Lab – Introduction to SQL Select – Part 2

## Overview

In this lab, we will practice using DML (mainly the SQL SELECT statement) and the myriad of options available to it that we learned in class, mainly:

* SQL Functions for manipulating dates, text, and converting data types.
* Column Aliasing.
* Use of the TOP and DISTINCT keywords
* Use of SQL JOINS to combine output from multiple tables
* Use of SQL aggregate functions and the GROUP BY and HAVING clauses

### Learning Objectives

Upon completion of this learning unit you should be able to:

* Compare, contrast scalar and aggregate functions.
* Describe and use various ways to join tables.
* Solve problems using aggregate functions and joins.
* Compare join and sub queries.
* Understand how to use special constructs on the SELECT statement.

### Lab Goals

This lab consists of 3 sections. You completed Sections 1 and 2 for Part 1. That just leaves Section 3 where you do the SQL writing. As a reminder of the entire lab:

1. Section 1 has been repeated here so you have the data model to help you construct your queries.
2. Section 2 is where you were provided a number of great samples.
3. Section 3 is where you will have craft your own SQL statements from my descriptions using our account on Microsoft SQL Server. Style points will be rewarded. If you get stuck on a question look back to your examples in Section 2 for a similar problem.

**You must complete Part 1 Section 1 in order to complete Part 2. An Answer Template is provided at the end. Please insert your answers and upload into BB.**

## Section 1: The Fudgemart Database Schema (Reloaded)

Throughout the semester we will use several different case studies to help enforce the concepts we learn in class. One of the recurring case-studies we will use in class and the labs is the Fudgemart database. This database supports the business operations of a fictitious mega-store retailer and e-tailer called Fudgemart. The Fudgemart database supports all aspects of the business from human resources, to payroll, to sales transactions, and e-commerce. In each lab we will add new database objects and data to the Fudgemart schema.

### 1a: The enhanced Conceptual Model

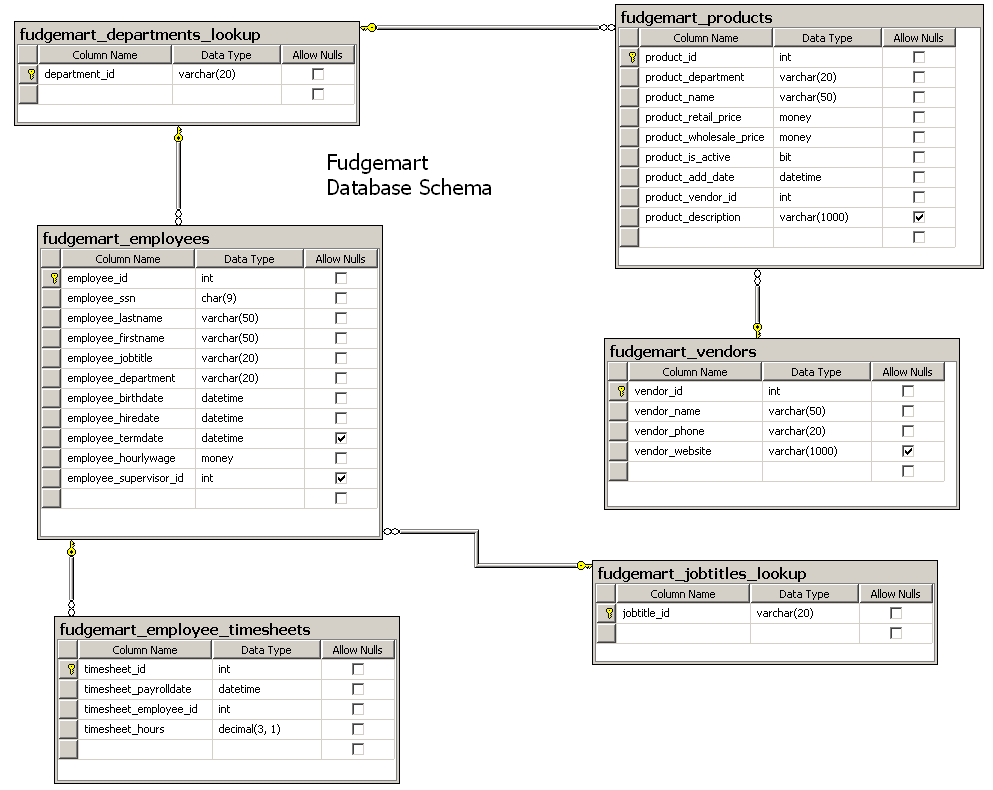
Let’s combine the conceptual models from the first two labs into a new mega Conceptual model. The Conceptual model is represented by a Chen Entity-Relationship Diagram.

As you’ve probably already figured out, the boxes will be tables in our Fudgemart database, and the lines connecting the boxes become foreign key constraints.



### 1b: Fudgemart Internal Model (Schema)

Here is the internal model. Remember there is a 1-1 correspondence between the logical model and the internal model. Also notice how much more “complicated” the internal model is when compared to the conceptual model. Remember, the conceptual model exists to define the *requirements* of the database, the logical model is a *blueprint* of that database, while the internal model *\*is\** the database. The only difference between the logical model and the internal model is the latter is actually implemented in a DBMS. Notice, that although not represented in the diagram, the check constraints, unique constraints, and default values all have all been implemented in the database schema just like they were in the previous labs.



## Section 3: Write the SQL Statement!

In this part you will attempt to **write the SQL statement** which best corresponds to the provided text description. Once working, copy your SQL into the answer sheet that will be uploaded into Blackboard for each of the following 10 queries.

|  |
| --- |
| 3.a) Write an SQL statement to list all products in the ‘Sporting Goods’ department. Sort by product name in ascending order. Your output should look like this: |
| 3.b) Write an SQL statement to list the employees who work in the customer service department and make more than $15/hr. Show their name, department and hourly wage. Your output should look like this: |
| 3.c) top 5 profit margin items. Write an SQL statement to list the product name, department, retail price, wholesale price, and product markup (difference between wholesale and retail) of the TOP 5 product markups. Your output should look like this: |
| 3.d) Write an SQL statement to use the TOP keyword to find the two highest paid employees (highest hourly wage) in fudgemart. Show the employees name (in one column) along with the hourly wage. Your output should look like this: |
| 3.e) Retrieve the 10 highest timesheet hours for December 2006. Show the timesheet\_id, payroll date, employee\_id and number of hours worked. Sample Output: |
| 3.f) Write an SQL query to display the vendor name, the associated product department and product name for the Hardware or Housewares department. Sort by vendor name, department then product name in ascending order. |
| 3.g) For the Mikey Vendor, display the product name, wholesale price, and product add date, sorted by add date: |
| 3.h) Write the same SQL query you wrote in 3.e but include names of employees rather than their employee ID. |
| 3.i) For all employees who have worked more than 40 hours on their timesheet, list the employee name, and their department. Sort the output by department, and only display unique rows of output. |
| 3.j) List the names and phone numbers of all vendors without a website. |

3.k Explain what the Information\_Schema is in SQL Server? What is the purpose? How can you make use of the information?

# Lab: SQL SELECT Statement Part 2 Answer Sheet Place Your Name Here:\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Place your Answers to Each Section in the Corresponding Area Below. Add as much additional space as you need so your answers are easy to read. Only Upload these Pages, not the Entire LaB.

**Answer 3.a-J This is your turn to write the SQL that would create the output as described. Work out your answer using Management Studio and then copy/paste just your sql commands below. You do not need to upload the result set.**

3.a

3.b

3.c

3.d

3.e

3.f

3.g

3.h

3.i

3.j

3.k