tmo-coding-challenge

**Task 1:**

**What is done well?**

1. Good use of Lazy loading in application. ‘feature-shell’ module is lazy loaded from ‘app’ module.
2. Saving data from API response in store created using ngRx.
3. ‘NoopAnimationsModule’ is used here in application instead of using ‘BrowserAnimationsModule’ which is suitable for this application as animation is not involved here.
4. Use of ‘StoreDevtoolsModule’ with  [*Redux Devtools Extension*](https://github.com/zalmoxisus/redux-devtools-extension/) helps me a lot while working on store to view enteries and ids of price query
5. Pipeable operators are used when using RxJs operators.

Ex. File name : ‘price-query.facade.ts’ , Line no: 6

import { map, skip } from 'rxjs/operators';

**What would you change?**

1. Use of ‘ngSubmit’ angular form submit event would be better than direct calling of a function on button click. **[FIXED]**
2. We should show a spinner till chart is loaded after submitting the form.
3. Once the development is done, we can remove ‘StoreDevtoolsModule’ as it is basically use to monitor ngRx Store.
4. Return type should be added to function. If a method does not return, then its return type could be set to void. **[FIXED]**

Ex.

fetchQuote(): void {

if (this.stockPickerForm.valid) {

1. Access modifier should be added for variables and methods.

**Are there any code smells or problematic implementations?**

1. Application is not running properly. On click of ‘Go’ button, it is not showing the chart **. [FIXED]**
2. Return type of methods are not defined.

Ex.

fetchQuote() {

if (this.stockPickerForm.valid) {

1. Appropriate Access Modifier are not used for variables and functions inside components.

It will increase the readability of code.

1. There are unused variables inside components.

Ex. File name: stocks.component.ts, Lines : 12 & 13  **[FIXED]**

1. Lint rules and naming convention are not proper. We should always declare variables or constants with a type other than any

Ex.

data: any;

columnNames: string[];

options: any;