**User Manual**

**Code College**

****

# Table of Contents

Contents

[Table of Contents 2](#_Toc449559989)

[Introduction 3](#_Toc449559990)

[Prerequisites 3](#_Toc449559991)

[Installation 3](#_Toc449559992)

[Login 5](#_Toc449559993)

[Sign Up 6](#_Toc449559994)

[Dashboard 7](#_Toc449559995)

[Exercise 7](#_Toc449559996)

[Add Exercise 8](#_Toc449559997)

[Exercise File 9](#_Toc449559998)

# Introduction

Code College is a website for learning Lua. It is developed using ASP.NET MVC 5. It is hosted in IIS 7 on a Windows Server machine. A user can login and pick an exercise from the dashboard, there are exercises on basic aspects of any procedural programming language that are taught using Lua.

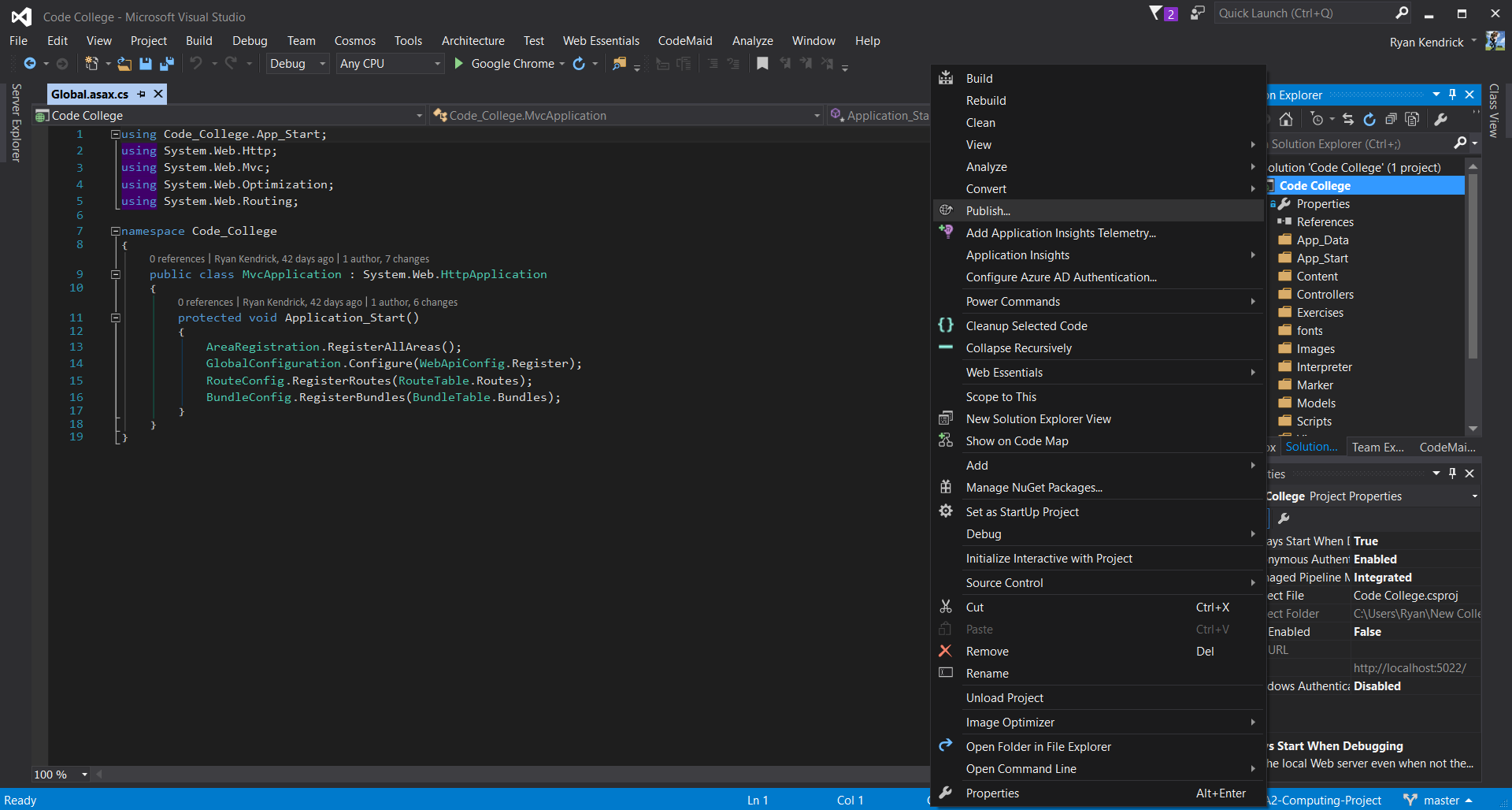
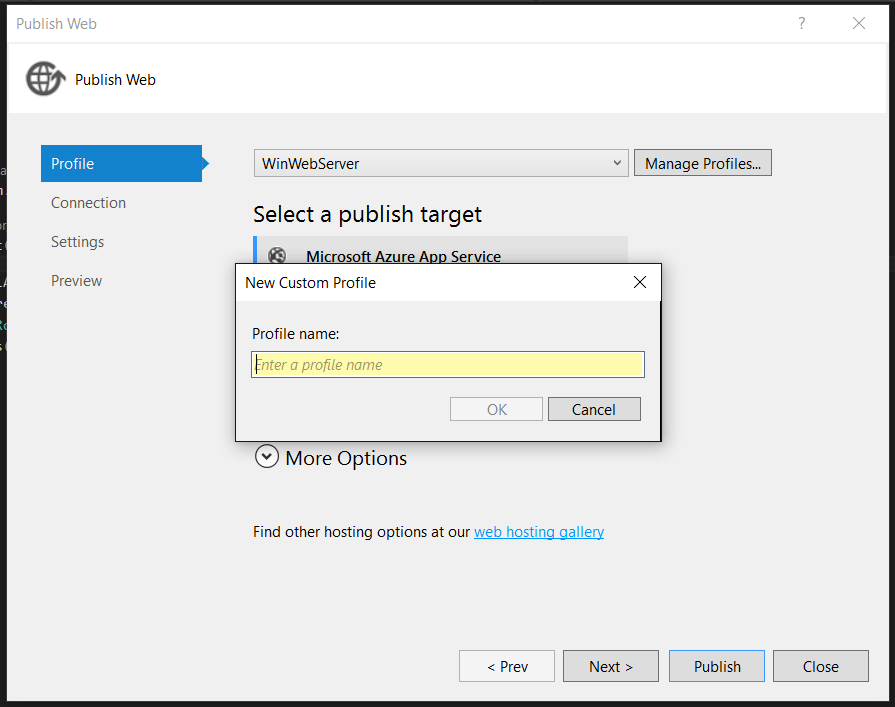
# Prerequisites

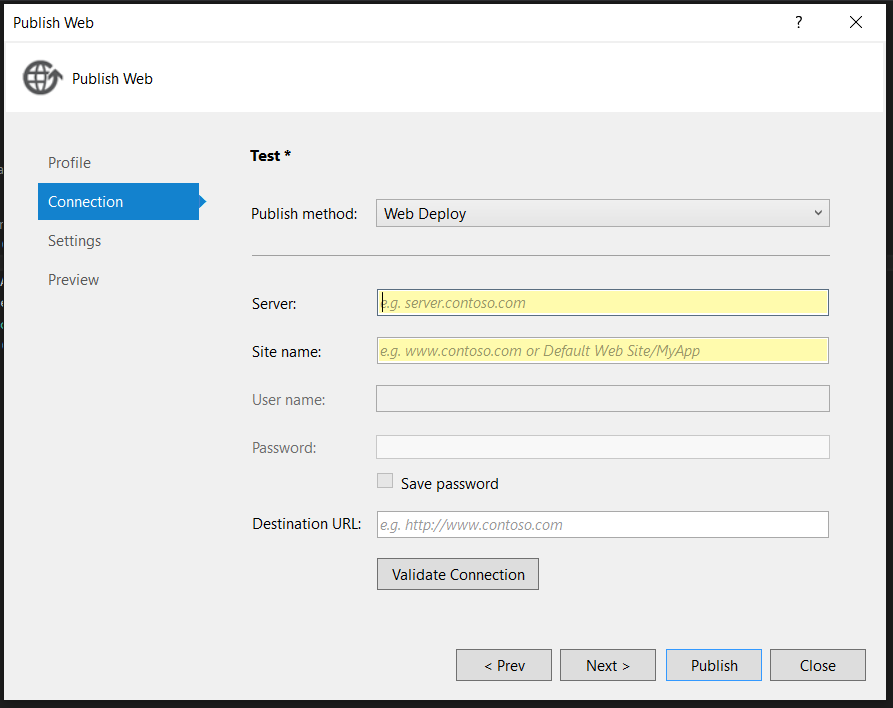
The prerequisites for hosting the site on a server are:

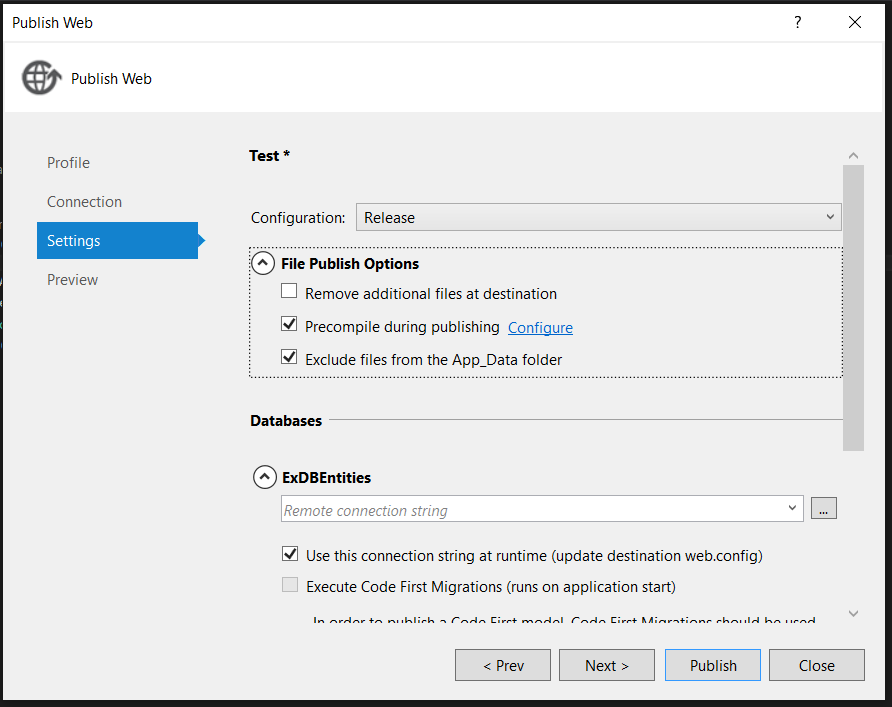
* A Windows Server 2008 R2 or higher, preferably at least 4GB of RAM and 2GHz quad-core CPU
* .NET Framework 4.6 or higher installed
* IIS 7 or higher installed
* SQL Server 2014 or higher installed
* Internet connection
* Public facing, static IP address, preferably with a DNS name pointing to the public IP address

# Installation

To deploy the web application on the server, you must set up Web Deploy in Visual Studio. You can get the solution from the GitHub repository.

1. You must open up the Visual Studio solution in Visual Studio 2012 or later.
2. Secondly, right click on the project in the solution explorer and click on the ‘Publish…’ option.
3. A window will open and you should create a new publish profile from the drop down box. A new window will open and you should put the profile name in it.
4. Click next and enter in details specific to your server.



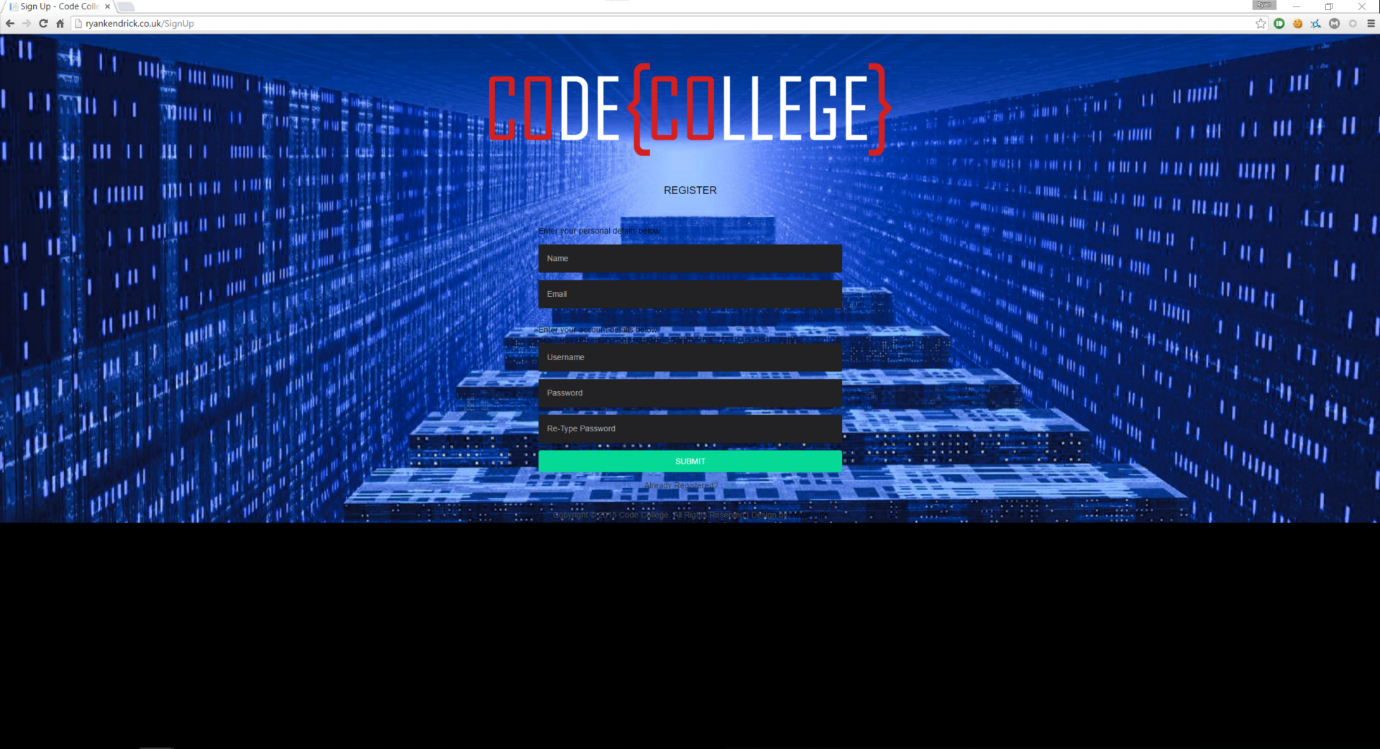
1. Tick the following checkboxes and set up the SQL Server connection string for your server.

Now when you click publish, the project will be compiled and deployed on the server automatically.

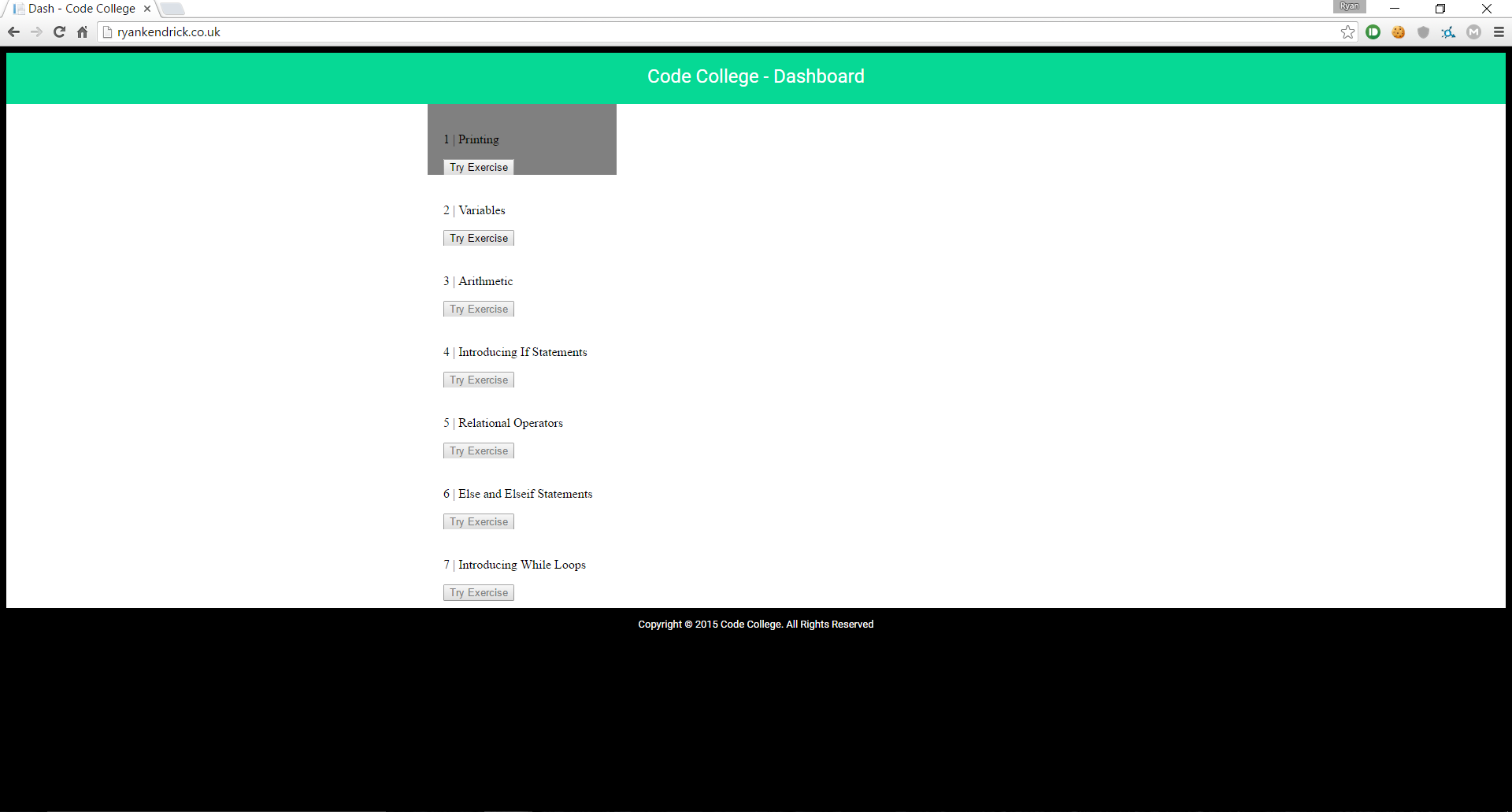
# Login

On the login screen, you must put your username and password in the assigned textboxes.

# Sign Up

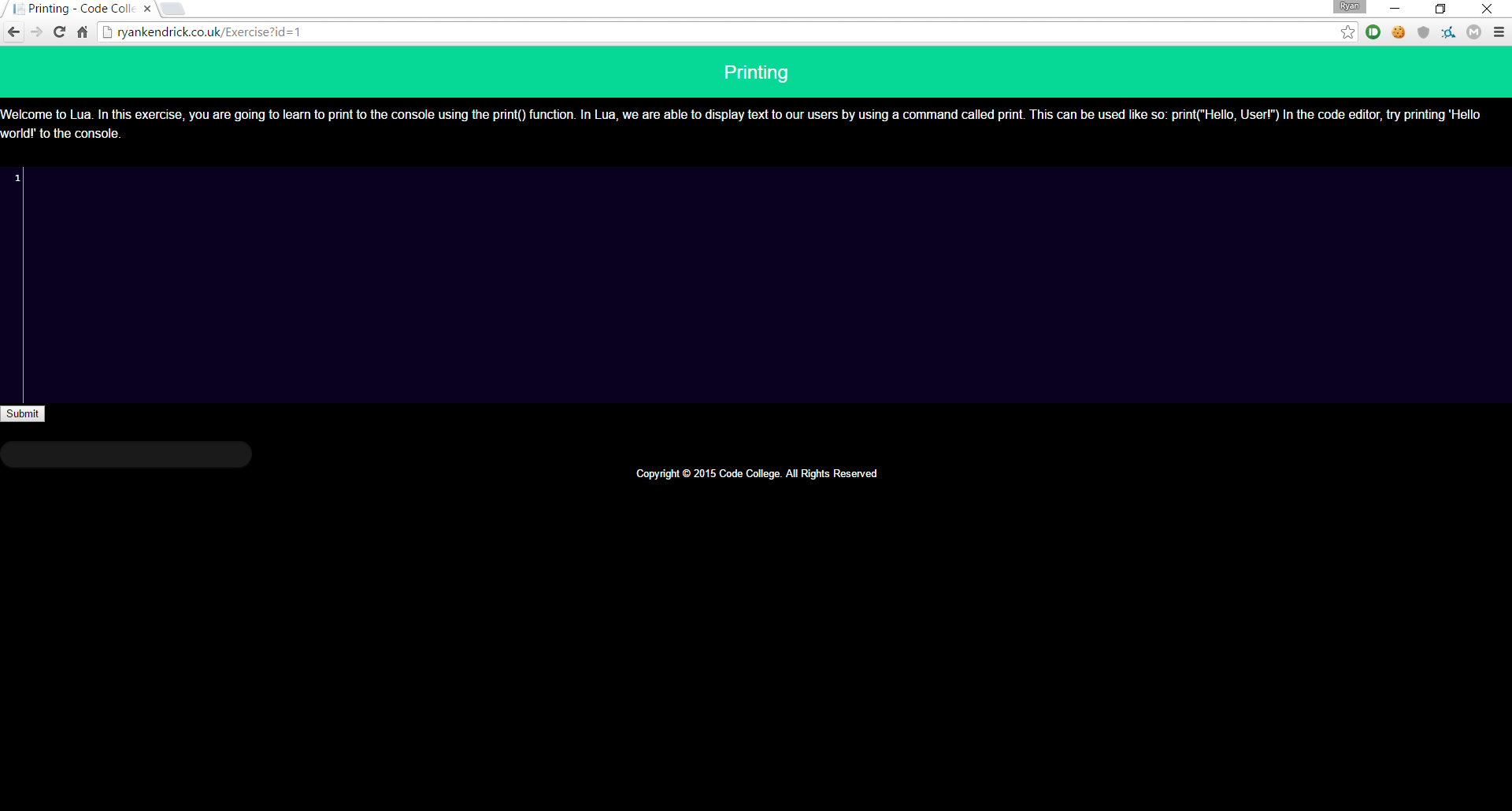
In the sign up page, you should put your data in the labeled textboxes.

# Dashboard

On the dashboard, you can click on the button of the exercise you want to complete. If you have not unlocked the exercise, the button is disabled and you cannot complete it. If you have already done an exercise, the exercise tile has a grey background.

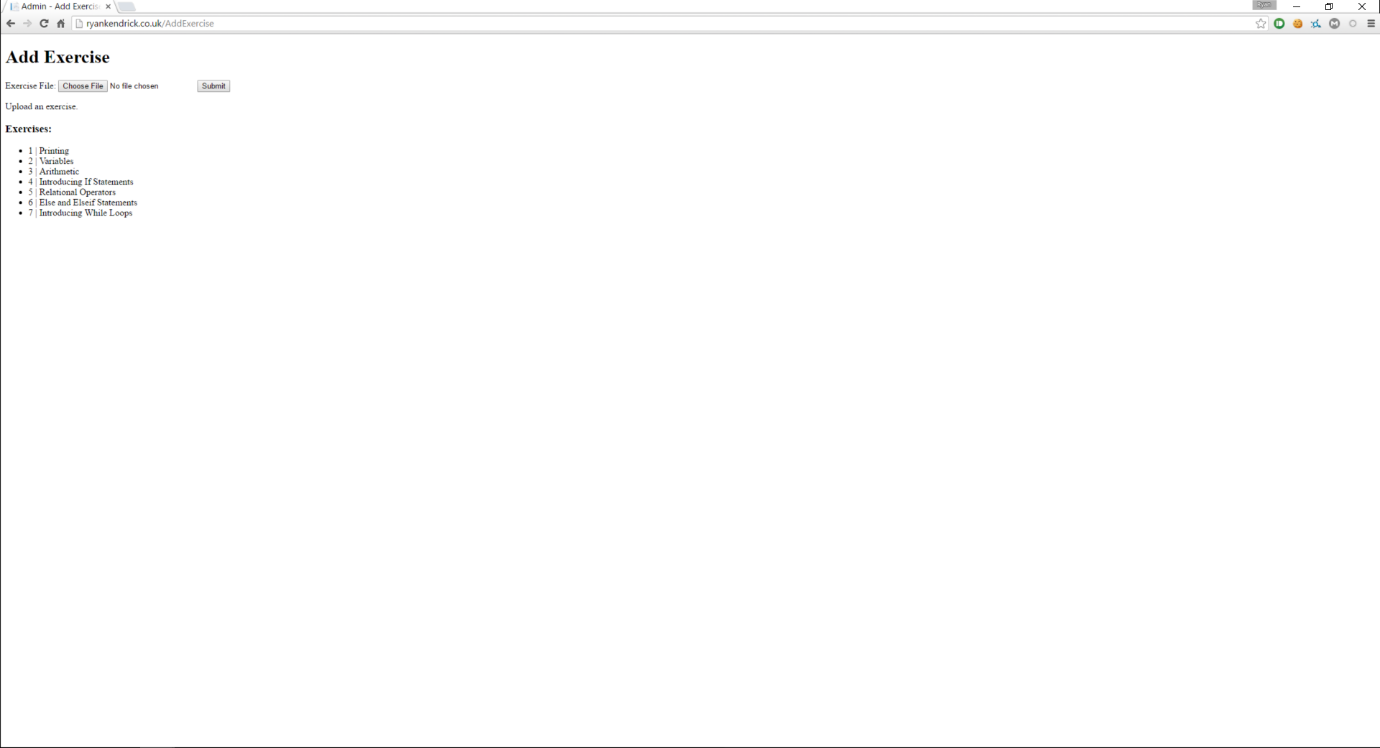
# Exercise

On the exercise page, you should carefully read the exercise description and complete the task. Write the code in the editor and submit it by hitting the submit button. If the code is correct, the program output is written to the console and the console goes green. If it is incorrect, the console goes red and the interpreter’s error is printed in the console. After the exercise is completed, the submit button turns into a next exercise button. You should click it to go to the next exercise.



# Add Exercise

The AddExercise page is for admins only.

An admin can upload an exercise file using the file upload control. You can see the exercises in the database are they are listed on the page. When an exercise file is uploaded, the page reloads and the exercise is added to the database.

# Exercise File

To create an exercise file you should create a file with the suffix ‘.ex’. It should be in the following form:

[ExID]

// Exercise ID

[ExTitle]

// Exercise Title

[ExDescription]

// Exercise Description

[ExCodeTemplate]

// Exercise Code Template (what is in the editor when the user loads the exercise)

[ExAppendCode]

// Exercise Code Appendage (what is added to the code after the user hits submit, the user does not see this)

[ExMarkScheme]

<?xml version="1.0" encoding="utf-8"?>

<MarkScheme>

<Output></Output>

<Variables>

</Variables>

<ControlStructures>

</ControlStructures>

</MarkScheme>