라이브러리
- drizzle에서는 이더리움과의 통신을 위해 사용

```
`-- drizzle@1.2.4  
  `-- redux-saga@0.16.2
```



<https://redux-saga.js.org/>



2. redux-saga

```
this.deedToken.methods.totalSupply.cacheCall();
```

```
contract.store.dispatch({  
  type: 'CALL_CONTRACT_FN',  
  contract,  
  fnName,  
  fnIndex,  
  args,  
  argsHash  
})
```

redux-saga

```
function * contractsSaga () {  
  yield takeEvery('SEND_CONTRACT_TX', callSendContractTx)  
  yield takeEvery('CALL_CONTRACT_FN', callCallContractFn)  
  yield takeEvery('CONTRACT_SYNCING', callSyncContract)  
  yield takeEvery('LISTEN_FOR_EVENT', callListenForContractEvent)  
  yield takeEvery('ADD_CONTRACT', addContract)  
}
```

```
const callResult = yield call(txObject.call, callArgs)  
yield put({ type: 'GOT_CONTRACT_VAR', ...dispatchArgs })
```

reducer

```
if (action.type === 'GOT_CONTRACT_VAR') {  
  return {  
    ...state,  
    [action.name]: {  
      . . . . .  
      value: action.value  
    }  
  }  
}
```

①

②

③

2. redux-saga

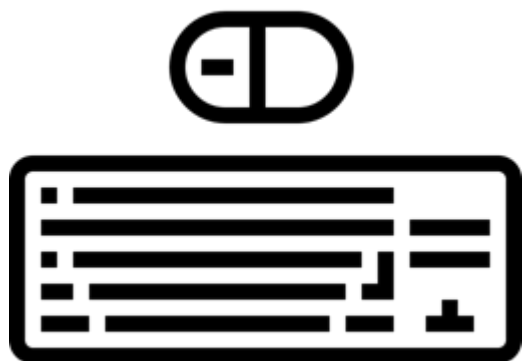
- `yield takeEvery` : (saga helper) 모든 action을 watch하여 비동기 처리
`yield takeEvery('CALL_CONTRACT_FN', callCallContractFn)`
- `yield call` : 함수 호출 - 컨트랙트 함수 비동기 호출
`yield call(txObject.call, callArgs)`
- `yield put` : action을 dispatch
`yield put({ type: 'GOT_CONTRACT_VAR', ...dispatchArgs })`

2. redux-saga

- 사용자의 saga 함수 추가

```
export default function* root() {  
  
  const sagas = [...drizzleSagas.map(saga => fork(saga)), fork(customSaga)];  
  yield all(  
    //drizzleSagas.map(saga => fork(saga))  
    sagas  
  );  
}
```

다음 시간에는...



- drizzle middleware 