# Lab: Code Coverage

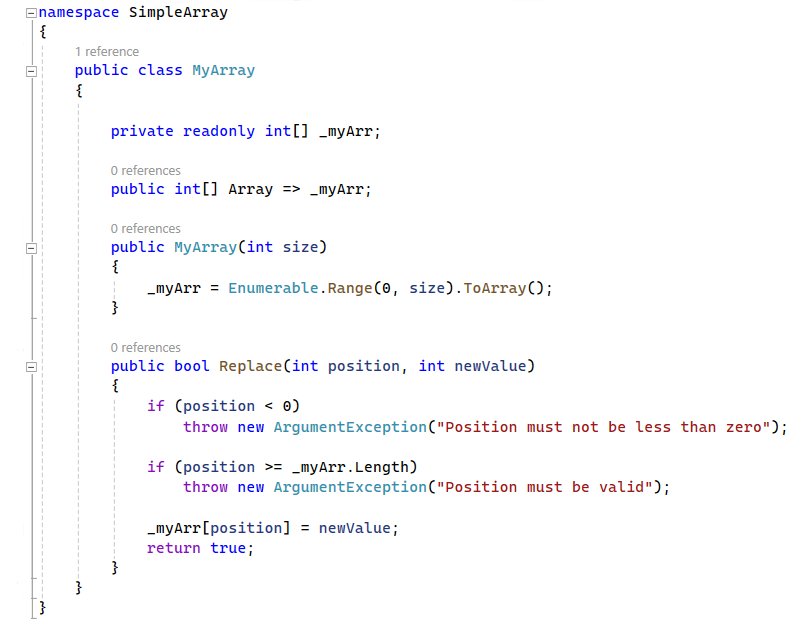
Lab problems for the ["Back-End Technologies Basics"](https://softuni.bg/trainings/4398/back-end-technologies-basics-january-2024) Course @ SoftUni.

# Code Coverage in C#

## MyArrayProject

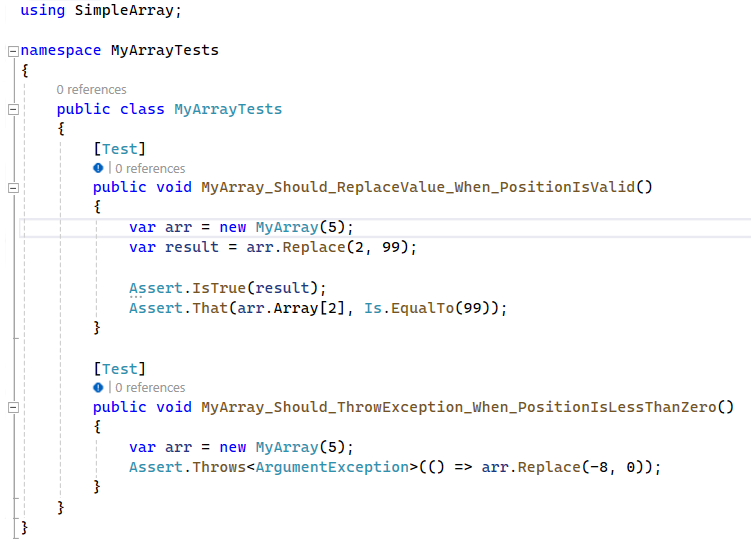
### The project

Let's create a class library with a single class with only one method:



### Tests

Then, create a test project with a test class and write some tests for the MyArray class:



### Install Report Generator

Open the PowerShell with admin privileges and run the following commands:

**dotnet tool install -g dotnet-reportgenerator-globaltool**

**dotnet tool install dotnet-reportgenerator-globaltool --tool-path tools**

**dotnet new tool-manifest**

**dotnet tool install dotnet-reportgenerator-globaltool**

### Install Run Coverlet Report

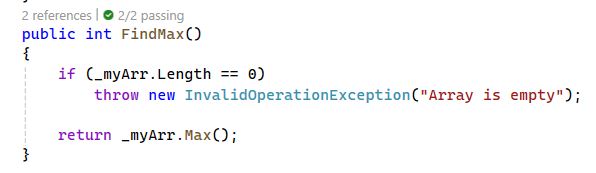
From the Extensions menu, search for and install Run Coverlet Report. Restart VS.  
Run Tests with Coverage

Observe the results. Can you write some other test to reach 100% coverage?  
Try Code Coverage with Fine Code Coverage

Keep in mind that Fine Code Coverage is not perfect, so we suggest that you disable the extension when not using it. It is perfectly safe, but sometimes it makes the tests to run twice or other funny stuff.

### **Implement a new feature in the MyArray class**

Implement a new feature in the MyArray class, a method to find the maximum value in the array. Check Code coverage again. Then, write tests for this new feature, aiming for 100% coverage of the new code as well.



## Simple Library Management System

Library Management System is a backend simulation that will allow us to manage a collection of books and provide basic functionalities such as adding books to the system, checking out books, and returning them.

#### The Book Class

The Book class represents a book in our library.

#### The Library Class

This class maintains a private list of Book objects, representing all the books that are part of the library's collection. It provides the following methods:

**AddBook(Book book**)**:** Adds a new Book object to the library.

**CheckOutBook(int bookId):** Marks the book with the specified Id as checked out, if it is not already checked out. It returns true if the checkout was successful, false otherwise.

**ReturnBook(int bookId):** Marks the book with the specified Id as returned, making it available for others to check out. It returns true if the return was successful, false otherwise.

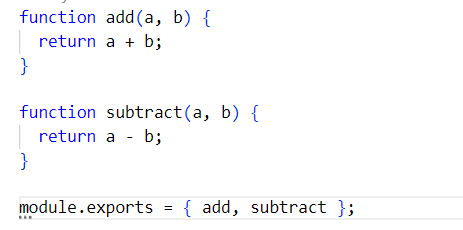
### Writing Tests for 100% Code Coverage

Your task is to write tests that achieve 100% line and branch coverage.

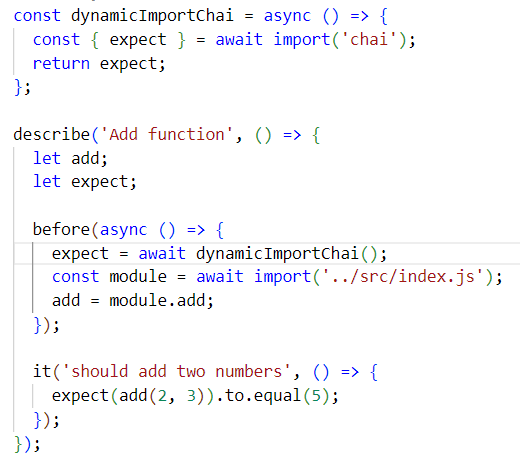
# Code Coverage in JS

## Mocha-Chai-Demo

### Simple Project

You are given a simple project with add and subtract functions:  


### Tests

The test for the Add function is already written for you:  


### Add and configure NYC in your project

Add Mocha and Chai if you’re not done so. Following the steps given in the lecture’s presentation, install and configure NYC.

### Run the tests with coverage

Analize the results. View lines that are not covered.

### Refactor the tests to achieve 100%

Include subtract function in the tests.