|  |  |
| --- | --- |
| **Programmes:** | MSc Computer Science / MSc Information Science (Data Analytics) |
| **Module Code:** | KC7013 |
| **Module Title:** | *Database Modelling* |
| **Distributed on:** | Friday 27th March 2020 |
| **Submission Time and Date:** | Wednesday 20th May 2020 by 23.59 GMT |
| **Date by which Work and Feedback will be returned to Students:** | 15th June 2020 |
| **Weighting** | This coursework accounts for 50% of the total marks for this module |
| **Submission of Assessment** | **Electronic Management of Assessment (EMA):** Please note your assignment is to be submitted electronically and it will be submitted online  It is your responsibility to ensure that your assignment arrives before the submission deadline stated above. See the University policy on late submission of work. |

**Instructions on Assessment:**

* You are expected to produce a word-processed answer to this assignment. Please use Arial font and a font size of 12. For SQL code and output, you can use courier new font, which preserves SQL format and layout.
* You are required to use the Harvard Style of referencing and citation. The *“Cite them right”* guide is recommended for *referencing and citation* (Pears and Shields, 2008) which should be followed throughout your answer especially Part 3. A good alternative is Northumbria (2018). Please do not include references to lecture notes.

**Mapping to Programme Goals and Objectives**

This assignment covers the following learning outcomes for the module:

Knowledge & Understanding:

1. Key concepts of data warehousing.

Intellectual / Professional skills & abilities:

1. Conceptual data modelling, relational database design and implementation in SQL & PL/SQL, and object-based databases.
2. Design and Implementation of a data warehouse using Oracle database system.

**Assessment Regulations**

You are advised to read the guidance for students regarding assessment policies. They are available online [here](https://www.northumbria.ac.uk/about-us/university-services/academic-registry/quality-and-teaching-excellence/assessment/guidance-for-students/).

**Late submission of work**

Where coursework is submitted late without approval, after the published hand-in deadline, the following penalties will apply.

For coursework submitted up to 1 working day (24 hours) after the published hand-in deadline without approval, **10% of the total marks available for the assessment** (i.e.100%) **shall be deducted** from the assessment mark.

Coursework submitted more than 1 working day (24 hours) after the published hand-in deadline without approval will be regarded as not having been completed. **A mark of zero will be awarded for the assessment and the module will be failed**, irrespective of the overall module mark.

These provisions apply to all assessments, including those assessed on a Pass/Fail basis.

The full policy can be found [here](https://www.northumbria.ac.uk/about-us/university-services/academic-registry/quality-and-teaching-excellence/assessment/guidance-for-students/).

***Students must retain an electronic copy of this assignment (including ALL appendices) and it must be made available within 24hours of them requesting it be submitted.***

**Academic Misconduct**

The Assessment Regulations for Taught Awards (ARTA) contain the ***Regulations and procedures applying to cheating, plagiarism and other forms of academic misconduct***.

The full policy is available at [here](https://www.northumbria.ac.uk/about-us/university-services/academic-registry/quality-and-teaching-excellence/assessment/guidance-for-students/)

You are reminded that plagiarism, collusion and other forms of academic misconduct as referred to in the Academic Misconduct procedure of the assessment regulations, which are taken very seriously. Assignments in which evidence of plagiarism or other forms of academic misconduct is found may receive a mark of zero.

**Criteria for success:**

### *For textual components :*

**80-100%** - The description will excellently cover all the specific topics requested. The written work will be fluent, clearly presented and of out-standing quality.

**70-79%** - The description will comprehensively cover all the specific topics requested. The written work will be fluent and clearly presented and of distinctive quality.

**60-69%** - The student will show a very good knowledge of the specific topics, with very good presentation skills and quality.

**50-59%** - The student will show an above average knowledge of the specific topics, with above average presentation skills and quality.

**40-49%** - There will be an adequate description of a significant proportion of the topics requested. There will be no major failures in presentation clarity.

**Less than 40%** - There will be little or no information conveyed in an intelligible manner on the specific topics requested.

### *For ERD, SQL and other database technical components:*

**80-100%** - The students will produce exceptional models (conceptual/logical/physical), and will demonstrate the use of notation/language, which have outstanding syntactic accuracy (e.g., adhering to ER modelling standards, error free SQL code), with exceptional semantic relevance (e.g., are relevant to the requirements of the particular scenario).

**70-79%** - The students will produce fully complete models (conceptual/ logical/ physical), and will demonstrate the use of notation/language, which have high syntactic accuracy, with high semantic relevance.

**60-69%** - The students will produce almost complete models (conceptual/ logical/ physical), and will demonstrate the use of notation/language, which have appropriate syntactic accuracy with reasonably well semantic relevance.

**50-59%** - The students will produce fairly complete models (conceptual/ logical/ physical), and will demonstrate the use of notation/language, which have adequate syntactic accuracy with reasonable semantic relevance.

**40-49%** - The students will produce models, and will demonstrate the use of notation/language, which have some syntactic accuracy and semantic relevance.

**Less than 40%** - The students will not produce sufficient models, and/or will be unable to demonstrate the use of notation/language with significant syntactic accuracy and/or significant semantic relevance.

**Assessment Background and Scenario**

This assessment is based on two scenarios as follows:

1. The scenario *‘Academic Information System (AiS)”* of a fictitious university called University of Gharnata**.** The university wants to develop an information system to support its academic activities. The details of this scenario are provided in Appendix 1.
2. The Sales History (SH) Data Warehouse scenario. SH is a sample database schema provided by Oracle, which has been extensively used in the Oracle’s Data Warehousing Guide (Lane, 2013). The details of this scenario are provided in Appendix 2.

**Assignment Questions**

***Part 1: Implementing object-relational database for AiS (50 Marks)***

This part is based on the ‘AiS’ scenario as described in Appendix 1.

*(A) Using object-oriented / extended entity relationship modelling, produce a conceptual and logical database model for AiS.*

(8 marks)

You have the option to produce a fresh conceptual and logical database model using object-oriented or extended entity relationship modelling or incorporate advanced entity modelling concepts (e.g., aggregation & composition, inheritance or generalisation / specialisation/, etc) into your entity relationship model for AiS from your assignment 1.

*(B) Create an object-based database for AiS using object-relational (O-R) features of Oracle 10g//11g/12c based on the conceptual and logical database model from (A) above.*

(20 marks)

Note that all relationships (e.g., one-to-one, one-to-many, many-to-many) must be bi-directional. Moreover, each to-many side of a relationship (e.g., in case of many-to-one and many-to-many) should be implemented using nested tables. In addition, your implementation should allow all objects to be **shareable** (i.e. all relationships should be **REF** based). Oracle (2005a) provides a more detailed documentation on Oracle object-relational features.

*(C) Populate the O-R version of your database using data from relational version of the AiS database you developed in Assignment 1.*

(10 marks)

Using PL/SQL and/or SQL, populate your O-R database by querying the relational tables you implemented earlier during your Assignment 1 solution. This task can be seen to contain the following sub-tasks:

(i) Creating / inserting objects in your object tables.

(ii) Populating one-to-many, many-to-one, and many-to-many relationships among the objects created in sub-task (i).

*(D) Query the O-R version of your AiS database*

(8 marks)

For this part, you must use PL/SQL procedures (2 procedures, each worth 4 marks). Answer the following questions after populating the object-relational database:

1. Display names of students, details of the course they study, details of the module they have studied and their marks for all postgraduate students.
2. Display details of all people of ‘AiS’ (students and academic staff), e.g., their names, their home addresses and name of the department where they work or study.

*NOTE the following important points for Part 1 (A to D):*

* You must submit all the SQL and PL/SQL code used for creating, populating and querying the O-R database.
* You must submit the output of running your code (e.g., SPOOL files).

*(E) Compare and contrast your relational and object-relational versions of the AiS database. You should highlight the advantages and disadvantages of both versions. Comment on which version will best suit the AiS database.*

(4 marks)

***Part 2:*  *Data Warehousing Tasks (50 Marks)***

This part is based on the Sales History scenario as described in Appendix 2.

***You must submit all the SQL queries and any other code that you wrote in answering any of the tasks / questions (e.g., the use of Explain Plan statements for the queries and their SPOOL outputs).***

1. Study the index definitions in sh\_idx.sql. Discuss in detail (using cost-based analysis) why these indexes (*at least two of them*) are useful for answering queries over the SH2 and DWn versions of the database. *You should not run the sh\_idx.sql script at all.*

(10 marks)

1. Identify *two* new indexes and justify why they could be useful. Write the SQL code for creating these indexes under your DWn user. Give example queries with cost-based analysis for both DWn (which will have the new indexes) and SH2 users (which will not have any of your new indexes).

(10 marks)

1. Given the materialized views (MVs) defined in sh\_cremv.sql, discuss in detail why these MVs are useful for users of the SH database. You should provide detailed examples (cost based analysis, e.g., using Explain Plan for running sample queries on both SH2 and DWn) to illustrate your answer. *You should not run the sh\_idx.sql script at all.*

(10 marks)

1. Identify *two* other MVs based on the base tables in the SH schema under your DWn user and justify why they would be useful. Write the SQL code for creating these MVs.

(10 marks)

1. Prior to the introduction of the aggregation function ROLLUP, there was no possibility to express an aggregation over different levels within one SQL statement without using the set operation UNION ALL. Every different aggregation level needed its own SQL aggregation expression, operating on the exact same data set *n* times, once for each of the *n* different aggregation levels. With the introduction of ROLLUP in the recent editions, Oracle provided a single SQL command for handling the aggregation over different levels within one single SQL statement, not only improving the runtime of this operation but also reducing the number of internal operations necessary and reducing the workload on the system.
2. Using ROLLUP, write an SQL query over the SH schema under your DWn user involving **one** fact table (SALES or COSTS) and at **least two** dimension tables. Provide reasons why your query may be useful for users of the SH data warehouse.

(3 marks)

1. Using set operation UNION ALL (and not ROLLUP), write an SQL query that produces the same result as the query in (a) above.

(3 marks)

1. Using EXPLAIN PLAN, provide a detailed discussion analysing costs of evaluating the above queries (i.e. with and without ROLLUP).

(4 marks)

**Hand-in procedure**

You should hand in your answer for this assignment as a single word processed document electronically on the ELP. A sample assignment template file will be uploaded on the ELP near the submission deadline.

# Appendix 1

**‘Academic Information System (AiS)” Scenario**

The scenario described here is that of a fictitious university called *University of Gharnata***.** The university wants to develop an information system to support its academic activities.

The university has several academic departments. Each department provides one or more academic courses. Each course is composed of several modules, where a module may be part of more than one course. A student enrolls on a course and every year takes a specified number of modules. Note that several students are usually registered for a course. Every student is assigned a tutor at the start of the course, who is a faculty member (e.g., lecturer) in the department providing the course. A faculty member works for a department and usually teaches on several modules. Each module has a module tutor who is also a faculty member. A faculty member may be tutor for several modules.

Each department is chaired by a professor, who is a faculty member and works for the same department. It is important that the system makes a note of when a professor became the chair of a department. The details of a department include its name, a primary location / address where its main building is located, a secondary address where it may provide its services when needed, telephone and fax numbers, etc.

Each course is assigned a course leader (a faculty member), who manages the day-to-day issues of the course. Details of a course include a name, whether it is undergraduate/postgraduate/research course, standard duration of the course in months and the number of credit hours to complete to pass the course.

For each module, the system needs to store its details (e.g., title, number of credit hours, level (1, 2, 3, M, R), etc). For each student who is doing a module, the system needs to record marks.

For each student, the system needs to store details like name (first, middle, last), term address (street, city, region, postal code, etc), home address, telephone numbers, email address, date of birth, gender, and details of next-of-kin (e.g., name, address, relationship, telephone numbers). The system needs to record whether a student is from the UK, EU, Commonwealth or other.

Each faculty member is assigned a line manager, who is also a faculty member and works in the same department. Details of a faculty member include name, home address, office location and room number, telephone extension, email address, home and mobile phone numbers, data of birth, gender, next-of-kin details, date joined the university, salary, and details of qualifications. When a faculty member is assigned to teach on a module, the system needs to record how many hours a week he/she is expected to be teaching on that module.

# Appendix 2

**The Sales History (SH) Data Warehouse**

SH is a sample database schema provided by Oracle, which has been extensively used in the Oracle’s *Data Warehousing Guide* (Lane, 2013). The SH schema, as shown in Figure 3, consists of a big fact table, SALES, and five relatively small dimension tables: TIMES, PROMOTIONS, CHANNELS, PRODUCTS and CUSTOMERS. The additional COUNTRIES table linked to CUSTOMERS creates a simple snowflake. The model and the attributes are aimed at demonstrating data warehousing functionality like star transformation and query rewrite. They do not necessarily represent the optimal approach for this kind of data warehouse in real productive environments; and such a design would be driven more by business requirements than by the star itself. Table 1 shows the cardinalities of the tables in the SH schema.

|  |  |
| --- | --- |
| **Table** | **Number of Rows** |
| Customers | 50,000 |
| Countries | 19 |
| Products | 10,000 |
| Sales | 10,16,271 |
| Costs | 7,87,766 |
| Times | 1,461 |
| Promotions | 501 |
| Channels | 5 |

Table 1: Cardinalities of SH tables (as per SH2 implementation).

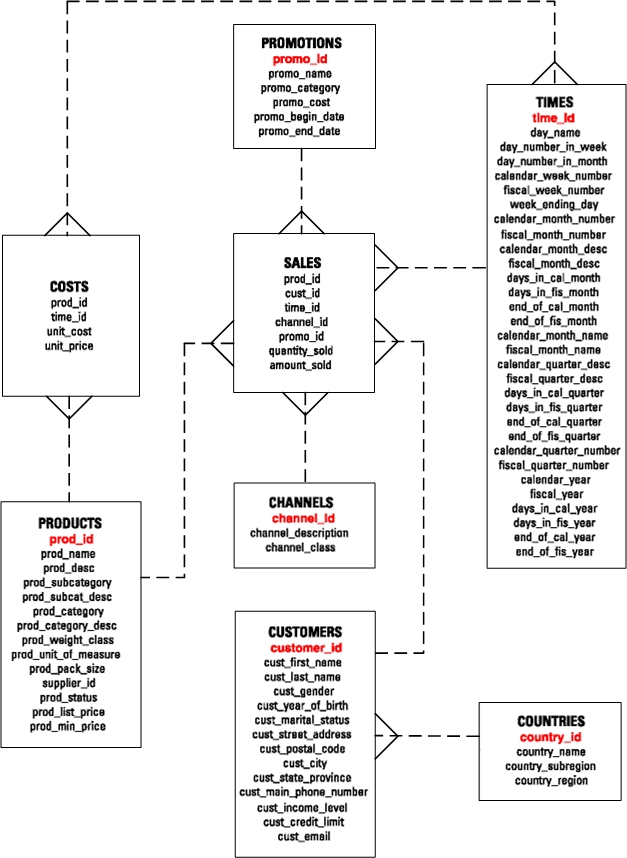


Figure 3: SH Star Schema.

Usually, a star schema includes a single fact table and few dimensional tables, however, the SH schema designer identified another useful fact table called COSTS, which is linked to the TIMES and PRODUCTS dimensional tables. Figure 3 shows two fact tables: SALES and COSTS.

**Implementation of SH schema**

Table 2 shows the files that constitute the implementation of the SH schema using Oracle 10g/11g/12c.

|  |  |
| --- | --- |
| **File** | **Description** |
| sh\_cre.sql | Script for creating the base tables of the SH data warehouse. |
| sh\_cons.sql | Script for enabling constraints. |
| sh\_drop.sql | Script for dropping the SH schema objects. |
| sh\_idx.sql | Script for creating indexes. |
| sh\_pop1.sql | Script for loading data into COUNTRIES and CHANNELS tables. |
| sh\_pop2.sql | Script for populating the TIMES dimensional table. |
| sh\_schema\_objects.sql | Script for calling other scripts e.g. sh\_cre.sql for creating base tables and populating the small dimensional tables. |
| sh\_cremv.sql | Script for creating materialized views. |
| sh\_ext\_table.sql | Script for creating an external table. |
| sh\_main.sql | Script that calls several other scripts including e.g. sh\_cremv.sql, sh\_idx.sql etc. |
| sh\_costs.sql | Script for populating the COSTS fact table. |
| sh\_hiera.sql | Script for creating dimensional hierarchies, which are called DIMENSION in Oracle. |

**Table 2: Files used to implement the SH schema.**

**Two versions of SH schema/database**

* 1. The default and optimized version of the SH database is created under the SH2 user. You have *read-only* access to query any of the tables of SH database by prefixing any table or view name by “SH2.”, e.g., to query the contents of CHANNELS table, use:

SELECT \* FROM SH2.CHANNELS;

* 1. An un-optimized version of the SH database has been created under your own username DWn. You will be given a special username and password for doing this part of the assignment (i.e. an Oracle account starting with DW and followed by a number e.g. DW1, DW2, etc). Throughout this part, you **MUST** use this special username. However, it is assumed that you will not create any indexes on any of the tables in your own personal version of the SH until asked to do specifically. Moreover, neither materialized views (see sh\_cremv.sql) nor dimensional hierarchies (see sh\_hiera.sql) should be created in your version of the SH database.

**Important Note**

Whenever, you need to *assess the performance* of existing database structures (e.g., materialized view, index, dimensional hierarchies) you must refer to SH2 as above (using the “SH2.” prefix). Any new database structures that you need to create should be created under your username. Likewise, when you need to assess the performance of any new database structures that you create under your DWn username, you need to prefix all tables/views/materialised views accordingly, e.g., to query the contents of your CHANNELS table, use:

SELECT \* FROM DWn.CHANNELS;

**References**

Lane, P. (2013) Oracle Database Data Warehousing Guide, 11*g* Release 2 (11.2). Part Number E25554-02. Available at:

<https://docs.oracle.com/cd/E11882_01/server.112/e25554/toc.htm> (Accessed: 17 January 2020).

Northumbria (2018) Quick guide to Referencing and Plagiarism. Available at: <https://cragside.northumbria.ac.uk/Everyone/skillsplus/database_uploads/87.pdf> (Last accessed: 11 September 2019)

Pears, R. and Shields, G. (2008) Cite them right: the essential referencing guide. Newcastle upon Tyne: Pear Tree Books. Available at: <http://nuweb2.northumbria.ac.uk/library/skillsplus/loader.html?55388321> (Last accessed: 11 September 2019)

Oracle (2005a) Application Developer's Guide - Object-Relational Features 10g Release 2 (10.2). Part Number B14260-01. Available at: <http://download.oracle.com/docs/cd/B19306_01/appdev.102/b14260/toc.htm> (Accessed: 17 January 2020).

Oracle (2005b) Oracle Database PL/SQL User's Guide and Reference 10g Release 2 (10.2). Available at: <http://docs.oracle.com/cd/B19306_01/appdev.102/b14261/toc.htm> (Accessed: 17 January 2020).