COEN 178 Intro To. Database Systems Winter 2019

Assignment-2 (Due 18th Feb, 10 PM) Individual **Total (100 pts)**

Smart Meal Plans is an on-line store that takes customer orders for meal plans and delivers the required ingredients for each meal.

It maintains a database of **MealItems**, **MealOrders and Order_Items.**

Using the **relational schemas** given below, create tables using Oracle database and implement queries in Part 2, in **SQL.**

Part 1 (creating tables and inserting values): 20 pts

MealItem (**itemId** *(string)*, *name, price (floating point number with 2 digits of precision), calories (integer).*

MealOrder (**orderId (***string***)**, **name, phone**): The primary keys of each table are underlined.

Order_Item (**orderId** (string), **itemid** (string)) – **orderid** is a foreign key into MealOrder table and **itemid** is a foreign key into MealItem).

Create tables using schemas given above, clearly defining the **primary** and **foreign key** constraints.

Insert some sample data (some values are given below) into the tables:

MealItem (**itemId** (char string), name, price, calories:

```
 'I1', 'oatmeal',3.00,120
'I2', 'fruit_plate',7.50,220
'I3', 'steak',20.99,420
'I4', 'chicken pie',12.50,350
'I5', 'broccoli pie',10.00,200
```

MealOrder (***ordertId***, *name, phone*)

```
'O1','Smith','4085551212'
'O2','Jones','4085554444'
'O5','Clark','4083331212'
'O7','Chen','4086661212'
```

```
'O8','Smith','4085551212'
```

Order_Item (**orderId**, **itemId**, orderedDate)

```
'O1','I1'
'O1','I2'
'O1','I3'
'O2','I4'
'O8','I1'
```

Insert more tuples of your choice into each of the tables to test your queries in Part 2.

Part 2: 80 pts

Write the following queries using SQL.

1) Show the names of MealItems and prices, sorted by price.
2) Show the names of MealItems, price and calories, sorted by price and calories.
3) Show the name(s) of the MealItem and calories with most no. of calories.
4) Show the name(s) of the item that is most popular (most no. of occurences) in the Order_Item table.
5) Show the names and prices of the items that are not in Order_Item table.
6) Show the phone of the person with most orders in the Orders table. An incomplete query is given below. Complete (you are free to modify it) it.

```
Select phone,count(*)
from MealOrder
Group by Phone
```

7) Show the orderId and total calories for all the items in that order (from Order_Item table)
8) Change the price of those items (in MealItem table) to 10% less than the current price, if that item is not in any of the orders (Order_Item table).

What to submit:

- Schemas of all the tables you have created. Include the "Create table statements" in a script file called **a2_tables_*yourfirstinitalLastName*.sql**.
- Data in the tables (the results of *Select * from table name*). Include the "insert into table" statements into a script file called **a2_values_*yourfirstinitalLastName*.sql.**

  The SQL queries for the queries described in Part 2. Include the queries in a script file called **a2_queries_*yourfirstinitalLastName*.sql.**
- **Spool your results into a file called a2_results_*yourfirstinitalLastName*.txt.**

Some tutorial links:

http://www.tutorialspoint.com/sql/