signalStore

Deep Dive

Function Roles and Lifecycle

Function Categories

- signalStore
- Feature creators
- Feature parameters
- Computed expressions
- Method implementation
- Hooks

```ts
some.store.ts

export const SomeStore = signalStore(
    withState(initialSomeSlice),
    withComputed(store => ({
        isLarge: computed(() => store.someValue() > 100)
    })),
    withMethods(store => ({
        increment: () => patchState(store, updaters.increment()),
        decrement: () => patchState(store, updaters.decrement())
    })),
    withHooks(store => ({
        onInit: () => console.log('SomeStore initialized')
    }))
);
```

Function Categories

- signalStore
- Feature creators
- Feature parameters
- Computed expressions
- Method implementation
- Hooks

```typescript
export const SomeStore = signalStore(
    withState(initialSomeSlice),
    withComputed(store => ({
        isLarge: computed(() => store.someValue() > 100)
    })),
    withMethods(store => ({
        increment: () => patchState(store, updaters.increment()),
        decrement: () => patchState(store, updaters.decrement())
    })),
    withHooks(store => ({
        onInit: () => console.log('SomeStore initialized')
    }))
);
```

some.store.ts

Function Categories

- **signalStore**
- Feature creators
- Feature parameters
- Computed expressions
- Method implementation
- Hooks

```ts
some.store.ts

export const SomeStore = signalStore(
    withState(initialSomeSlice),
    withComputed(store => ({
        isLarge: computed(() => store.someValue() > 100)
    })),
    withMethods(store => ({
        increment: () => patchState(store, updaters.increment()),
        decrement: () => patchState(store, updaters.decrement())
    })),
    withHooks(store => ({
        onInit: () => console.log('SomeStore initialized')
    }))
);
```

Function Categories

- signalStore

signalStore(feature , feature , feature ,…)

⇨ class
constructor

feature

feature

feature

What is a "Feature"?

- Feature is a "**Partial Constructor**"

feature(partial store instance)

⇨ partial store instance ++

Store construction

empty store → feature → partial store + → feature → partial store ++ → feature → store