ENGLISH

University of South-Eastern Norway

EXAMINATION INFORMATION PAGE

Home exam

General information about the exam:

Subject code: BID3000-1

Subject name: Business Intelligence and Data Warehouse

Responsible course manager: Ali Chelli

Campus: Ringerike

Faculty: HH

Assignment given (date and time): 05.12.2023 16:00

Submission time (date and time): 19.12.2023 16:00

Number of assignments: 1

Number of attachments: 3

Number of pages: 5 (excluding attachments)

Important information for all types of home exams:

A. EXAMINATION SUPPORT MATERIAL AND COOPERATION:

Permitted examination support material: All examination support material permitted.

Examination form (tick off): Individual ☐ Group ☒

In an individual home exam, it is not allowed to collaborate with others on how to write the answer. The text, method for solving the task, academic reflections, etc. (your answer) that is submitted for the exam should be your own work (or the group's if it is a group exam). It is not allowed to have other persons or tools/functions (such as artificial intelligence) write the exam answer for you.

We would like to remind you that plagiarism control will be conducted on all answers, and unnatural similarities, illegal use of aids, etc. - or if you have not written the answer yourself - will result in USN creating a case for cheating. Cheating can result in the annulment of the answer, loss of the right to take exams at USN and other universities and colleges, or temporary suspension from USN. Description of individual exams and illegal collaboration can be found on the USN intranet.

Exam assignment:

The assignment consists of a pdf file, an attached SQL file called “classicmodels.sql”, an attached power BI file called “Dashboard_0.pbix” and an attached pdf document called “Restoring A database Using MySQL Workbench.pdf”.

The assignment counts for 40% of the final grade in the course. It can be solved in groups of 4, 5 or 6 students. You must submit your answer as a pdf document in WiseFlow. **Remember to upload the KTR files and the Power BI file as attachments in WiseFlow.** The names of the KTR files should be chosen as indicated in the specific questions below. **Remember to indicate the candidate numbers of all participants.**

In case you have any questions regarding the exam please send those questions to the exam office by email: eksamen.ringerike@usn.no

You will use the Classic Models database as the source system to create a Business Intelligence solution. Classic Models database is provided to you on WiseFlow as an SQL file called “classicmodels.sql”.

Your first step consists of restoring the database “classicmodels” in MySQL using the file called “classicmodels.sql”. More information on how to do this task can be found in the attached document “Restoring A database Using MySQL Workbench.pdf”.

Classic Models is an online store that sells models of cars, motorcycles, planes, and boats. The data model looks like this:

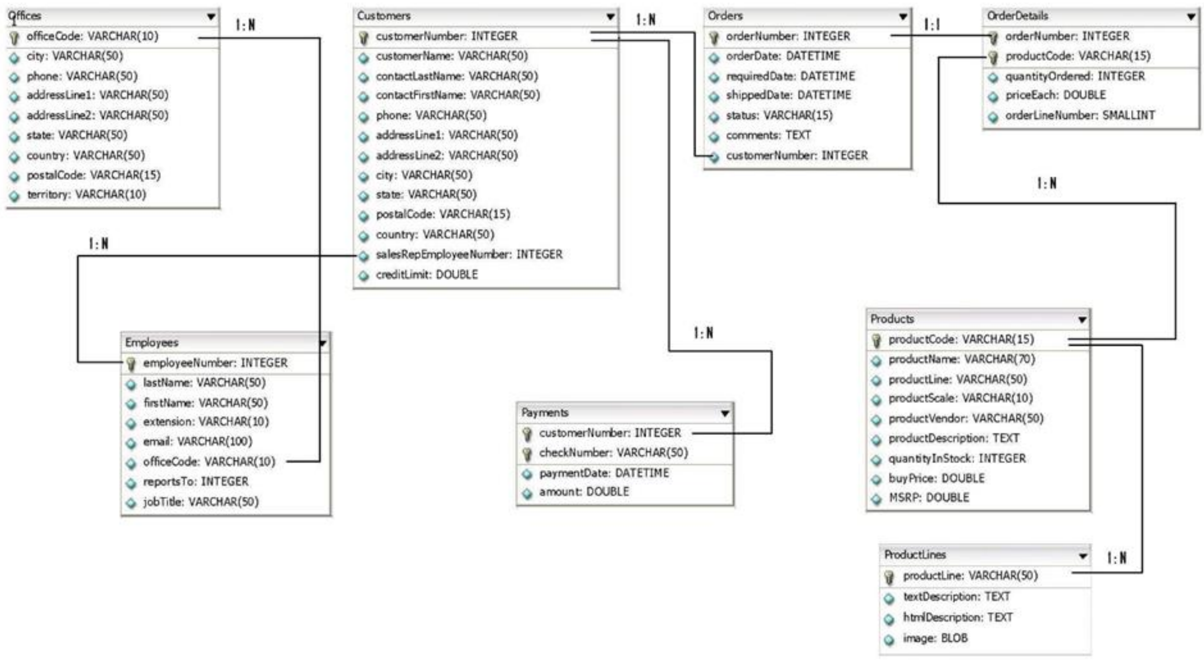

Figure 1 Entity Relationship Diagram of the Classic Models database

In this home exam, the first aim is to create a data warehouse called “classicmodels_dw” using as a data source system: the “classicmodels” database.

The dimensional model for the data warehouse “classicmodels_dw” looks like this:

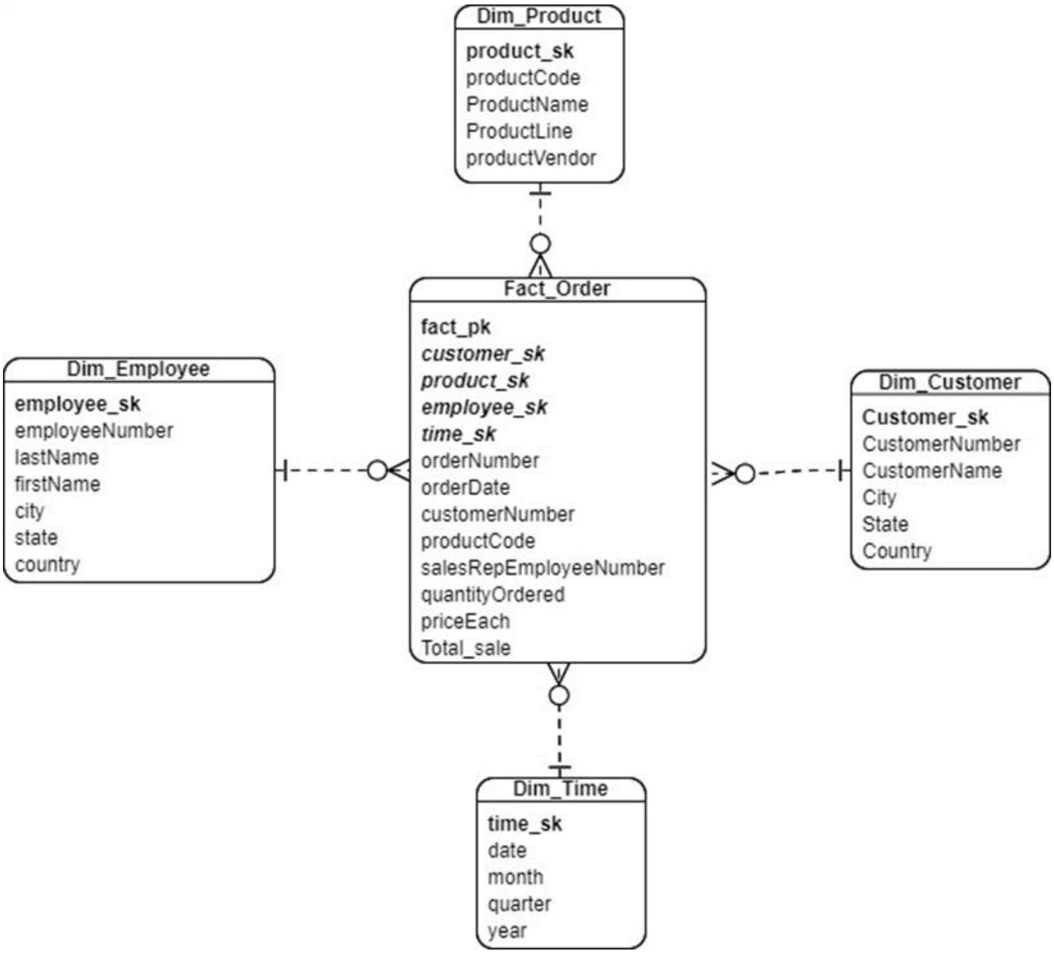

Figure 2 Dimensional Model of the Data Warehouse

The dimensional model has one fact table called Fact_Order and 4 dimensional tables: Dim_Time, Dim_Customer, Dim_Product, and Dim_Employee. The primary keys and surrogate keys are written in bold font in Figure 2.

The field Total_sale in the Fact_Order table can be computed as:

Total_sale = priceEach * quantityOrdered.

Part 1: Building the data warehouse using SQL

The aim of this part is to create the dimension and the fact tables and fill them with data from the source database. These aims will be achieved using SQL and by carrying out the following tasks:

1.1. Create a data warehouse **calssicmodels_dw**. Provide the SQL code for this task.
1.2. Provide the SQL code that create the **Dim_Customer** table that belongs to the data warehouse calssicmodels_dw. The fields of this table are shown in Figure 2.
1.3. Fill the **Dim_Customer** table from the source database. Provide the SQL code for this task.
1.4. Provide a screenshot of: (1) the **Dim_Customer** table filled with data, and (2) the SQL code that display the content of the table **Dim_Customer**. If the result is too long and does not hold in a single screenshot, it is OK to present part of the result (This remark is valid for all the required screenshots in this exam).

1.5. Provide the SQL code that create the **Dim_Employee** table that belongs to the data warehouse calssicmodels_dw. The fields of this table are shown in Figure 2.
1.6. Fill the **Dim_Employee** table from the source database. Provide the SQL code for this task.
1.7. Provide a screenshot of: (1) the **Dim_Employee** table filled with data, and (2) the SQL code that display the content of the table **Dim_Employee**.
1.8. Provide the SQL code that create the **Dim_Product** table that belongs to the data warehouse calssicmodels_dw. The fields of this table are shown in Figure 2.
1.9. Fill the **Dim_Product** table from the source database. Provide the SQL code for this task.
1.10. Provide a screenshot of: (1) the **Dim_Product** table filled with data, and (2) the SQL code that display the content of the table **Dim_Product**.
1.11. Provide the SQL code that create the **Dim_Time** table that belongs to the data warehouse calssicmodels_dw. The fields of this table are shown in Figure 2.
1.12. Create a procedure named "**Fill_timedimension**" that fills the **Dim_Time** table. This procedure takes the start and end date as input parameters. Provide the SQL code for this procedure.
1.13. Provide the value of the start date and the value of the end date by examining the source database.
1.14. Call the procedure **Fill_timedimension** with the right start and end date. Provide the SQL code for this task.
1.15. Provide a screenshot of: (1) the **Dim_Time** table filled with data, and (2) the SQL code that display the content of the table **Dim_Time**.
1.16. Create the **Fact_Stage_Order** table. Provide the SQL code for this task.
1.17. Provide the SQL code used to fill the **Fact_Stage_Order** table from the source database.
1.18. Add the surrogate keys to the staging table **Fact_Stage_Order**. Provide the SQL code for this task.
1.19. Create the **Fact_Order** table that belongs to the data warehouse calssicmodels_dw. Provide the SQL code for this task. The fields of this table are shown in Figure 2.
1.20. Provide the SQL code used to fill the **Fact_Order** table from the **Fact_Stage_Order** table.
1.21. Provide a screenshot of: (1) the **Fact_Order** table filled with data, and (2) the SQL code that display the content of the table **Fact_Order**.

Part 2: Building the data warehouse using Spoon.

The aim of this part is to create the dimension and the fact tables and fill them with data from the source database using Spoon. This objective will be achieved by carrying out the following tasks:

2.1. Create a data warehouse **calssicmodels_spoon_dw** in MySQL. Provide the SQL code for this task.
2.2. Create the **Dim_Time** table and fill it with data by creating a spoon transformation. Name this transformation **2_2.ktr**. The **Dim_Time** table must belong to the data warehouse calssicmodels_spoon_dw.
2.3. Provide screenshots for the spoon transformation in step 2.2. In addition, upload the KTR file **2_2.ktr** on WiseFlow**.**
2.4. Provide a screenshot of the Dim_Time table filled with data.
2.5. Create the **Dim_Customer** table and fill it with data by creating a spoon transformation. Name this transformation **2_5.ktr**. The **Dim_Customer** table must belong to the data warehouse calssicmodels_spoon_dw.
2.6. Provide screenshots for the spoon transformation in step 2.5. In addition, upload the KTR file **2_5.ktr** on WiseFlow**.**
2.7. Provide a screenshot of the **Dim_Customer** filled with data.

2.8. Create the **Dim_Product** table and fill it with data by creating a spoon transformation. Name this transformation **2_8.ktr**. The **Dim_Product** table must belong to the data warehouse calssicmodels_spoon_dw.

2.9. Provide screenshots for the spoon transformation in step 2.8. In addition, upload the KTR file **2_8.ktr** on WiseFlow**.**

2.10. Provide a screenshot of the **Dim_Product** filled with data.

2.11. Create the **Dim_Employee** table and fill it with data by creating a spoon transformation. Name this transformation **2_11.ktr**. The **Dim_Employee** table must belong to the data warehouse calssicmodels_spoon_dw.

2.12. Provide screenshots for the spoon transformation in step 2.11. In addition, upload the KTR file **2_11.ktr** on WiseFlow**.**

2.13. Provide a screenshot of the **Dim_Employee** filled with data.

2.14. Create the **Fact_Stage_Order** table and fill it with data by creating a spoon transformation. Name this transformation **2_14.ktr**.

2.15. Provide screenshots for the spoon transformation in step 2.14. In addition, upload the KTR file **2_14.ktr** on WiseFlow**.**

2.16. Provide a screenshot of the **Fact_Stage_Order** filled with data.

2.17. Create the **Fact_Order** table and fill it with data using the **Fact_Stage_Order** table. Do this task by creating a spoon transformation.

2.18. In the same spoon transformation in step 2.17, add the surrogate keys to the **Fact_Order** table.

2.19. Provide the KTR file for the transformation in steps 2.17 and 2.18. Name this KTR file as **2_19.ktr.** In addition, upload the KTR file **2_19.ktr** on WiseFlow**.**

2.20. Provide a screenshot of the Fact_Order filled with data.

Part 3. Power BI

3.1. The file **Dashboard_0.pbix** contains already some loaded data but does not contain any visuals. Your task is to create 10 different visuals using the data in **Dashboard_0.pbix**. To ensure the clarity of the visuals, the final dashboard should have 5 pages. Each page should contain 2 visuals. So, in total you will create 10 different visuals.

3.2. After creating the 10 visuals save the document as **Dashboard_final.pbix** and upload it on WiseFlow.

Save the report containing your answers as a pdf file for delivery in WiseFlow. Remember to upload the KTR files and the Power BI file as attachments in WiseFlow.