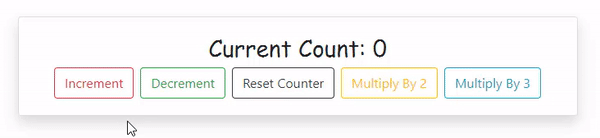
**Demo Steps:**

**Problem Statement:-** Create an angular app with **counter**component as shown using ngRx core concepts i.e. **Store, Actions, Reducers and Selectors.**



1. Create an **angular app** and install **@ngrx/store** in the application using the below command:

1. npm install @ngrx/store --save

2. Create **app/Store/actions/counter.actions.ts** within **src**with the following code:

1. import { createAction, props } from '@ngrx/store';
2. export const increment = createAction('[Counter Component] Increment');
3. export const decrement = createAction('[Counter Component] Decrement');
4. export const multiply = createAction('[Counter Component] Multiply', props<{ multiplyBy: number }>());
5. export const reset = createAction('[Counter Component] Reset');

3. Create **app/Store/reducers/counter.reducer.ts** within **src**with the following code:

1. import { createReducer, on } from '@ngrx/store';
2. import { increment, decrement, reset, multiply } from '../actions/counter.action';
3. export const initialState = {
4. counter: 0
5. };
6. const \_counterReducer = createReducer(initialState,
7. on(increment, state => { return { ...state, counter: state.counter + 1 } }),
8. on(decrement, state => { return { ...state, counter: state.counter - 1 } }),
9. on(multiply, (state, { multiplyBy }) => { return { ...state, counter: state.counter \* multiplyBy } }),
10. on(reset, state => initialState),
11. );
12. export function counterReducer(state, action) {
13. return \_counterReducer(state, action);
14. }

4. Create**app/Store/selectors/counter.selector.ts**within **src**with the following code:

1. import { createSelector } from '@ngrx/store';
2. export interface FeatureState {
3. counter: number;
4. }
5. export interface AppState {
6. count: FeatureState;
7. }
8. export const selectFeature = (state: AppState) => { return state.count };
9. export const selectFeatureCount = createSelector(
10. selectFeature,
11. (state: FeatureState) => state.counter
12. );

5. Create a component **"counter"** and add the following code to **counter.component.ts**:

1. import { Component, OnInit } from '@angular/core';
2. import { Store, select } from '@ngrx/store';
3. import { Observable } from 'rxjs';
4. import { increment, decrement, reset, multiply } from '../Store/actions/counter.action';
5. import \* as fromRoot from '../Store/selectors/counter.selector';
6. @Component({
7. selector: 'app-counter',
8. templateUrl: './counter.component.html',
9. styleUrls: ['./counter.component.css']
10. })
11. export class CounterComponent implements OnInit {
12. count$: Observable<number>;
13. constructor(private store: Store<fromRoot.AppState>) {
14. }
15. ngOnInit(): void {
16. this.count$ = this.store.pipe(select(fromRoot.selectFeatureCount))
17. }
18. increment() {
19. this.store.dispatch(increment());
20. }
21. decrement() {
22. this.store.dispatch(decrement());
23. }
24. reset() {
25. this.store.dispatch(reset());
26. }
27. multiplyByTwo() {
28. this.store.dispatch(multiply({ multiplyBy: 2 }))
29. }
30. multiplyByThree() {
31. this.store.dispatch(multiply({ multiplyBy: 3 }))
32. }
33. }

6. Add the following code to **counter.component.html:**

1. <div class="card shadow row col-md-6 offset-md-3 mt-5 text-center">
2. <div class="card-body">
3. <h3 style="font-family: cursive;">Current Count: {{ count$ | async }}</h3>
4. <button class="btn btn-outline-danger mr-2" id="increment" (click)="increment()">Increment</button>
5. <button class="btn btn-outline-success mr-2" id="decrement" (click)="decrement()">Decrement</button>
6. <button class="btn btn-outline-dark mr-2" id="reset" (click)="reset()">Reset Counter</button>
7. <button class="btn btn-outline-warning mr-2" id="multiply" (click)="multiplyByTwo()">Multiply By 2</button>
8. <button class="btn btn-outline-info" id="multiply" (click)="multiplyByThree()">Multiply By 3</button>
9. </div>
10. </div>

7. **app.module.ts**:

1. import { BrowserModule } from '@angular/platform-browser';
2. import { NgModule } from '@angular/core';
3. import { AppRoutingModule } from './app-routing.module';
4. import { AppComponent } from './app.component';
5. import { CounterComponent } from './counter/counter.component';
6. import { StoreModule } from '@ngrx/store';
7. import { counterReducer } from './Store/reducers/counter.reducer';
8. @NgModule({
9. declarations: [
10. AppComponent,
11. CounterComponent
12. ],
13. imports: [
14. BrowserModule,
15. AppRoutingModule,
16. StoreModule.forRoot({ count: counterReducer })
17. ],
18. providers: [],
19. bootstrap: [AppComponent]
20. })
21. export class AppModule { }