**------NGRX(Reactive State for Angular)------**

NgRx Store provides **reactive state management** for Angular apps inspired by Redux. Unify the events in your application and derive state using RxJS.

**ISOLATE SIDE EFFECTS:-** NgRx Effects gives you a framework for isolating side effects from your components by connecting observables of actions to your store.

search$ = createEffect(() =>

this.actions$.pipe(

ofType(BookActions.search),

exhaustMap(action =>

this.googleBooksService.search(action.query)

) ));

**@ngrx/store:** Store is RxJS powered state management for Angular applications, inspired by Redux. Store is a controlled state container designed to help write performant, consistent applications on top of Angular.

**Key concepts**

**Actions:** Actions describe unique events that are dispatched from components and services.

**Reducers:** State changes are handled by pure functions called reducers that take the current state and the latest action to compute a new state.

**Selectors:** Selectors are pure functions used to select, derive and compose pieces of state.

**State:** State is accessed with the Store, an observable of state and an observer of actions.

Installation