21/12/22

דוגמאות לשימוש בICL

לדעתי זה מכסה את החלק של העבודה בתקשורת מול השרת. נתתי פה שני אופציות לעבודה מול השרת, עדיף לבחור אחת מהן.

יש צורך להבין מה זה NGRX ואיך עובדים אתו (בצורה בסיסית) בעיקר הנושא של Selector-ים.

קובץ app.module.ts

@NgModule({

declarations: [

AppComponent,

MainComponent,

SplashComponent

],

imports: [

CommonModule,

BrowserModule,

AppRoutingModule,

StoreModule.forRoot({}, {

runtimeChecks: {

strictStateImmutability: true,

strictActionImmutability: true

}

}),

EffectsModule.forRoot([]),

StoreDevtoolsModule.instrument({

maxAge: 50

}),

ClientCacheModule.forRoot(new ClientCacheConfig({ serverUrl: environment.apiUrl })),

DistributionPollingModule.forRoot(new DistributionPollingConfig({ mode: 'Batch', serverUrl: environment.apiUrl })),

DistributionSignalrModule.forRoot(new DistributionSignalrConfig({ serverUrl: environment.apiUrl })),

MaterialModule,

IclCoreModule,

],

providers: [],

bootstrap: [AppComponent],

})

export class AppModule { }

פעולות CRUD:

פעולות CRUD באחריות שירות ClientCacheService. פעולות:

1. createEntity(entity: IclEntityModel, onSuccess?: () => void, onError?: (exception: any) => void)
2. updateEntity<TEntity extends IclEntityModel>(entity: TEntity, onSuccess?: () => void, onError?: (exception: any) => void)
3. deleteEntity(entity: IclEntityModel, onSuccess?: () => void, onError?: (exception: any) => void)
4. getEntity(entityType: string, id: string, onSuccess?: () => void, onError?: (exception: any) => void)
5. getAllEntities(entityType: string, options?: DistributionOptions, onSuccess?: () => void, onError?: (exception: any) => void)
6. clearAllEntities(entityType: string, options?: DistributionOptions)

פעולת getAllEntities אוטומטית נרשמת להפצה

פעולת clearAllEntities אוטומטית מתנתקת מהפצה

this.ccService.getAllEntities('EntityA', { type: 'PollingDistributionType', override: false });

this.ccService.getAllEntities('EntityB', { type: 'PollingDistributionType', override: false, filter: {

ResultSettings: {

PageSize: 100,

PageCount: 0,

}

} });

this.ccService.getAllEntities('EntityC'', { type: 'SignalrDistributionType' });

לשליפה של מידע מ Store (NGRX) עובדים עם מחלקת עזר ClientCacheSelector

ישנם שלושה Selector-ים מוכנים מראש.

entitySelector(entityType: string, id: string)

entitiesSelector(entityType: string)

entityChangesSelector(entityType: string)

דוגמא לשימוש:

this.data$ = this.ccService.select(this.ccSelector.entitiesSelector('EntityA')) as Observable<EntityA[]>;

this.data$.subscribe(x => { //Your Code… });