



Technical Exercises Documentation

Author: Ambreen Hussain

Version: 1.0

Contents

| | |
|--|---|
| 1. Introduction | 3 |
| 2. Pre-requisites | 3 |
| 3. Technologies used..... | 3 |
| 4. Project Folder and Files Structure..... | 3 |
| 5. Opening the project (“Pearson Technical Exercises”)..... | 3 |
| 6. Launch the project with the Chrome Browser..... | 3 |
| 7. Troubleshooting problems..... | 4 |
| 7.1 Issue of running the project with Chrome Browser..... | 4 |
| 7.2 Issue of the project failing to launch despite installed Chrome extension..... | 4 |
| 8. Executing the test cases..... | 5 |

1. Introduction

This document provides an overview of how to launch the web application for viewing the Pearson technical exercises output. All the technical exercises were completed using JavaScript. The web application includes a front-end display for the test cases. The test cases were coded using Mocha and Chai.

2. Pre-requisites

- Visual Studio Code
- Chrome Browser

3. Technologies used

- HTML
- CSS
- JavaScript
- Mocha/Chai

4. Project Folder and Files Structure

Project folder name: *“Pearson Technical Exercises”* - The primary project folder holds the files shown below.

Project file names:

- index.html
- main.js
- site.css
- test.js
- loading.gif
- launch.json

5. Opening the project (“Pearson Technical Exercises”)

Save the project on a drive or the desktop. Launch Visual Studio Code, select open folder and navigate to the location where the project was saved.

6. Launch the project with the Chrome Browser

After the project “Pearson Technical Exercises” is opened using Visual Studio Code, press the F5 key to launch the project. If prompted to select a browser, choose the Chrome Browser. The project should launch the web application with Chrome.

7. Troubleshooting problems

7.1 Issue of running the project with Chrome Browser

If running the F5 command fails to show an option for launching the application with Chrome, in this case, the required *“Debugger for Chrome Nightly”* extension is not installed. The debugger for Chrome is useful for debugging JavaScript code in the Browser. Install the extension (Debugger for Chrome Nightly) with Visual Studio Code by using the command CTRL + SHIFT + X which launches Extensions. Search for the “Debugger for Chrome Nightly” in the Extensions search bar and install the extension.

7.2 Issue of the project failing to launch despite installed Chrome extension

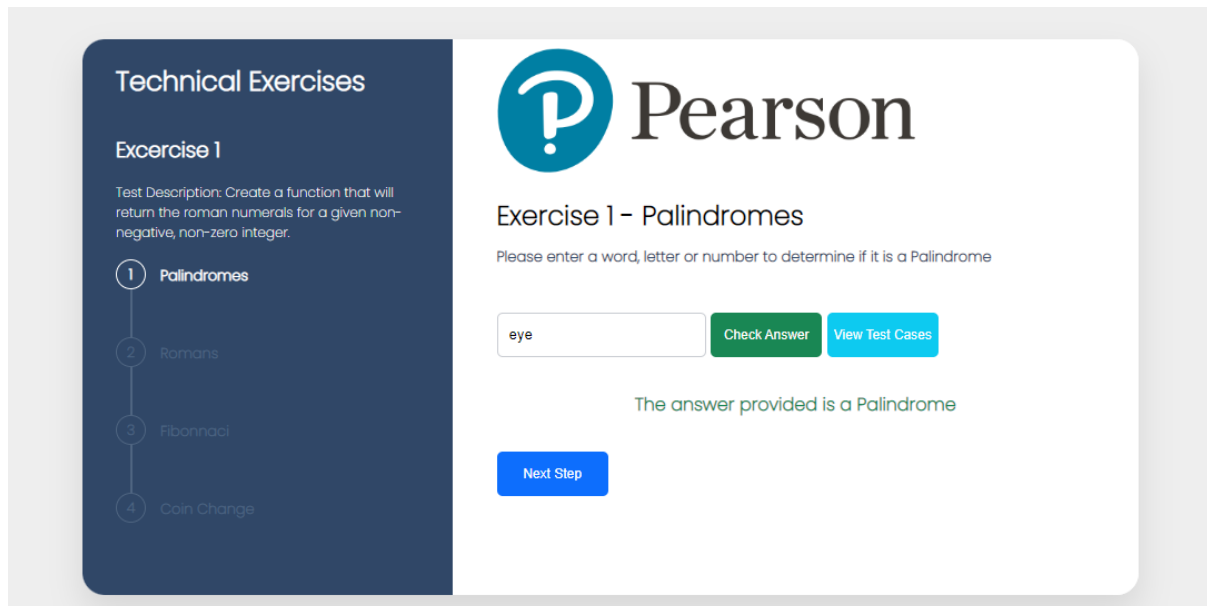
In the event the project fails to launch despite having the Chrome Browser extension installed, check the settings for the **launch.json** file are set as the following:

```
{
  "version": "0.2.0",
  "configurations": [
    {
      "name": "Launch index.html",
      "type": "chrome",
      "request": "launch",
      "file": "${workspaceFolder}/index.html"
    }
  ]
}
```

```
{
  // Use IntelliSense to learn about possible attributes.
  // Hover to view descriptions of existing attributes.
  // For more information, visit: https://go.microsoft.com/fwlink/?linkid=83
0387
  "version": "0.2.0",
  "configurations": [
    {
      "name": "Launch index.html",
      "type": "chrome",
      "request": "launch",
      "file": "${workspaceFolder}/index.html"
    }
  ]
}
```

8. Executing the test cases

On completion of an exercise, the web application will display all test cases for the exercise. For example, exercise 1 – Enter the word ‘eye’ in the input field and click the ‘*Check Answer*’ button to finish the palindrome status check. When the check to confirm the palindrome status is completed, a button named ‘*View Test Cases*’ is displayed. See example below:



Clicking on the ‘*View Test Cases*’ button will show the results for test cases:

