|  |  |
| --- | --- |
| MIT%20Award%20smal(bw) | **Faculty of Business and Information Technology: BDT – Bachelor of Digital Technologies**  **502.632 - Full-Stack Web Development**  **PRACTICE for TEST 02**  **Assessment Cover Sheet** |
| Course | 502.632 - Full-Stack Web Development |
| Assessment name | PRACTICE for TEST02 |
| Assessment type | Practical Test – "Open Book" – reference allowed to The Internet, notes, examples, Canvas Resources. Information exchange by any means between students and other humans, or AIs of human cognitive level, is not allowed during the test. |
| Date | PRACTICE TEST – Real Test will be Thu, 06 Sep 2018 |
| Course contribution | **30% -** (Test has 30 marks - a 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 course 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**   1. Demonstrate an understanding of the architecture of web applications including the roles of clients and servers and the use of protocols for interaction. (Q 1,2,4) 2. Learn and apply a server-side programming platform and/or language. (Q 3, 5, 6) 3. Implement dynamic web user interfaces using HTML, CSS, JavaScript, and AJAX. (Q 7, 8) 4. Implement and publish web-based software solutions that interact with a variety of data sources and public APIs.(Q 4,6,7,8) | |

**PRACTICAL TEST02 - AdventureWorks Bicycle Shop -**Details and mark allocations are all subject to change for the "real test"

Download the file provided **"TestAW-183Q.zip"** from the Canvas page provided for this test.  
Download or move this to folder **"C:\Users\<your username>"** on your machine (or lab machine).  
Unzip there by right-click and "Extract Here".  
Rename the extracted folder to **"TestAW\_<your name>"**

At the END of the test, zip the subfolder **"TestAW\_<your name>"** and hand-in this zip to Canvas.

**Question ONE – App Setup and Remote Database testing** **[1 mark]**  
  
Use Visual Studio to create a new ASP.NET Core 2.1 (or 2.0) web app under folder "TestAW\_<your name>"  
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.

The remote database is provided. Change the connection string to this supplied:

Server=citizen.manukautech.info,6303;Database=AdventureWorks;UID=tron01;PWD=inVoice[22];Encrypt=true;TrustServerCertificate=true

Test the App Setup with this login:  
Username: admin@example.com

Password: inVoice[22]

**Question TWO [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.

Show your results for Question ONE and Question TWO 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]**  
Get database table "Customer" to display as an on screen web user interface

with Create, Update and Delete capabilities.

This table includes "PasswordHash", "PasswordSalt" and "RowGUID" fields.   
Modify code so that these fields and their data do NOT display on the Index page

**Question FOUR** **[7 marks]**  
4-1 Provide a Customers Search. [3 marks]  
 Do this by modifying the Customers Controller and View code   
 to provide a search across the fields FirstName, LastName and CompanyName.  
 The user enters a single Search String and your code looks for a match in any  
 of these 3 fields.

4-2 Make this an auto-suggest search [2 marks]

4-3 Make this an auto suggest search with user-friendly display of the suggestions   
 as clickable list items under the search input field. [2 marks]

**Question FIVE [2 marks]**  
Get database table "Product" to display as an on screen web user interface

with Create, Update and Delete capabilities.

This table stores binary image data in field "ThumbNailPhoto". This data  
looks bad in the "Index" view display.   
Modify Index.cshtml so it does NOT display this field.

**Question SIX [3 marks]**

Analyse the SQL supplied below for this question, and its data fields, to create a ViewModel.

Then use that ViewModel with the SQL to create an on-screen display

by adding a Method "ReportProd01" and its View to the "ProductsController" from Question FIVE.

SELECT p.ProductID, p.Name, pm.Name AS ProductModel, pmx.Culture, pd.Description

FROM dbo.Product p   
 INNER JOIN dbo.ProductModel pm ON p.ProductModelID = pm.ProductModelID   
 INNER JOIN dbo.ProductModelProductDescription pmx ON pm.ProductModelID = pmx.ProductModelID   
 INNER JOIN dbo.ProductDescription pd ON pmx.ProductDescriptionID = pd.ProductDescriptionID

**Question SEVEN [5 marks]**

Analyse the SQL supplied below for this question, and its data fields, to create a ViewModel.

Then use that ViewModel with the SQL to create an on-screen display

by adding a Method "ReportProd02" and its View to the "ProductsController" from Question FIVE.

SELECT ProductID, Name, Color, Size, Weight, StandardCost, ListPrice FROM Product

Improve this report by adding a new column that displays the Profit   
which is ListPrice - StandardCost

Improve this report by calculating the Total Profit and displaying it at the bottom of the new Profit column.

**QUESTION EIGHT** **[4 marks]**  
Demonstrate "AngularJS" by displaying the same report as Question SEVEN above   
using AngularJS  
For the Server-Side code, create a new Method "ReportProd2Ang" in ProductsController   
with a copy of your code from Question SEVEN as the starter code.

**QUESTION NINE [6 marks]**

SUPPLIED is a plain text file *“Bicycles.txt”*

The text is divided into a 5 blocks of text information.

Use the supplied images. Display a suitable Youtube Video.

9-1 Add as a new page to your working app.   
 Implement this as a new Index View from a new empty Controller.

9-2 Apply the Bootstrap resource already existing in MVC, to this page  
 Produce a result that has:

For large screens, 1st block as header plus 4 columns

|  |  |  |  |
| --- | --- | --- | --- |
| col-lg-12 | | | |
| col-lg-3 | col-lg-3 | col-lg-3 | col-lg-3 |

9-3 For medium screens. - 2 columns then 3 columns

|  |  |  |  |
| --- | --- | --- | --- |
| col-md-6 | | col-md-6 | |
| col-md-4 | col-md-4 | | col-md-4 |

9-4 For small screens – 1 column then 2 columns and 2 columns

|  |  |
| --- | --- |
| col-sm-12 | |
| col-sm-6 | col-sm-6 |
| col-sm-6 | col-sm-6 |

Customise Bootstrap by using the additional custom file "Site.css"   
to give different background colours to each block of text.

**END OF TEST**