NullInjectorError: No provider for ReducerManager

Asked 5 years, 9 months ago Modified 3 years, 8 months ago Viewed 36k times



I am using the new ngrx 5. This is the file that holds the reducers and the featureSelector:

35





```
import AppState from '../interfaces/app.state'
import { ActionReducerMap, createFeatureSelector } from '@ngrx/store'
import { partnerReducer } from './partner.reducer'
export const reducers: ActionReducerMap<AppState> = {
  partnerState: partnerReducer
export const getAppState = createFeatureSelector<AppState>('appState')
```

This is how I am importing the storeModule

```
@NgModule({
declarations: [...],
imports: [...
  RouterModule.forRoot(ROUTES),
  StoreModule.forFeature('appState', reducers)
],
providers: [...],
bootstrap: [AppComponent],
entryComponents: [...]
})
export class AppModule { }
```

I have followed this tutorial

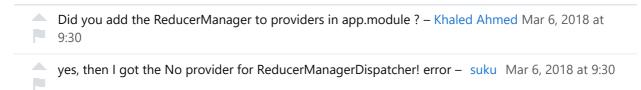
When I run the app, I get the following error:

```
"StaticInjectorError(AppModule)[StoreFeatureModule -> ReducerManager]:
\n StaticInjectorError(Platform: core)[StoreFeatureModule -> ReducerManager]:
     NullInjectorError: No provider for ReducerManager!"
```

But if I do provide ReducerManager in the providers, I get this error:

```
No provider for ReducerManagerDispatcher!
angular
          ngrx-store
                     ngrx-store-4.0
Share Edit Follow Flag
```

asked Mar 6, 2018 at 9:23 suku **10.5k** 16 76 121



2 Answers

Sorted by: Highest score (default) \$



Managed to solve this by adding StoreModule.forRoot({}), in the imports.

73

StoreModule.forRoot should only be called once in the root of your project NgModule. If you wan't to register a feature, use StoreModule.forFeature. Using forRoot registers the global providers needed for Store.



Check the github discussion <u>here</u> on this issue. The above reason was stated in the same discussion

1

Share Edit Follow Flag

answered Mar 6, 2018 at 9:38



I had the same problem and I found this solution





Share Edit Follow Flag

answered Mar 13, 2020 at 14:35



1 his is same as the answer below – suku Mar 14, 2020 at 10:39