|  |
| --- |
| OVERVIEW This first part of the project will require the completion of a database design for a given case study. It will require you to undertake logical and physical data modelling making appropriate decisions about the database design based on a case study specification you have been provided with. It is possible, and recommended, that you undertake this element as part of a group but you can complete it as an individual if you prefer.  The second part of the project will require that you build a physical database in Oracle that implements the design you have developed, populate this with data and then build some more complex queries. This part of the project must be completed individually. Each individual must generate their own data, which they must ensure is different to other members of their groups, and there should be sufficient difference between the queries they implement.  Please read the details of the assignment carefully and ensure you understand what is required both in terms of content and in terms of submission.  Please use your lab classes to get assistance with this assignment and to complete as much of it as possible. |
| DUE DATE/TIME **Sunday 8th December 2019 23:59** |
| MARKS ACHIEVABLE This assignment will be completed in two parts each of which will be marked out of 100. You will also receive an overall total mark.  The overall mark you receive will be weighted in the calculation of your final CA mark to reflect that overall it counts for 30% of the CA of the module. |

|  |
| --- |
| NOTES  1. Unfair practice is a very serious offence in the DIT and you must acknowledge any material used by including a referenced bibliography in your report. Any issues will be investigated and those considered serious will be handled via the DIT Plagiarism policy (details are available in the General Assessment Regulations). 2. Assignments must be submitted via Brightspace through the assignment section. Email submissions will be ignored. 3. Extensions due to acceptable personal circumstances must be requested by email in advance of the deadline. 4. For late submissions (i.e. without an agreed extension), a penalty of 5% will be applied for every day a submission is late. 5. No submissions will be accepted after Friday December 13th 2019 @ 23:59 unless an extension has been agreed.   **NB**: Anything submitted later than this date without agreement will be ignored.   1. Assignments which do not adhere to the requirements or which are submitted incorrectly will attract a penalty of up to 5% (e.g. incorrectly named submissions, omitting student details from model/sql file). 2. Email submissions will not be accepted. 3. No resubmission of assignments after feedback is given is allowed. |

|  |
| --- |
| INSTRUCTIONS  1. Form your group (of 3 or maximum 4 students) and add your group details to this google doc <https://docs.google.com/spreadsheets/d/1_XxNejq3WFzjcWqwNDCGojbIuEz6Fy20oiHP5s9w3a0/edit#gid=0>   Assign the next available group number (Group 1, Group 2 etc.) and add the names of the students in your group.   1. Working as a group or an individual if you prefer, create a Data Model (Logical, Relational and Physical) using Oracle Data Modeller for the Case Study provided (see file in bright space). 2. Working as an individual    * Create the SQL to create the physical database based on the data model created.    * Create the SQL to insert data into the relevant tables.    * Create the SQL to meet the requirements set. |
| REQUIREMENTS FOR DATA MODEL For the case study provided (see case study file in Brightspace) you are required to complete the following:   1. Create a logical, relational and physical data model using Oracle Data Modeller. 2. You need to ensure that you have adhered to the following for the data model you submit:    * You should include a note on the logical model the states the following:      + Student Number(s) of group members      + Name(s) of group members      + Programme Code(s) of group members;    * All entities are named appropriately for the case study e.g. Customer rather than E/1;    * Entities start with a capital letter, attributes start with a lowercase letter;    * Appropriate datatypes chosen for your domain types      + Where a datatype requires a size/precision/scale this should be provided;      + In the case study you will be provided with specific guidance for some decisions but must use your knowledge and judgement to make others;    * Appropriate domain types used for your attributes;    * Primary keys identified for each entity;    * Foreign keys identified where appropriate;    * Appropriate NOT NULL, UNIQUE and CHECK constraints are implemented;    * Name your key constraints, unique and check constraints according to the following conventions:      + Primary keys: should be named for the entity followed by \_PK e.g. Customer\_PK      + Foreign keys: should be named for the two entities involved followed by \_PK: e.g. a foreign key between Customer and Order should be Customer\_Order\_FK      + Check constraints: should be named for the attribute and the type of check e.g. CHECK\_SALEAGE CHECK SALEAGE <10 (in this example CHECK\_SALEAGE is the name of the constraint). 3. You need to create a short word document outlining key decisions you made. The decisions you need to document are those relating to:    * + Entity choices and naming (Logical and Relational)      + Domain types      + Attribute Domain types      + Keys (Primary, Foreign, Alternate)      + Relationships (Logical and Relational)      + Constraints. |

|  |
| --- |
| REQUIREMENTS FOR SQL  1. You need to generate statements to create the tables from your physical model.    * Table, attribute and constraints need to match your model (including naming).    * You can generate this using data modeller. 2. You need to generate data to populate your tables to fulfil the queries required and create the insert statements to achieve this.  * Approx. 5 rows per table will be needed. * Data should be persisted. * **NOTE1:** If you worked on the model as a group this data needs to be sufficiently different from that of your other group members. * **NOTE2:** You need to be careful in the data you choose to insert. You need to ensure that the queries you design will result in data being returned for all the queries/alterations you are asked to do.  1. Write SQL to demonstrate the following:    * One UPDATE/DELETE using a subquery.    * One query using a selection function (CASE/DECODE)    * One INNER join using a GROUP Function    * One INNER join using a GROUP Function with a restriction on the group (using a different group function to the previous join)    * One LEFT OUTER Join    * One RIGHT OUTER Join (using different tables to your right outer join)    * One UNION    * One INTERSECT    * One VIEW (which can use any of the SQL you have previously created). |

|  |
| --- |
| SUBMISSION  1. **Part 1 (Data Model)**   You can only submit your CA via the relevant assignment box in bright space **CA PROJECT PART I SUBMISSION.**   * ONE SUBMISSION PER GROUP (Up to 5 attempts at submission allowed) to be made by ONE member of the group. * You will need to submit One ARCHIVE FILE (.zip, .rar)named with your group namee.g. CA Project GROUP 2.Zip * This should contain your   + Oracle data modeller file (.dmd) plus any additional files your model needs (a folder of the same name plus your datatypes as .xml). Your data model and datatypes file should be named with the name of your GROUP e.g. CA Project Group 1.dmd, CA Project Group1.xml   + A short word document outlining key decisions you made e.g. CA Project Group1.docx.  1. **Part 2 (SQL)**   You can only submit your CA via the relevant assignment box in Brightspace **CA PROJECT PART 2 SUBMISSION.**   * ONE SUBMISSION PER PERSON (Up to 5 attempts at submission allowed). * The file should be named with your student number .SQL e.g. D123456.SQL   + You need to include your Student Number, Name and programme code at the start of the file.   + You need to clearly comment each SQL/set of SQL statements you include to indicate what aspect of PART 2 the SQL is intended to address.   + The SQL must execute on the DIT Oracle server.   **NOTE**: Attempts to submit either part of the project via email will be ignored. |

## Marking Scheme

|  |  |
| --- | --- |
| **Data Model** | **0** |
| Entities correctly identified and named with correct attributes | 15 |
| Domain types correct | 10 |
| Appropriate domain types used for attributes | 5 |
| Correct primary keys identified for each entity | 10 |
| Relationships between entities are of correct type with correct cardinality on correct attribute | 15 |
| NOT NULL constraints correctly defined | 10 |
| VALUE constraints correctly defined and named | 10 |
| Foreign key constraints correctly defined and named | 10 |
| Relational model contains valid entities, attributes and relationships | 10 |
| Physical model created for Oracle db | 5 |
|  | 100 |

|  |  |
| --- | --- |
| **SQL** |  |
| Create tables plus constraints (keys + value) consistent with data model | 5 |
| Insert data (which results in output being generated for all queries and differentiated from the group if relevant) | 10 |
| One UPDATE/DELETE using a sub-query | 5 |
| One query using a selection function (CASE/DECODE) | 5 |
| One INNER join using a GROUP Function | 10 |
| One INNER join using a GROUP Function with a restriction on the group (using a different group function to the previous join) | 10 |
| One LEFT OUTER Join | 15 |
| One RIGHT OUTER Join (using different tables to your right outer join) | 15 |
| One UNION | 10 |
| One INTERSECT | 10 |
| One VIEW (can use previous SQL) | 5 |
|  | 100 |