**University of Westminster** Computer Science & Software Engineering

|  |  |
| --- | --- |
| **5COSC002W DATABASE SYSTEMS COURSEWORK (2018/19)** | |
| **Module leader** | **Francois ROUBERT** |
| **Unit** | **Database Systems Coursework – INDIVIDUAL COURSEWORK** |
| **Weighting:** | **70%** |
| **Qualifying mark** | **30%** |
| **Description** | **Produce a conceptual data model & a logical data model following given specs. Write SQL and PHP to complete specific tasks. Produce a supporting report.** |
| **Learning Outcomes Covered in this Assignment:** | * LO1 design a data model using standard EERM constructs and convert this model into a relational database schema; * LO2 use UML notations to produce a design that would encompass procedural aspects of a data management system; * LO3 apply normalisation up to BCNF to a set of relational database tables; * LO5 produce SQL queries to retrieve information from one or more tables; * LO6 write a simple application program in a procedural language that interacts with a relational database |
| **Handed Out:** | **Week starting Friday 05 October 2018** |
| **Due Date** | * **FRIDAY 26 OCTOBER 2018 at 23:00:00** – **Part A** * **MONDAY 26 NOVEMBER 2018 at 23:00:00** – **Part A + Part B** |
| **Expected deliverables** | * **TUESDAY 23 OCT 2018 at 13:00:00** – **Intermediary** **Report: Part A**   Report in MS Word or PDF format, font Calibri size 11   * + *1 cover page for part A, student details & tutorial group attended*   + *1 side featuring conceptual ERD*   + *4 data dictionary tables supporting conceptual ERD* * **MONDAY 26 NOV 2018 at 13:00:00 – FINAL REPORT:** **Part A + Part B**   Report in MS Word or PDF format, font Calibri size 11,   * + *1 cover page for part A+B, student details & tutorial group attended*   + *1 side featuring conceptual ERD*   + *4 data dictionary tables supporting conceptual ERD*   + *1 side featuring logical ERD*   + *2-3 sides featuring step-by-step guide for logical ERD*   + *SQL code (DDL) for creating 2 tables and screenshot*   + *SQL code (DDL) for inserting record into one table and screenshot*   + *PHP code to allow user to enter data and screenshot*   + *PHP code to add record to a table and screenshot* |
| **Method of Submission:** | **Online in ‘2018-2019 SUBMIT COURSEWORK’ section of Blackboard site.** |
| **Type of Feedback and Due Date:** | **PART A**: verbal formative feedback on conceptual ERD in tutorial following submission (week 5), failure to attend will restrict opp. for feedback.  **PART A + B**: online feedback and marks 15 working days (3 weeks) after the submission deadline.  **All marks provisional until formally agreed by Assessment Board**. |
| **BCS ACCREDITATION CRITERIA** | 2.1.1 Knowledge and understanding of facts, concepts, principles & theories  2.1.2 Use of such knowledge in modelling and design  2.2.1 Specify, design or construct computer-based systems  2.3.2 Development of general transferable skills  3.1.3 Knowledge of systems architecture  3.2.1 Specify, deploy, verify and maintain information systems |

Assessment regulations

For detailed information regarding University Assessment Regulations on how you are assessed, penalties and late submissions, what constitutes plagiarism etc. please refer to the following website**:** [**http://www.westminster.ac.uk/study/current-students/resources/academic-regulations**](http://www.westminster.ac.uk/study/current-students/resources/academic-regulations)

Penalty for Late Submission

**If you submit your coursework late but within 24 hours or one working day of the specified deadline, 10 marks will be deducted from the final mark, as a penalty for late submission, except for work which obtains a mark in the range 40 – 49%, in which case the mark will be capped at the pass mark (40%). If you submit your coursework more than 24 hours or more than one working day after the specified deadline you will be given a mark of zero for the work in question unless a claim of Mitigating Circumstances has been submitted and accepted as valid.**

**It is recognised that on occasion, illness or a personal crisis can mean that you fail to submit a piece of work on time. In such cases you must inform the Campus Office in writing on a mitigating circumstances form, giving the reason for your late or non-submission. You must provide relevant documentary evidence with the form. This information will be reported to the relevant Assessment Board that will decide whether the mark of zero shall stand. For more detailed information regarding University Assessment Regulations, please refer to the following website:**

[**http://www.westminster.ac.uk/study/current-students/resources/academic-regulations**](http://www.westminster.ac.uk/study/current-students/resources/academic-regulations)

**Coursework Specifications Part A: Conceptual ERD**

**[35 Marks]**

**Part A Project Brief: AudioVizzion**

AudioVizzion is a large retail chain that offers optical and audiology services and essentially sells glasses and hearing devices to the public. AudioVizzion operates globally from a large number of branches located all around Britain. Every branch offers AudioVizzion customers the opportunity to get tested on their vision and/or on their hearing before being able to purchase what the company refers to as Visual Devices (i.e. frames and lenses for glasses) and/or Hearing Devices (i.e. hearing aids).

A customer first contacts AudioVizzion to make a booking for a test, either for a vision test or for an audio test, depending on their specific needs.

Once the booking for the test is agreed and a date and time is set for the test to take place at a specific branch, it is conducted by a specialised member of staff from AudioVizzion: an optician for a vision test or an audiologist for a hearing test.

A vision test typically measures visual acuity, retinoscopy, and refraction, while an audio test simply measures the patient’s hearing level. At the end of a test, the member of staff records a number of comments and from there a prescription is produced.

The prescription provides a detailed summary of the test results, but also the date from which it is available (i.e. the date when the test took place) and the date until when it is available (i.e. the date when the test took place to which 2 years are added)

The customer can then proceed and place an order to purchase a device (either a Visual Device or a Hearing Device) based on the obtained prescription.

If they wish to purchase a Visual Device, they can select a frame for glasses (the company offers many brands and models) and/or lenses (the lenses are identified with their own serial numbers and have different vision types, tints and levels of thinness).

If they are interested in ordering a Hearing Device, they can choose from several makes and models. They can place their orders at any of the AudioVizzion branches, as most convenient, and the order is processed by a sales assistant at the branch who is responsible for the order and for notifying the customer of the expected collection date and time, and for contacting them when their order is actually ready for collection. The sales assistant also subsequently updates the status of the order.

It is to be noted that not all members of staff at AudioVizzion work at a specific branch and that there are other members of staff who are not necessarily involved in the direct provision of optical and audiology services and work in other parts of the company.

**Part A Questions**

You have been hired by AudioVizzion as a database architect to undertake a database project to support the data needs of the company. Your goal in this first part is to produce a high-quality **CONCEPTUAL RELATIONAL DATA MODEL** and to produce a data dictionary to document and support your model.

**⚠️ *Prefix the names of all entities and attributes with your id number starting with w.***

1. Produce a complete **Conceptual Entity-Relationship Diagram** for AudioVizzion. This needs to include all the **entities**, **relationships**, **multiplicities**, **attributes** and **primary keys** that you have identified. This needs to fit on one page of the report.
2. Create a **data dictionary** to document how you identified the **entities** for AudioVizzion. To achieve this, fill in the 2 tables below to summarise and explain how you identified the entities for your data model.

|  |  |
| --- | --- |
| *Entity name* | *Justification* |
|  |  |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
| *General entity* | *Specialised entity* | *Justification* |
|  |  |  |
|  |  |  |
|  |  |  |

*For more information, please refer to page 510 of the 6th edition of the Connolly’s textbook.*

1. Create a **data dictionary** to document how you identified the **relationships** and **multiplicities** for AudioVizzion. To achieve this, fill in the table below to summarise and explain how you identified the relationships and multiplicities for your data model.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Entity name* | *Multiplicity* | *Relationship* | *Multiplicity* | *Entity name* | *Justifications for the multiplicity*  *(4 statements for each relationship)* |
|  |  |  |  |  |  |
|  |
|  |
|  |
|  |  |  |  |  |  |
|  |
|  |
|  |

*For more information, please refer to page 513 of the 6th edition of the Connolly’s textbook.*

Create your own separate table if you identify complex relationships (e.g. non-binary relationships) to identify the entities involved and provide justifications.

1. Create a **data dictionary** to document how you identified the **attributes** and **primary keys** for each entity for AudioVizzion. To achieve this, fill in the table below to summarise and explain how you identified the **attributes** and **primary keys** for your data model.

|  |  |  |
| --- | --- | --- |
| *Entity name* | *Attributes for this entity (include PK)* | *Justification* |
|  |  |  |
|  |  |  |
|  |  |  |

*For more information, please refer to page 516 of the 6th edition of the Connolly’s textbook.*

 **Interactive FAQ**

In order to provide you with the support you may require as a Database Architect and answer any questions you may have about the AudioVizzion brief, an interactive FAQ is offered on padlet, a Web tool to build a virtual wall / bulletin board. This allows you to ask specific targeted questions to the AudioVizzion Managing Directors about the AudioVizzion business so that to improve and refine your conceptual ERD and for these questions and their answers to be shared with the entire class. The padlet is available on <https://padlet.com/soriad/audiovizzion_faq>

**Part A Marks Allocation**

**Part A** will be marked based on the following marking criteria:

|  |  |
| --- | --- |
| Criteria | Mark per component |
| **Correct identification of entities + data dictionary tables** | **10** |
| **Correct identification of relationships + data dictionary table** | **10** |
| **Correct identification of multiplicities + data dictionary table** | **10** |
| **Correct identification of attributes and primary keys + table** | **05** |
| **PART A TOTAL** | **35** |

**Coursework Specifications Part B: Logical ERD, SQL & PHP**

**[65 Marks]**

**Part B Project Brief: Futurity**

Futurity is the careers and professional development service offered by the University of Westminity. Futurity employs several members of staff who act as career consultants to provide guidance and advice to the students who use the service to find employment opportunities. Students can apply to a number of opportunities offered by employers e.g. graduate scheme, jobs and placements. To assist the students with their applications, staff at Futurity organise support sessions to help them with their application forms, CVs and cover letters. Futurity staff also organise a number of events to showcase jobs and graduate schemes.

**Part B Questions**

You have been hired by Futurity as a database architect to undertake a database project to support the data needs of the firm. In this second part, you are given a conceptual data model for Futurity (figure 5) and your first goal is to **map it** onto a high-quality **LOGICAL RELATIONAL DATA MODEL** to logically represent how the key business data needs can be organised as a set of interrelated tables that can then be implemented. These tables need to be interconnected according to the strict rules of the relational model to be implementable. You also have to create a number of tables in SQL and write PHP code to insert some data in these tables.

⚠️ ***Prefix the names of all tables and attributes/columns with your id number starting with w.***

1. Produce a complete **Logical Entity-Relationship Diagram** for Futurity. This needs to include all the **correct tables, relationships, multiplicity constraints, attributes, primary keys** and **foreign keys**. This needs to fit on one page of the report.
2. Provide a **step-by-step guide** explaining how you produced the Futurity **logical ERD** i.e. how you **mapped** the Futurity conceptual ERD into a full **logical relational schema**. To achieve this, write a series of numbered bullet points (e.g. [1], [2], [3], etc.) to explain how you applied the **10 rules of the Logical Data Modelling Methodology** (see Lecture 04), and in which order, to convert the given Conceptual ERD into a Logical ERD. This needs to fit on no more than two or three pages of the report.
3. Write some SQL code to **create the** **Company** and the **Employee tables** in the MySQL RDBMS and **insert 4 records in the Company table.** Followthe instructions below:

* Both tables need to have primary keys and one of the tables needs to have one foreign key.
* No field should be left empty.
* Remember to prefix the names of the tables and columns with your id number starting with w.
* Include the **SQL code** used to create both tables and **screenshots of the structure of the tables**.
* Include the **SQL code** used to insert the 4 records in the Company table (primary keys: 10, 20, 30 and 40) and a **screenshot of the content of the Company table**.
* The screenshots should clearly show your student id numbers (in the names and of the table and attributes and also right above the tabs for the content and the structure of the table in PHPMyAdmin).

1. Create a PHP file **EITHER** called **addemployee\_simple.php** **OR** **addemployee\_advanced.php** to display a Web-based form to allow an administrator to enter the details of new employee in the Employee table in the MySQL RDBMS. Followthe instructions below:

* Choose either **addemployee\_simple.php**, the **simple version** (see figure 1), or **addemployee\_advanced.php,** the **more advanced version** (see figure 2) and write the PHP code accordingly.
* Your page needs to clearly display a **Web form** to capture the details of a new employee.
* Include the **PHP code** and a **screenshot of your Web page,** either for the simple version (just like on figure 1) **OR** the advanced version (just like on figure 2).
* The screenshot should clearly show your student id number (in the URL bar).

1. Create a PHP file called **getemployee.php** to display a Web-based confirmation page displaying the added customer as for the instructions below.

* Your page needs to clearly display a confirmation that the **last added employee** was added into the database (see figure 3).
* Your page needs to provide a certain level of **SQL error handling** to validate the data entered e.g. correct data type, validity of the primary key, etc.
* Include the **PHP code**, a **screenshot of your Web page** (see figure 3) and a **screenshot of the content of the Employee table with the last record as having been added** (see figure 4).
* The screenshots should clearly show your student id number, in the URL bar for the Web page and right above the tabs for the content and the structure of the table in PHPMyAdmin.

**Part B Marks Allocation**

**Part B** will be marked based on the following marking criteria:

|  |  |
| --- | --- |
| Criteria | Mark per component |
| **Correct mapping of complex relationships + sound explanation** | **12** |
| **Correct mapping of relationships (M:M) + sound explanation** | **16** |
| **Correct mapping of simple relationships (1:M & 1:1) + sound explanation** | **07** |
| **Correct SQL Data Definition Language (DDL) query to create table** | **05** |
| **Correct PHP code to add and retrieve data into and from table** | **25** |
| **PART B TOTAL** | **65** |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

⚠️ **Key Requirements for the entire coursework**

* Only **UML notations** are accepted, as introduced in this module.
* You need to **prefix** all your entities and attributes with “**w + the 7 digits of your ID number**” as provided by the University.

For example, if my name is Francois Roubert and my ID number is w1234567, when I identify the entity “Module” and its attributes “moduleCode”, “moduleName” and “moduleType”, I will have to represent it this way

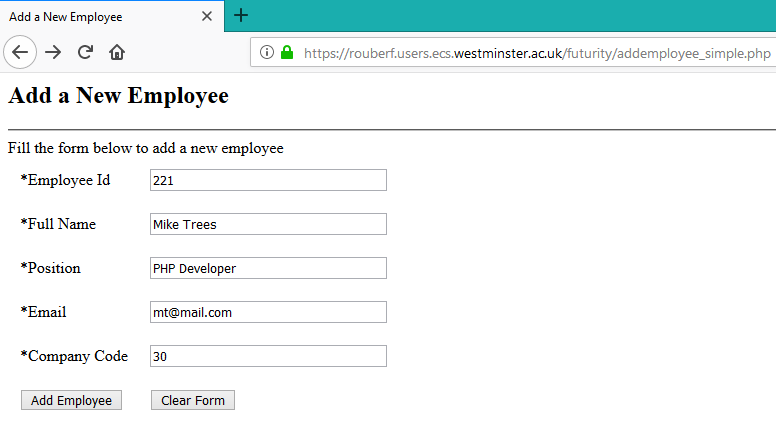
w1234567\_Module

w1234567\_moduleCode{PK}

w1234567\_moduleName

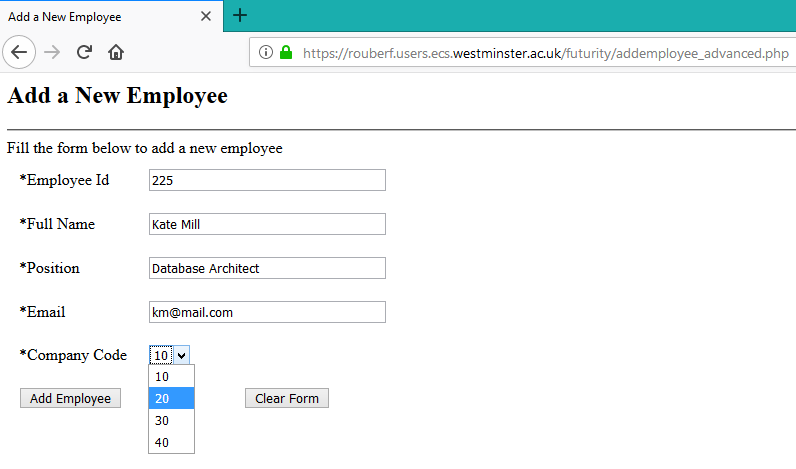
w1234567\_moduleType

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



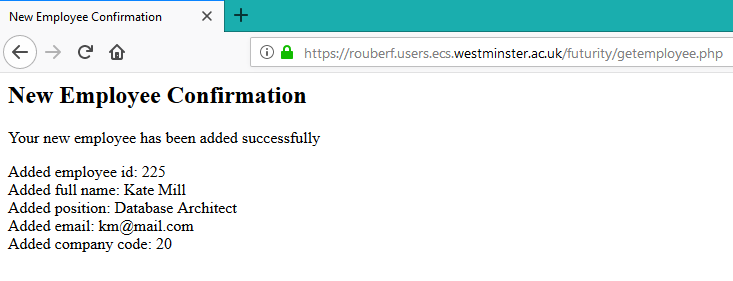
Value of the company code entered in a text box

**Figure 1: screenshot for addemployee\_simple.php**



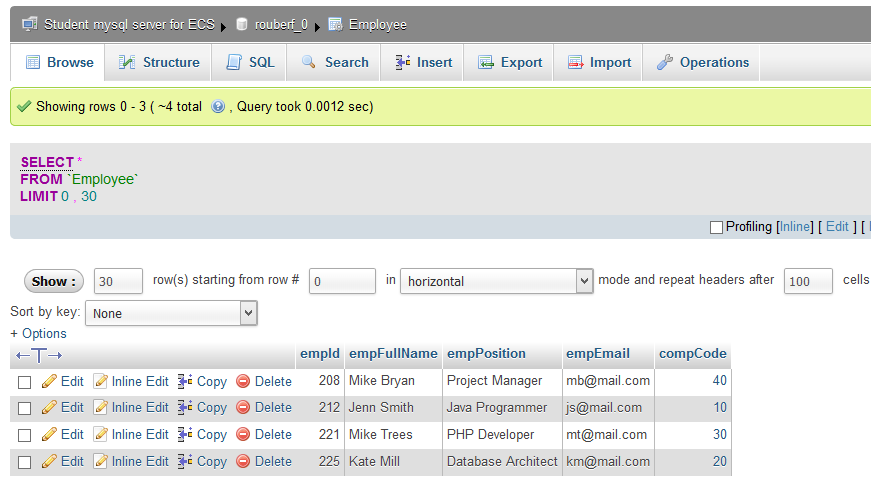
Value of the company code selected using a drop-down list

**Figure 2: screenshot for addemployee\_advanced.php**



Confirmation of the latest record being added onto the database table

**Figure 3: screenshot for getemployee.php**



Last record added in table

**Figure 4: screenshot of the content of the Employee table**

**Figure 5: Futurity CONCEPTUAL Entity Relationship Diagram for PART B**

**Opportunity**

oppCode{PK}

oppName

oppDescrip

oppDeadline

startDate

Grad\_Scheme

schemeDuration

Job

Placement

placementType

Company

compCode{PK}

compName

compDescrip

Employee

empId{PK}

empFullName

empPosition

empEmail

Staff

staffNo{PK}

staffFullName

staffEmail

Event

eventCode{PK}

eventDateTime

eventDescrip

Presentation

duration

Job\_Fair

Student

stdtId{PK}

stdtFullName

stdtEmail

applicantDescrip

Graduate

gradDate

degreeClassif

Application

applicId{PK}

applicDateTime

applicStatus

Interview

intervCode{PK}

intervDateTime

intervFeedback

Support\_Session

sessCode{PK}

startDateTime

endDateTime

sessFeedback

**is for**

places

leads to

offers

employs

takes part in

is assigned

provides

assists with

is presented at

is advertised at

{optional, or}

{optional, and}

{optional}

0..\*

1..1

0..\*

1..1

1..1

0..\*

1..1

0..\*

0..\*

1..\*

0..\*

0..\*

1..1

0..\*

0..\*

1..1

0..\*

1..1

0..\*

attends

0..\*

1..1

0..1

0..\*

1..1

is allocated

0..\*

1..\*

0..\*

0..1

mentors