# _LITfinalLOGO

# SUMMER EXAMINATIONS 2014

**Thursday, 15th May 2014, 14.30 p.m. – 17.30 p.m.**

**KSDEM\_8\_Y2**

**Course:** Bachelor of Science (Hons) in Software Development

**Year:** Two

**Subject:** Data Driven Applications

**Time Allowed:** 3 Hours

**Instructions: 1. ANSWER FOUR QUESTIONS AS FOLLOWS**

* ANSWER Question One (1) in PART A
* ANSWER Question Two OR Question Three in PART A
* ANSWER Question Four AND Five in PART B

**2.** Indicate clearly on the front of the answer book which

questions have been attempted.

**Additional Attachments or Exam Material to accompany this paper:**

### Candidates are permitted to access their instance of Microsoft SQL Server which is required to complete the practical questions.

**Internal Examiners: External Examiners:**

Mr. Gerry Guinane Mr Paul Powell

**PART A: Written answers required**

**Q1. DATABASE SYSTEMS TERMINOLOGY**

**a)** Describe the main features of the Relational Database (RDB) Model.

**(5 marks)**

**b)** Outline the logical database (DB) design process when using Functional Dependency (FD) design method.

**(10 marks)**

**c)** What are Enterprise Rules. Provide examples of Enterprise Rules that may exist in a car rental business. How are Enterprise rules used in the logical DB design process. **(5 marks)**

**(Total 20 Marks)**

**Q2. Logical Database (DB) Design – ER Method**

**a)** Identify and explain, using examples based on the table below, three anomalies arising due to bad logical DB design that can cause data integrity issues. **(12 marks)**



Contd/..

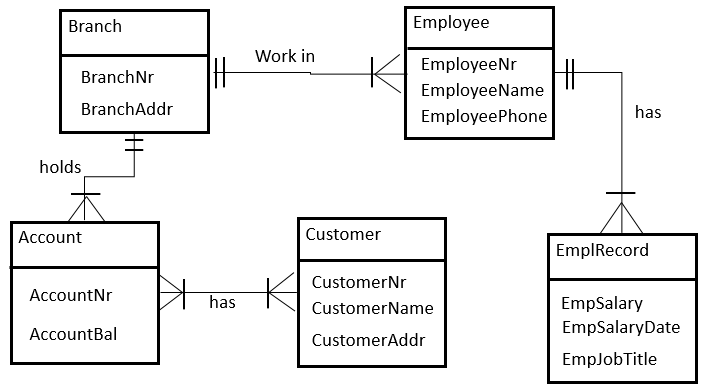
**b)** What are the diagrammatic conventions used in Entity Relationship (ER) modelling to represent data entities and the relationships between them? **(4 marks)**

**c)** Given the ER diagram below and associated enterprise rules – complete the design process to achieve a logical, normalised database design. Document your answer by drawing a final ER design and writing a relational notation description of each table (no need to identify data types).

Enterprise Rules:

* Customers may have multiple accounts in one or more branches
* Joint accounts are permitted
* Customers have only one address
* Branch number, customer number and account number are unique across the entire DB
* More than one employee can share a phone extension in a branch
* Employees may not necessarily have a phone or work in a branch.

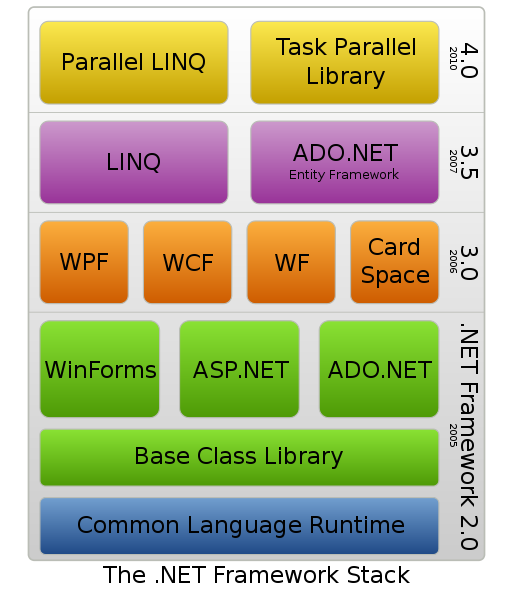
ER Diagram:



**(14 marks)**

**(Total 30 Marks)**

**Q3. ADO.NET**

[](//upload.wikimedia.org/wikipedia/commons/d/d3/DotNet.svg)

**a)** With reference to the diagram above that illustrates the Microsoft .What is the purpose of ADO.NET? **(4 marks)**

**b)** ADO.NET has two main components - “DataSet” and “Data Provider”. What is the purpose of each of these components? **(6 marks)**

**c)** Given the code below which is a console application that connects to a database and displays some data. Answer the following questions:

1. What is the purpose of Lines 20 and 21? What ADO.NET object is created and what is its purpose? **(4 marks)**
2. What is the network address of the database server that this application connects to? What is the name of the database on that server? What kind of authentication is used in the connection? **(4 marks)**
3. Identify the names of the DataSet and Data Provider objects in this program code. **(4 marks)**
4. What is the purpose of Line 37 and 39 ? **(4 marks)**
5. Describe in detail what is happening in Line 46. What type of object is dRow? **(4 marks)**

Contd/..

|  |  |  |
| --- | --- | --- |
| 1 | : | using System; |
| 2 | : | using System.Collections.Generic; |
| 3 | : | using System.Linq; |
| 4 | : | using System.Text; |
| 5 | : | using System.Data; |
| 6 | : | using System.Data.SqlClient; |
| 7 | : |  |
| 8 | : | namespace ADOConsoleApp |
| 9 | : | { |
| 10 | : | class Program |
| 11 | : | { |
| 12 | : | static void Main(string[] args) |
| 13 | : | { |
| 14 | : | string userinput,str1; |
| 15 | : | string studentID, studentFirstName, studentLastname; |
| 16 | : | DataRow dRow; |
| 17 | : | int MaxRows = 0; |
| 18 | : | int counter = 0; |
| 19 | : |  |
| 20 | : | System.Data.SqlClient.SqlConnection con; |
| 21 | : | con = new System.Data.SqlClient.SqlConnection(); |
| 22 | : |  |
| 23 | : | con.ConnectionString="Data Source= SQL4.student.litdom.lit.ie\inst29;Initial Catalog=COLLEGE;Integrated Security=True"; |
| 24 | : |  |
| 25 | : | System.Data.SqlClient.SqlDataAdapter da; |
| 26 | : |  |
| 27 | : | DataSet ds1; |
| 28 | : | ds1 = new DataSet(); |
| 29 | : |  |
| 30 | : | con.Open(); |
| 31 | : | Console.WriteLine("Database Connection Open"); |
| 32 | : | userinput = Console.ReadLine(); |
| 33 | : | //Set up the SQL Query |
| 34 | : | string sql = "SELECT \* From STUDENT"; |
| 35 | : |  |
| 36 | : |  |
| 37 | : | da = new System.Data.SqlClient.SqlDataAdapter(sql, con); |
| 38 | : |  |
| 39 | : | da.Fill(ds1, "STUDENTDATA"); |
| 40 | : |  |
| 41 | : | MaxRows = ds1.Tables["STUDENTDATA"].Rows.Count; |
| 42 | : | str1 = System.Convert.ToString(MaxRows); |
| 43 | : | //Display records |
| 44 | : | while (counter < MaxRows) |
| 45 | : | { |
| 46 | : | dRow = ds1.Tables["STUDENTDATA"].Rows[counter]; |
| 47 | : | studentID = dRow.ItemArray.GetValue(0).ToString(); |
| 48 | : | studentFirstName = dRow.ItemArray.GetValue(1).ToString(); |
| 49 | : | studentLastname = dRow.ItemArray.GetValue(2).ToString(); |
| 50 | : |  |
| 51 | : | Console.Write(studentID); |
| 52 | : | Console.Write(studentFirstName); |
| 53 | : | Console.WriteLine(studentLastname); |
| 54 | : |  |
| 55 | : | counter++; |
| 56 | : | } |
| 57 | : | //Close the database connection |
| 58 | : | con.Close(); |
| 59 | : | Console.WriteLine(""); |
| 60 | : | Console.WriteLine("Database Connection Closed"); |
| 61 | : | userinput = Console.ReadLine(); |
| 62 | : | } |
| 63 | : | } |
| 64 | : | } |

**(Total 30 Marks)**

**PART B: Practical Tasks**

IMPORTANT!! Save your work REGULARLY

**Q.4 DATABASE WEB APPLICATION - DATA PRESENTATION**

The aim of this question is to modify a web application to present data in a formatted table and use object oriented MySQLi API for database interaction.

Instructions:

1. Locate the SQL backup script itschool.sql. Restore it to your MySQL Server.
2. Locate the files dataviewer.php and tablestyle.css in your Q4 Folder.
3. Save the two files in an appropriate folder in your Apache Web Server
4. View the output of the file in your browser. It should appear as it does in Fig 1:

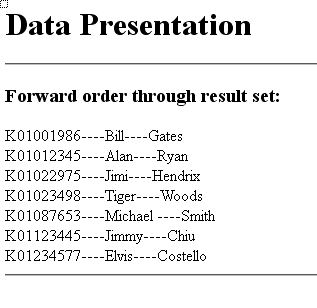


Fig 1 : Output in browser of unmodified dataviewer.php script

Contd/..

Tasks:

1. The dataviewer.php code you have found in the Q4/SOURCEFILES folder uses ext/MySQL procedural code to connect to the database. You are required to change the code to Object Oriented using the ext/MySQLi API for all database interaction  **(15 marks)**
2. Change the code in dataviewer.php to present the data in a formatted table to look like the data table in Figure 2: **(10 marks)**
3. Use the provided stylesheet (tablestyle.css) to achieve required formatting. **(5 marks)**

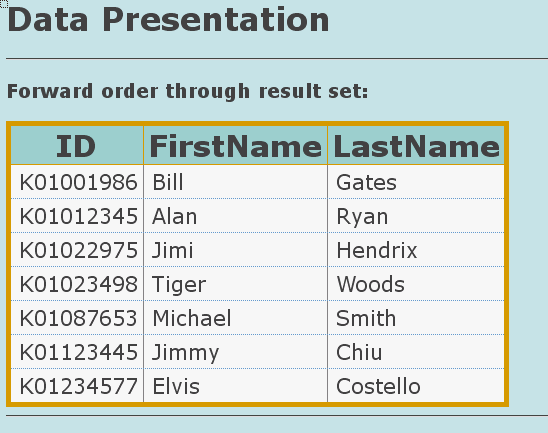


Fig 2 : Desired output in browser of modified dataviewer.php script

Deliverable:

1. Implement the required changes in dataviewer.php and SAVE the file in Q4/ANSWERFILES folder

**(Total 30 Marks)**

**Q. 5 STORED PROCEDURES - MySQL**

Instructions:

* Locate the “projectparts.sql” database backup file in your Q5/SOURCEFILES Folder.
* Execute a restore operation in MySQL to restore the “projectparts” database.

Carry out the following tasks:

1. Create a stored procedure named **sp\_weight\_colour(weight,colour)** which provides all part numbers which are weight exceeds the weight parameter and whose colour is the same as the colour parameter. **(3 marks)**
2. Create a stored procedure **named sp\_project(city)** which provides a list of project names where the project is located in the city defined by the city parameter. **(3 marks)**
3. Create a stored procedure named **sp\_weight(supplierNr)** to report the total weight of each part supplied by the supplier specified in the supplierNr parameter – ie we would like to know what the total weight (quantity times weight) of each part which is supplied by the selected supplier. **(7 marks)**
4. Create a stored procedure **named sp\_proj\_suppliers(city)** to list all of the suppliers involved in projects in the specified city and the total quantity of each part they supply by part number. **(7 marks)**

Contd/..

Deliverables:

* Call each procedure with suitable arguments. Take a screenshot of the result. Save the screenshots in a WORD document.
* Make a backup of the “projectparts” database after you have completed the above tasks.
* Make sure you have included your stored procedures in the backup.
* Save your backup and the WORD document in your exam drive Q5/ANSWERFILES folder.

**(Total 20 Marks)**