Lesson Plan 03, ISTA-420

Chapter 2, T-SQL Fundamentals

June 5, 2017

1 Class Discussion

1. What is a *primary key constraint*? What two other constraints is it equivilent to?
2. What is a *nullability constraint*? What does it prevent?
3. What is a *unique constraint*? What does it prevent?
4. What is a *foreign key constraint*? What does it allow?
5. What is a *check constraint*? What does it allow?
6. What is a *default constraint*? What does it allow?
7. What is *domain integrity*? This is not in your text book, but it's important.
8. What is the difference between the *where* and the *having* clauses? How are they alile?
9. What SQL operator has the highest precedence? What SQL operator has the lowest precedence?
10. Yes or no: In the SQL standard, is *NULL* equal to *NULL*? Why or why not?

2 In Class Exercises

Using SQLite and the Northwind database, write a SQL script that executes the following queries. Your deliverables should be your SQL script and the text output.

1. List the countries with most customers from high to low?
2. What is the average price of products for each product category?
3. What product category has the lowest number of units in stock?
4. What postal codes experienced a significant delay in shipping on average?
5. Do we make more money selling a few high priced items or many lot priced items?
6. I need a list of our customers and the first name only of the customer representative.
7. I want to see when customers placed orders in December. Give me a data file showing the day of all December orders.

```sql
select country, count(country) as count from customers group by country order by count desc;

select categoryID, avg(UnitPrice) as averagePrice from products group by categoryID order by averagePrice desc;

select categoryID, sum(UnitsInStock) as inStock from products group by categoryID order by inStock;

select shipPostalCode, 'average␣lag␣time', avg(OrderDate - ShippedDate) as lagTime from orders group by shipPostalCode having lagTime > 7;

select productID, unitprice, count(Quantity) as quantity, sum(UnitPrice * quantity) as total from order_details  where unitprice < 10 group by productID having count(quantity) > 40 order by total desc;
select productID, unitprice, count(Quantity) as quantity, sum(UnitPrice * quantity) as total from order_details where unitprice > 150 group by productID having count(quantity) < 100 order by total desc;

select substr(contactname, 1, pos-1) as first_name,
      companyname
from
  (select *,
         instr(contactname,'␣') as pos
   from customers)
order by first_name;

select strftime("%d", orderdate) as december_day from orders where orderdate like '19___12%';
```

3 Graded Labs

Lab 2, exercises 1, 2, and 3, from 20761A, page 2-23. Labs will be directed by assigned students.

4 Course Project

4.1 Software Process

Requirements analysis and requirements engineering, UML and use cases.

4.2 Version Control

We will learn how to add files to a Git repository.

4.3 Project Assignment

Create a basic requirements specification. Write and execute a SQL script that meets your requirements specification. You can use either SQLite or SQL Server, your choice. We will probably be using SQLite in the early stages of the project, but we will move to SQL Server later on. This has two purposes: (1) SQLite is simple and easy for developing prototypes of projects in the early stages, and (2) you need to learn how the SQL dialects understood by different database systems are mutually intelligible, and how they differ.

5 Homework

5.1 Readings

Read chapter 2 of the textbook T-SQL Fundamentals, pages 60 through 93.

5.2 Exercises

- Do exercises 6 through 10 beginning on page 93. You are allowed to look at and copy the solutions, but make sure you understand the concepts. For example, if you give a query similar to exercise 1 but for another month, would you understand how to do it?
- Read the Microsoft T-SQL documentation for the functions used in these exercises.