CSE4110 - Database System

Project2. Normalization and Query Processing

Spring 2020

Package Delivery System

"You are a DBA in this company"

Goal : The goal of this project is to provide a realistic experience in the physical design, query processing implementation and maintenance of a small relational database you made in Project 1.

Application Description(Same as Project 1) :

The application is a package delivery company (e.g. FedEx, UPS, DHL, CJ Logistics). The company needs to keep track of packages shipped and their customers. To find out more about this application, think about any experiences you may have had shipping packages and receiving packages, and browse shippers' web sites.

In our hypothetical company, the manager assigned to solicit database design proposals is not very computer literate and is unable to provide a very detailed specification.

Here are a few points to consider :

- There are different kinds of service possibly based upon the type of package(flat envelope, small box, larger boxes), the weight of the package, and the timeliness of delivery (overnight, second day, or longer).

- Some customers have a contract with the shipper and bill their shipments to an account number. They are billed monthly. Other customers are infrequent customers and pay with a credit card. Certain shipments are prepaid, as is might be the case of someone is returning something that was purchased by phone or Internet (e.g. returning clothes that don't fit, or returning malfunctioning electronics, returning damaged book).

- For the most part, the shipping company does not care what is being shipped. However, there are cases where it matters. Some examples include

- hazardous materials
- international shipments, for which a customs declaration stating the contents and their value is needed

- The company needs to track packages from the time the customer drops it off (or it is picked up by the company) until the time is is delivered and signed for. Take a look at the online tracking offered by various shipping companies to get an idea of how this service works. If you are having something shipped to you, you'll find you can get every little detail of where the package is, where it has been, and to where it is currently headed. Beyond what the customer sees, the company itself needs to know on which truck or plane or warehouse the package is at any point in time.

- Tracking is not just an "in the present" issue. The company may want to look back in time and find out where the package was yesterday, for example. It may also want to look at data from the standpoint of a truck or warehouse.

- There are other aspects to the operation of the company besides package tracking such as the routing of trucks and planes, the assignment of staff to them, etc. For this project, we'll consider only the package handling and billing aspects of the database.

Project Requirements :

1. BCNF Decomposition
   - Decompose your Logical Schema to BCNF
   - Be sure to every entities and relationships satisfies BCNF dependency
   - Same process learned in lecture (Chap 7.5.1 in textbook)

2. Physical Schema Diagram
   - After creating an decomposed logical schema diagram, also creating Physical Schema diagram

- Create the “physical” schema diagram in ERwin Data Modeler
- Be sure to identify data types, domain, constraints, relationship type, allowing nulls and so far.

3. Queries

The queries listed below are those that your client (the manager from the package delivery company) wants turned in. We now called these query category as TYPE. (total 7 different query types)

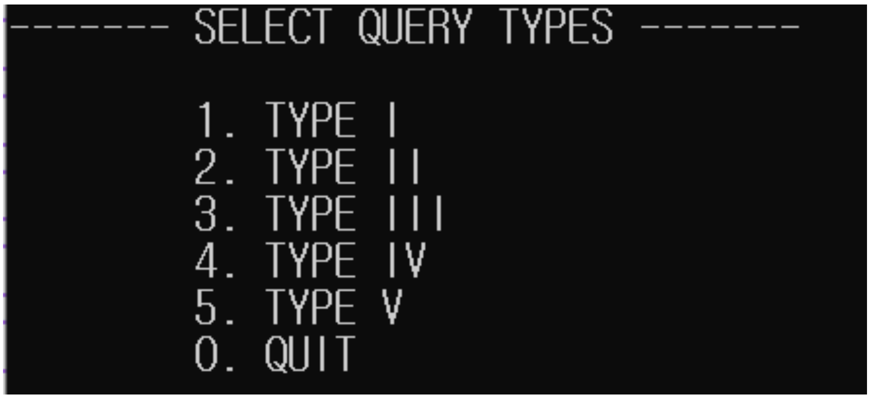
- **(TYPE I)** Assume truck 1721 is destroyed in a crash.
  - **(TYPE I-1)** Find all customers who had a package on the truck at the time of the crash. (1)
  - **(TYPE I-2)** Find all recipients who had a package on that truck at the time of the crash. (2)
  - **(TYPE I-3)** Find the last successful delivery by that truck prior to the crash. (3)
- **(TYPE II)** Find the customer who has shipped the most packages in the past certain year. (4)
- **(TYPE III)** Find the customer who has spent the most money on shipping in the past certain year. (5)
- **(TYPE IV)** Find those packages that were not delivered within the promised time. (6)
- **(TYPE V)** Generate the bill for each customer for the past certain month. Consider creating several types of bills. (7)
  - A simple bill: customer, address, and amount owed.
  - A bill listing charges by type of service.
  - An itemize billing listing each individual shipment and the charges for it.

- We should execute these queries within MySQL DBMS what you managed. The customers can check all results from these query execution.

- We use ODBC with C language API to implement your Database model (will be discussed in practice session)

4. Code implementation

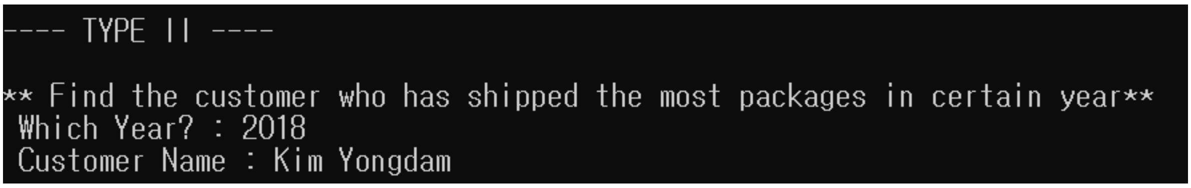
- We will use Visual Studio 2019 and MySQL connector(ODBC)
- Implement C code to execute your queries in MySQL DBMS you built
- The program must use file input from text file which have CRUD queries as text format
  e.g. CREATE, INSERT, UPDATE, DELETE
- The program has a user interface to handle each TYPE. Program runs until put quit instruction.



```
------- SELECT QUERY TYPES -------

        1. TYPE I
        2. TYPE II
        3. TYPE III
        4. TYPE IV
        5. TYPE V
        0. QUIT
```

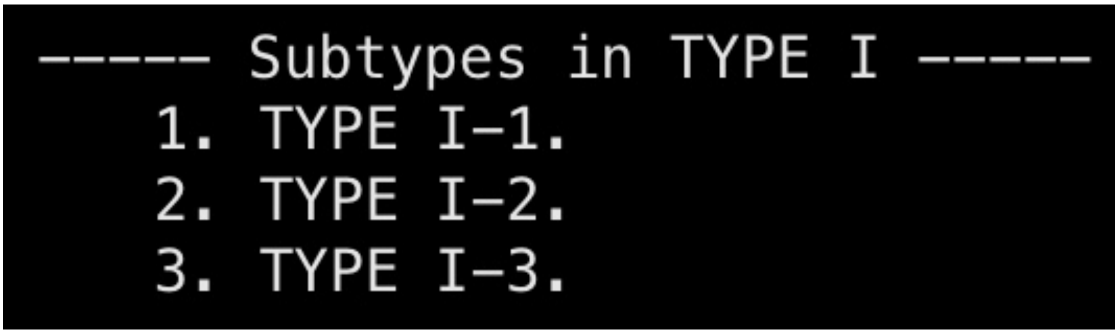
- After select one of type in menu, user put values of query as stdin and print result of query execution as stdout.



```
---- TYPE II ----

** Find the customer who has shipped the most packages in certain year**
 Which Year? : 2018
 Customer Name : Kim Yongdam
```

- Only TYPE I has submenu for subtypes.



```
----- Subtypes in TYPE I -----
    1. TYPE I-1.
    2. TYPE I-2.
    3. TYPE I-3.
```

- Keep running each type of query until put “0” in stdin. (for all types of query)
- After exit from query menu, back to select menu.

What to turn in :

- Decomposed Logical Schema diagram (.png)
  - student_id.png (submitted filename) **e.g. 20101621.png**

- “Physical” Scheme diagram ERwin file (.erwin)
  - student_id.erwin (submitted filename) **e.g. 20101621.erwin**

- C code (.c or .cpp)
  - student_id.c (submitted filename) **e.g. 20101621.c**
  - Describe your code with comment (if necessary)
  - It should be able to build and execute in Visual Studio 2019 with MySQL C API.

- Report file (.pdf)
  - student_id.pdf (submitted filename) **e.g. 20101621.pdf**
  - Describe the detail explanation about your physical Schema diagram and ODBC implementation within MySQL that you made.
  - MAKE YOUR OWN DESCRIPTION on physical schema and ODBC C language codes.
  - Describe the detail explanation how you decompose each relation to BCNF, see testing one of your relation by simplified test in lecture notes.
  - Feel free to use any template you made.

NOTICE :

- **2020.06.29(Mon) 23:59**
- Submit your soft copy with title **“[DBproject2]student_id”** to **sgdbsystem2020@gmail.com** **(softcopy includes png, erwin, c code and pdf file you wrote)**

- **BEFORE YOU TURN IN, PLEASE CHECK NOTICE BOARD TO FIGURE OUT HOW YOU ZIP YOUR SUBMITTED FILES.**

- **DON’T COPY ANYTHING FROM YOUR FRIENDS AND WEB SOURCES. IF YOU VIOLATE THIS, YOU WILL GET “F” FOR THIS COURSE.**