Type Script question:

You are developing an application for the library book management using Angular.

On the preview page you can see the following elements:

The form for adding new books with one input field for the book title.

List of added books with "Delete" buttons.

Acceptance Criteria

Scenario: Submitting filled form

Given The form is not emtpy

When The "Add Book" button is clicked

Then The book is added to the end of the list

And The form input field is cleared

Scenario: Submitting empty form

Given The form is empty

When The "Add Book" button is clicked

Then Form input field stays as is

And The book is not added to the list

Scenario: Deleting one of the books

Given Books list is not empty

When "Delete" button is clicked for one of the books

Then This book is removed from the list

Notes

You can expand the test results to see a more detailed report.

Please use the provided template, as the tests access the elements by using the CSS selectors. If you change the structure of the DOM elements, the tests will fail.

[execution time limit] 10 seconds

Provide the below content, need to **write the code where it is placed // TODO: Write your code here**

const { Component, VERSION } = ng.core;

const { FormGroup, FormControl } = ng.forms;

@Component({

selector: '#app',

template: `

<div class="content-container">

<h2>Add new book</h2>

<form class="add-book-form" [formGroup]="bookForm">

<input

class="add-book-form\_\_title-input"

placeholder="Book title"

type="text"

/>

<input type="submit" class="add-book-form\_\_btn" value="Add Book" />

</form>

<h2>Your books</h2>

<div class="book-list" \*ngFor="let book of books">

<div class="book-list\_\_item">

<span class="book-list\_\_title">{{book.name}}</span>

<button

type="button"

class="book-list\_\_delete-btn"

>Delete</button>

</div>

</div>

</div>

`

})

class BookManagerComponent {

books = [{id: 1, name: 'First book'}];

bookForm = new FormGroup({});

addBook() {

// TODO: Write your code here

}

deleteBook(bookId) {

// TODO: Write your code here

}

}

// main.js

const { BrowserModule } = ng.platformBrowser;

const { NgModule } = ng.core;

const { CommonModule } = ng.common;

const { ReactiveFormsModule } = ng.forms;

@NgModule({

imports: [

BrowserModule,

CommonModule,

ReactiveFormsModule

],

declarations: [BookManagerComponent],

bootstrap: [BookManagerComponent],

providers: []

})

class BookManagerModule {}

const { platformBrowserDynamic } = ng.platformBrowserDynamic;

platformBrowserDynamic()

.bootstrapModule(BookManagerModule)

.catch(err => console.error(err));