|  |  |
| --- | --- |
| MIT%20Award%20smal(bw) | **School of Digital Technologies: BDT – Bachelor of Digital Technologies**  **502.632 - Full-Stack Web Development**  **PRACTICAL TEST 01**  **Assessment Cover Sheet** |
| Course | 502.632 - Full-Stack Web Development |
| Assessment name | PRACTICAL TEST 01 |
| Assessment type | Practical Test – "Open Book" – reference allowed to The Internet, notes, examples, Canvas Resources. Information exchange between students and/or other human agents by any means is not allowed during the test. |
| Date | Tue, 12 Mar 2019 |
| Course contribution | **20%** (Test has 20 marks - each test mark is a course mark) |
| Time Allowed | 135 minutes |
| **Student ID:** |  |
| **Student NAME:** |  |
| **Statement of Valid Authorship**  I hereby confirm that these test answers are my own work done in the test time. This means that to the best of my knowledge and belief, this test contains only my own created and assembled responses to these particular questions without communication with any other person except the supervising lecturer. I make this statement fully understanding that, should it be found false, I will, in most circumstances, receive zero marks for this test.  Signed by student: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_ | |
| **Relates to all 4 Learning Outcomes, but especially to 2 and 4:**   1. Demonstrate an understanding of the architecture of web applications including the roles of clients and servers and the use of protocols for interaction. 2. **Learn and apply a server-side programming platform and/or language.** 3. Implement dynamic web user interfaces using HTML, CSS, JavaScript, and AJAX. 4. **Implement and publish web-based software solutions that interact with a variety of data sources and public APIs.** | |

**PRACTICAL TEST 01 - "Tennis Club"**

**Question ONE - Web App Setup [1 mark]**From Canvas, Download file "Test\_Tennis.zip".  
Move this into "C:\Users\<Your Username>"  
Or Mac or personal laptop defined working area.

Unzip -- 7-Zip recommended -- using "Extract Here".  
You should now have folder " Test\_Tennis".  
as in "C:\Users\<Your Username>\ Test\_Tennis" for lab machines.  
Rename this adding your first name. e.g. John would rename his folder "Test\_Tennis\_John".

Use Visual Studio to create a new ASP.NET Core 2.2 (or 2.1) web app under " Test\_Tennis\_<FirstName>".  
Do NOT check (tick) the checkbox "Create directory for solution"   
Name the app "MVCManukauTech"  
Select option "Web Application (Model-View-Controller)"  
Click button "Change Authentication" and change authentication to "Individual User Accounts".  
Create this app.

**Question TWO - Database Setup** **[1 mark]**  
Use your database account to create a new empty database to work with.  
Name it in a similar way to your starter database using "Tennis" as the 3rd part of the database name.  
eg John has a starter database "F191\_John\_Starter"  
He creates this new one as "F191\_John\_Tennis"

Then load the supplied script "Tennis\_JPC\_v02.sql" into SQL Management Studio (or Visual Studio).   
Edit the name of your new database into the script .

Run the script to populate the database. Check that tables have appeared.  
Show the result to your marking lecturer who will assess this "on the spot".  
It is essential for you to complete this question to be able to move on to most of the other questions.  
For that reason the lecturer may help you complete Questions ONE and TWO but in such a case you will probably get reduced marks for these questions depending on the reasons for you getting stuck.

**Question THREE [1 mark]**Edit app file "appsettings.json" to include the connection string for the database you have created.  
Test this by registering a test user. This should generate the "AspNetUsers" and related Identity tables in the Database. Check for this result.

**Question FOUR** **[3 marks]**  
Add 4 new membership fields (columns) for extra member information   
in table "AspNetUsers". These new fields are:  
"IsCompeting", "NumberOfPenalties", "CommitteePosition", "BiggestPrizeAmount".  
You select and configure appropriate data types for them.

**Question FIVE** **[4 marks]**  
Make the necessary code and UI changes so that the Registration Form now includes   
these 4 extra fields and the code saves their user data entry to table "AspNetUsers".

**Question SIX [2 marks]**  
Create a "Scaffold-DbContext" command and run it in the "Package Manager Console"  
to cause your app systems to scan the database tables and generate the classes for data access.  
Do the necessary follow-ups to get data access working.

**Question SEVEN [1 marks]**Get database table "Player" to display as an on-screen web user interface   
with Create, Update and Delete capabilities.

**Question EIGHT [3 marks]**  
In the "Details" view generated in Question SEVEN:

* Simplify the display and the code by displaying only the fields   
  from "PlayerId" to "YearJoined".
* Replace the HTML Helpers and Tag Helpers with code that is as close to standard HTML as possible.

**Question NINE** **[4 marks]**Here is a URL with QueryString giving input of tennis results in JSON format.  
You will probably need to change the port number, highlighted below as 7154, to your port number.

http://localhost:7154/Calc/Result?j=[{"ResultId":1,"TeamId":1,"PlayerId":6,"Wins":3},{"ResultId":2,"TeamId":2,"PlayerId":3,"Wins":2},{"ResultId":3,"TeamId":2,"PlayerId":5,"Wins":3},{"ResultId":4,"TeamId":1,"PlayerId":9,"Wins":0},{"ResultId":5,"TeamId":2,"PlayerId":2,"Wins":2},{"ResultId":6,"TeamId":2,"PlayerId":3,"Wins":3},{"ResultId":7,"TeamId":1,"PlayerId":4,"Wins":1},{"ResultId":8,"TeamId":3,"PlayerId":7,"Wins":3},{"ResultId":9,"TeamId":3,"PlayerId":8,"Wins":2},{"ResultId":10,"TeamId":2,"PlayerId":4,"Wins":3}]

Create the Remote Method Web Service to calculate and display:  
the Mean number of Wins for Team 2 (the team with a TeamId of 2).

**Total [20 marks]**