School of Computing
ST1501 Data Engineering CA2
AY2023/2024 Semester 2

A. Instructions and Guidelines

1. You are to work in a group of 3 (recommended) or 4 members. The database to use is **MS SQL Server**. Penalty will be accorded if other databases are used.

2. This is a **group** assignment (55%) **with individual** component (45%), which requires students to design and set up a data warehouse. The assignment will require students to demonstrate competency in designing data warehouse and integrate data from various sources.

3. Submissions should be made via the ST1501 CA2 Assignment Submission links by the stated deadline.

4. Deliverables include:
   - An zip file for the group task with the file-naming convention:
     "**ST1501-YourClass-GroupName.zip**"
     Example ST1501-DITFT2B01-G1.zip

     The group task should consist of the following.
     - A submission template for group consisting:
       - A Data Warehouse Schema
       - An explanation for each query
     - All the SQL scripts to setup the data warehouse
     - All the SQL scripts for queries that can be supported by the data warehouse and their corresponding query results
     - An acdemic integrity declaration for the group

   - An zip file for the individual task with the file-naming convention:
     "**ST1501-YourClass-YourStudentID-YourName.zip**"
     Example ST1501-DITFT2B01-9999999-Benson Ang.zip

     The individual task should consist of the following.
     - A submission template for individual consisting:
       - List the SQL statements to create the documents for each collection
       - All the commands to execute the queries from the MongoDB and the results
       - An indiviual academic integrity declaration

5. As part of the assignment requirements, the team will require to present the assignment. The presentation should explain the data warehouse design and demonstrate the process of setup the OLTP database and the data warehouse. Each team member is required to demonstrate his ability to explain the DW and the questions fielded by the tutor during the presentation.

6. This assignment will account for **40%** of the module grade.

7. No marks will be awarded, if the work is copied or you have allowed others to copy your work.

8. 50% of the marks will be deducted for assignments that are received within ONE (1) calendar day after the submission deadline. No marks will be given thereafter. Exceptions to this policy will be given to students with valid LOA on medical or compassionate grounds. Students in such cases will need to inform the module tutor as soon as reasonably possible. Students are not to assume on their own that their deadline has been extended.

9. Warning: Plagiarism means passing off as one's own the ideas, works, writings, etc., which belong to another person. In accordance with this definition, you are committing plagiarism if you copy the work of another person and turning it in as your own, even if you would have the permission of that person. Plagiarism is a serious offence, and if you are found to have committed, aided, and/or abetted the offence of plagiarism, disciplinary action will be taken against you. If you are guilty of plagiarism, you may fail all modules in the semester, or even be liable for expulsion.

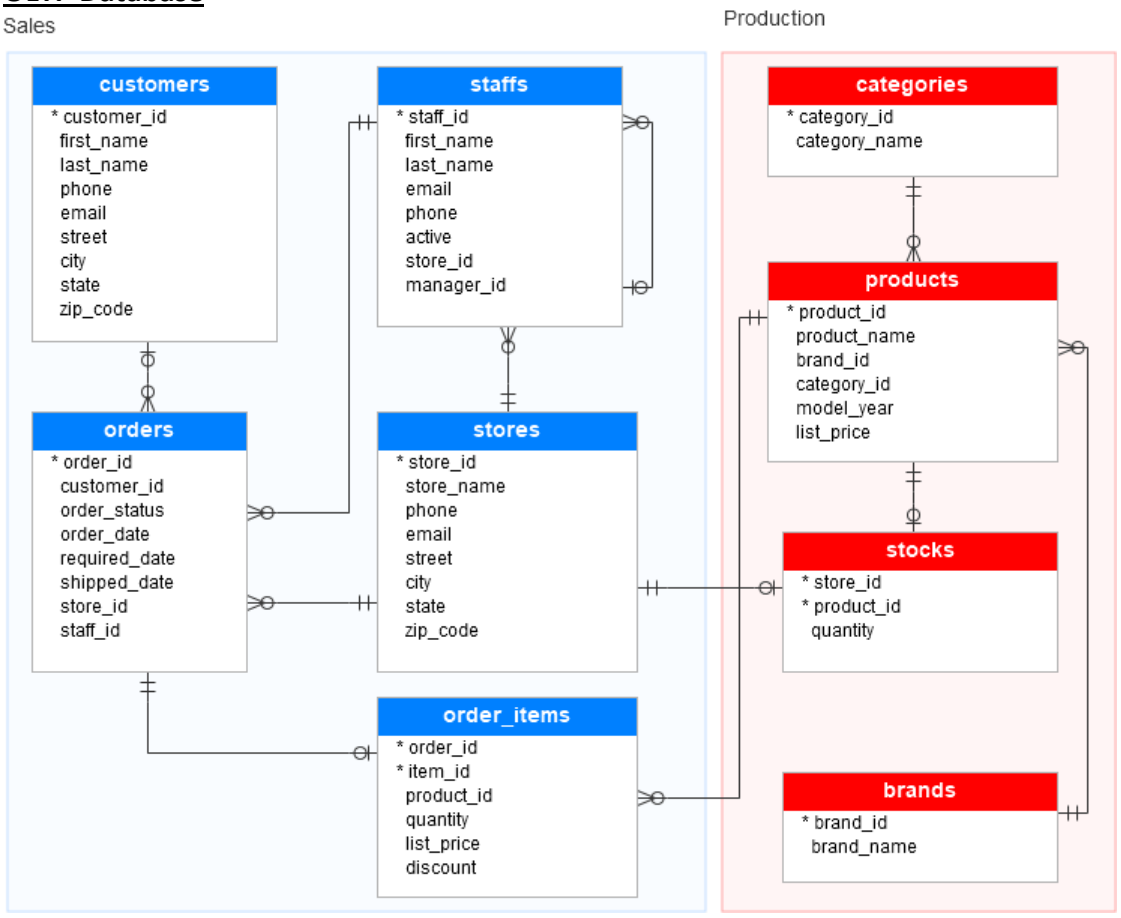
B. The Business Scenario

Trek Bicycle Store Inc. is a retailer of bicycles specializing in various types of bicycles. Currently, they are using an OLTP system in their day-to-day retail business operations. Trek Bicycle Store has a number of store outlets to serve the large community across various states in America. Each store has its unique name, however the outlets are owned by Trek Bicycle Store Inc.

Recently, the owner of the company Fabiola Jackson wanted to incorporate a Data Warehouse solution in this existing IT infrastructure after listening to a talk on business analytics.

The owner wants to have business queries on production as well as sales to be answered from Data Warehouse. He also wishes to keep as much details of time attributes so that he knows the peak and low seasons. Inventory control is also top on his mind so that he knows when to replenish bicycles from suppliers should the stock level runs low or out of stock. Fabiola currently has 3 managers reporting to him across the 3 stores.

OLTP Database



As you can see from the diagram, the database has two schemas sales and production, and these schemas have nine tables. A schema is term to describe a container to hold the various tables serving different functions the retailing business.

Database Tables

Table stores	This table includes the store's information. Each store has a store name, contact information such as phone and email, and an address including street, city, state, and zip code.

Table staffs	This table stores the essential information of staffs including first name, last name. It also contains the communication information such as email and phone. A staff works at a store specified by the value in the store_id column. A store can have one or more staffs. A staff reports to a store manager specified by the value in the manager_id column. If the value in the manager_id is null, then the staff is the top manager. If a staff no longer works for any stores, the value in the active column is set to zero.
Table categories	This table stores the bike's categories such as children bicycles, comfort bicycles, and electric bikes.
Table brands	This table stores the brand's information of bikes, for example, Electra, Haro, and Heller
Table products	This table stores the product's information such as name, brand, category, model year, and list price. Each product belongs to a brand specified by the brand_id column. Hence, a brand may have zero or many products. Each product also belongs to a category specified by the category_id column. Also, each category may have zero or many products.
Table customers	This table stores customer's information including first name, last name, phone, email, street, city, state and zip code.
Table orders	This table stores the sales order's header information including customer, order status, order date, required date, shipped date. Order status: 1 = Pending; 2 = Processing; 3 = Rejected; 4 = Completed It also stores the information on where the sales transaction created (store) and who created it (staff). Each sales order has a row in the sales_orders table. A sales order has one or many line items stored in the sales.order_items table. The owner would wish to track the status of the orders in order to assess their efficiency in delivery with respect to the shipping date, required date and order date. As the status is fast changing attribute and transient, we do not track its history and let the most recent value overrides the previous one in DW.

Table order_items	This table stores the line items of a sales order. Each line item belongs to a sales order specified by the order_id column. A sales order line item includes product, order quantity, list price and discount.
Table stocks	This table stores the inventory information i.e. the quantity of a particular product in a specific store. The quantity in stock is read on the date of the latest sales order for that particular product regardless of stores.

C. Your Task

As a data specialist team, your team is asked to design and setup 2 databases: the OLTP database, the data warehouse using MS SQL Server.

Group Task

(a) Design a star or snowflake schema for a data warehouse that will help answer various business questions regarding sales based on the scenario described in Section B.

(b) Setup the OLTP database (create the tables and insert the data) in the **Microsoft SQL Server** based on the OLTP database design given in Section B.

- The database should be named as **BikeSalesXXXX** where XXXX is to be replaced by the team name.
- All SQL scripts for loading data to the OLTP database are to be submitted.
- The data to be loaded to the OLTP are given as follows:
  - The data from the tables in the OLTP database have come from various sources. Some data are in CSV format, TEXT format and JSON format.
  - The **Customers data**, **Orders data**, **Order items data** and **Stocks** have been prepared in **CSV format.**
  - The **Products data** are in **JSON format**.
  - The **stores, staff, brands and categories data** are given in **textual format.**

(c) Create the data warehouse after your tutor has given feedback to your data warehouse design.

Create the database in the **Microsoft SQL Server** based on your data warehouse design.

- The database should be named as **BikeSalesDWXXXX** where XXXX is to be replaced by the team name.
- Implement surrogate keys for all dimensions **including the Time dimension** and fact tables.
- The SQL script to 'CREATE Table' is to be submitted

(d) Load data from the source system (the OLTP database).

- All the SQL scripts for loading the data to the warehouse are to be submitted.

(e) Besides the sales , the owner wants to have knowledge of the total stock quantity of bikes in the DW. As it is a dynamic value and changes from day to day depending on their orders. As an example, for product_id CHB21, Electra Cruiser 1 (24-Inch) – 2016, we have the following distribution across the 3 stores:

store_id	product_id	quantity
ST1	CHB21	24
ST2	CHB21	16
ST3	CHB21	8
Total quantity		48

The combined stock quantity across all stores is 48 based on the most recent sale of this product on 2018-01-09. The owner wants to keep track of such stock levels. However, there are products that have never been sold in any orders. For such product stocks, the date for stock take is assumed to be the current date of loading data in the DW.

Some products are not even found in the stocks and this have to reflected as 0 for total quantity and date for stock take is assumed to be the current date of loading data in the DW.

The owner does not wish to track the history of inventory changes but he needs to know the current stock of his products at point in time. The DW must be able to reflect the stock inventory.

You can ignore Slow and Fast Changing Dimensions in your DW.

(f) Provide meaningful queries that can be supported by your data warehouse.
The SQL script, the corresponding business question and it query result are to be submitted.

- Query1 – Demonstrate insights on sales/profits/discounts/revenue vary with time
- Query 2 – Demonstrate insights on sales across staff and stores
- Query 3 - Demonstate insights in regard to sales and inventories
- Query 4 – Demostrate insights on sales across customers and their orders
- Query 5 – Demosnrtate insights on sales across products and categories

The query should provide insightful findings to the owner. There may be several insgights drawn from 1 query statement. **Your group has to explaian what the inisghts are in the submission template.**

Remember, there is no right and wrong answers, only insightful or trivial observations. That will depend on how much you want to explore and your examination of the data. Stating the obvious will not eligible for high grade. Your tutor can tell whether enough thoughts have been put into the queries presented.

Individual Task

In the following, you shall experience how to store and manipulate data in NoSQL databases such as MongoDB. You are to show all commands and results in using **MongoDB command-line only (no other lanagugaes in-built into Mango**) for each item listed below:

(a) Mongo database creation
Taking reference from OLTP, create a Mongo database called Bikes with 3 collections:

- UnSold – create a collection to store those products which have never been sold
- ZeroStock – create a collection to store products which have zero stock
- Stock – create a collection to store products which have stock
  - For UnSold and ZeroStock collections, the documents must have the fields product_id, product_name, brand_name, category_name, model_year and list_price
  - For the Stock collection, the documents must have fields product_id, store_id and quantity
  - Find the count of documents in each collection

(b) Queries
Use MongoDB commands to demonstrate the following:

- Query 1 - Implement a query to find out those bike names which are less than $2000 and belonging to Road Bikes and have zero stock
- Query 2 - Implement a query to find out the unduplicated category names and unduplicated brand names separately for these unsold products

Hint: Use MongoDB Aggregate pipeline & $Lookup

- Query 3 - Implement a query to find the unduplicated sets of category names and brand names in combination
- Query 4 - Implement a query to find the information of those products which are unsold and have stock
- Query 5 – Making use of the product_id from Query 4, implement a query to find their total stock available arranged in descending order
- Query 6 – Modify Query 4 and implement a query to find the information of those products which are sold but do not have stock

-- End of Assignment --

The Assessment Criteria

Component	Tasks	Weightage	
Group Work 55%	**Data Warehouse Design (Database Diagram)** • The design supports the described business scenario. • The chosen table names, field names and attributes are descriptive. • The explanation of the design is clear and concise.	5%	
	OLTP Database Setup • The Create Table SQL script implements the database design, including the primary key and foreign key definition. • The Insert Records SQL script to load the given data. • The SQL Script to load the Product and Customer records • Explanation and the demonstration of OLTP setup is without error during the presentation.	5%	
	The database warehouse setup in MS SQL • The Create Table SQL script implements the star or snowflake design, including the primary key, surrogate key and foreign key definition. • The SQL Script to load the data into the data warehouse	15%	
	The queries in MS SQL • Five insightful query that covers each of the category (Sales, Staff/Office, Seasons of Sales, Orders/Customer, Products/brands/categories/inventory)	6% 6% 6% 6% 6%	30%
Individual Work 45%	**Setup MongoDB** • The commands to create the database • The commands to create the collections • The command to execute the viewing of the collection documents	10%	
	The queries in the MongoDB • The queries as implemented • The commands to execute • The results from MongoDB	3% 3% 3% 6% 7% 3%	25%
	Question and answer during the presentation	10%	
	Total	100%	

Appendix 1

Table creation

```
CREATE TABLE sales.stores (
 store_id varchar(5) PRIMARY KEY,
 store_name VARCHAR (255) NOT NULL,
 phone VARCHAR (25),
 email VARCHAR (255),
 street VARCHAR (255),
 city VARCHAR (255),
 state VARCHAR (10),
 zip_code VARCHAR (5)
);

CREATE TABLE sales.staffs (
 staff_id varchar(5) PRIMARY KEY,
 first_name VARCHAR (50) NOT NULL,
 last_name VARCHAR (50) NOT NULL,
 email VARCHAR (255) NOT NULL UNIQUE,
 phone VARCHAR (25),
 active int NOT NULL,
 store_id varchar(5) NOT NULL,
 manager_id varchar(5),
 FOREIGN KEY (store_id)  REFERENCES sales.stores (store_id)
 ON DELETE CASCADE
 ON UPDATE CASCADE,
 FOREIGN KEY (manager_id)  REFERENCES sales.staffs (staff_id)
 ON DELETE NO ACTION
 ON UPDATE NO ACTION
);

CREATE TABLE production.categories (
 category_id varchar(5) PRIMARY KEY,
 category_name VARCHAR (255) NOT NULL
);
CREATE TABLE production.brands (
 brand_id varchar(5) PRIMARY KEY,
 brand_name VARCHAR (255) NOT NULL
);

CREATE TABLE production.products (
 product_id varchar(10) PRIMARY KEY,
 product_name VARCHAR (255) NOT NULL,
 brand_id varchar(5) NOT NULL,
 category_id varchar(5) NOT NULL,
 model_year int NOT NULL,
```

```
 list_price DECIMAL (10, 2) NOT NULL,
 FOREIGN KEY (category_id) REFERENCES production.categories (category_id)
 ON DELETE CASCADE ON UPDATE CASCADE,
 FOREIGN KEY (brand_id) REFERENCES production.brands (brand_id)
 ON DELETE CASCADE ON UPDATE CASCADE
);
CREATE TABLE sales.customers (
 customer_id varchar(10) PRIMARY KEY,
 first_name VARCHAR (255) NOT NULL,
 last_name VARCHAR (255) NOT NULL,
 phone VARCHAR (25),
 email VARCHAR (255) NOT NULL,
 street VARCHAR (255),
 city VARCHAR (50),
 state VARCHAR (25),
 zip_code VARCHAR (5)
);
CREATE TABLE sales.orders (
 order_id varchar(10)  PRIMARY KEY,
 customer_id varchar(10),
 order_status int NOT NULL,
 -- Order status: 1 = Pending; 2 = Processing; 3 = Rejected; 4 = Completed
 order_date DATE NOT NULL,
 required_date DATE NOT NULL,
 shipped_date DATE,
 store_id varchar(5) NOT NULL,
 staff_id varchar(5) NOT NULL,
 FOREIGN KEY (customer_id) REFERENCES sales.customers (customer_id)
 ON DELETE CASCADE ON UPDATE CASCADE,
 FOREIGN KEY (store_id) REFERENCES sales.stores (store_id)
 ON DELETE CASCADE ON UPDATE CASCADE,
 FOREIGN KEY (staff_id) REFERENCES sales.staffs (staff_id)
 ON DELETE NO ACTION
 ON UPDATE NO ACTION
);
CREATE TABLE sales.order_items(
 order_id varchar(10),
 item_id INT,
 product_id varchar(10) NOT NULL,
 quantity INT NOT NULL,
 list_price DECIMAL (10, 2) NOT NULL,
 discount DECIMAL (4, 2) NOT NULL DEFAULT 0,
 PRIMARY KEY (order_id, item_id),
 FOREIGN KEY (order_id)  REFERENCES sales.orders (order_id)
 ON DELETE CASCADE ON UPDATE CASCADE,
 FOREIGN KEY (product_id) REFERENCES production.products (product_id)
```

```
 ON DELETE CASCADE
 ON UPDATE CASCADE
);
CREATE TABLE production.stocks (
 store_id varchar(5),
 product_id varchar(10),
 quantity INT,
 PRIMARY KEY (store_id, product_id),
 FOREIGN KEY (store_id)  REFERENCES sales.stores (store_id)
 ON DELETE CASCADE ON UPDATE CASCADE,
 FOREIGN KEY (product_id) REFERENCES production.products (product_id)
 ON DELETE CASCADE
 ON UPDATE CASCADE
  );
```

Appendix 2

Data

Staff

staff_id	first_name	last_name	email	phone	active	store_id	manager_id
3031	Fabiola	Jackson	fabiola.jackson@bikes.shop	(831) 555-5554	1	ST1	NULL
30310	Bernardine	Houston	bernardine.houston@bikes.shop	(972) 530-5557	1	ST3	3037
3032	Mireya	Copeland	mireya.copeland@bikes.shop	(831) 555-5555	1	ST1	3031
3033	Genna	Serrano	genna.serrano@bikes.shop	(831) 555-5556	1	ST1	3032
3034	Virgie	Wiggins	virgie.wiggins@bikes.shop	(831) 555-5557	1	ST1	3032
3035	Jannette	David	jannette.david@bikes.shop	(516) 379-4444	1	ST2	3031
3036	Marcelene	Boyer	marcelene.boyer@bikes.shop	(516) 379-4445	1	ST2	3035
3037	Venita	Daniel	venita.daniel@bikes.shop	(516) 379-4446	1	ST2	3035
3038	Kali	Vargas	kali.vargas@bikes.shop	(972) 530-5555	1	ST3	3031
3039	Layla	Terrell	layla.terrell@bikes.shop	(972) 530-5556	1	ST3	3037

Brand

brand_id	brand_name
BRD1	Electra
BRD2	Haro
BRD3	Heller
BRD4	Pure Cycles
BRD5	Ritchey
BRD6	Strider
BRD7	Sun Bicycles
BRD8	Surly
BRD9	Trek

Category

category_id	category_name
CHB1	Children Bicycles
CMB2	Comfort Bicycles
CRB3	Cruisers Bicycles
CYB4	Cyclocross Bicycles
ELB5	Electric Bikes
MOB6	Mountain Bikes
RDB7	Road Bikes

Store

store_id	store_name	phone	email	street	city	state	zip_code
ST1	Santa Cruz Bikes	(831) 476-4321	santacruz@bikes.shop	3700 Portola Drive	Santa Cruz	CA	95060
ST2	Baldwin Bikes	(516) 379-8888	baldwin@bikes.shop	4200 Chestnut Lane	Baldwin	NY	11432
ST3	Rowlett Bikes	(972) 530-5555	rowlett@bikes.shop	8000 Fairway Avenue	Rowlett	TX	75088