



School of Engineering and Computing

Introduction to Database Development

Level: 1

Module Code: M1G505190

Coursework 2

Issue: 30th November 2009

This coursework comprises 70% of the overall mark for the module.

Hand-in date: 11th December 2009

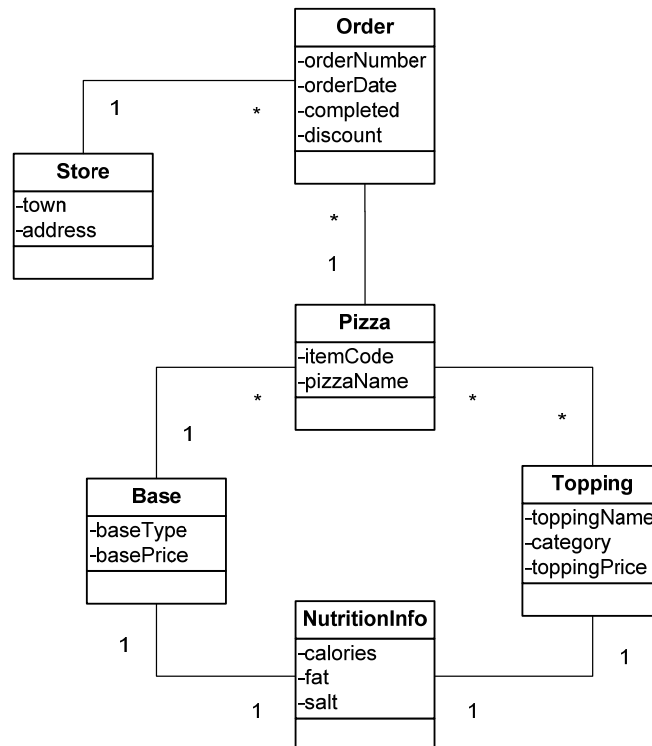
An average student should be able to complete this assignment in about 8 hours of work.

Attention is drawn to the university regulations on plagiarism. Whilst discussion of the coursework between individual students is encouraged, the actual work has to be undertaken individually. Collusion may result in a zero mark being recorded for the coursework for all concerned and may result in further action being taken.



Implementing a database for GCU Pizza

GCU Pizza need a new system to handle pizza orders. Some of the design work has been done, and there is a domain/data model, shown in the UML class diagram below.



NOTE:

- each order can include only one pizza
- pizzas are all the same size
- GCU Pizza does not allow extra toppings – there is a fixed menu of named pizzas, each with a specified list of toppings
- **pizzaName** is the name of the pizza which would appear on a menu, e.g Deep Pan Meat Combo
- **category** is the type of topping, e.g. meat, fish, vegetable
- the price of a pizza depends on the **basePrice** and the sum of **toppingPrice** for all the toppings on that pizza.
- **completed** indicates where the order has been completed (yes or no)
- **discount** is a percentage discount for an order

This coursework consist of **three tasks** related to this data model –attempt **all the tasks**:

- **Task 1:** Create and test a database (40 marks)
- **Task 2:** Queries (20 marks)
- **Task 3:** Form/Reports (10 marks)

Total 70 marks

Preparation and submission

Create a new Access database called **gcupizza_ *matricno*.mdb**, where *matricno* is your matriculation number. For example, if your matriculation number was S0999999 your database would be gcuoizza_S0999999.mdb.

Create a new Word document called **gcupizza_ *matricno*.doc**.

Save your work regularly on your network drive and/or a removable memory device.

Submission information will be given on Blackboard.

Task 1: Create and test the database (40 marks)

- a. **Use SQL** to implement the database schema in Access (20 marks)

Create and **save a query** in your database for each SQL statement you run. For example, you should save the SQL to create the Pizzas table in a query called *Create Pizzas Table*. You should also copy the SQL to your Word document under the heading **Part 1a**.

You will get credit for the following:

- Primary and foreign keys correctly implemented (10 marks)
- Appropriate choice of fields and data types (5 marks)
- Appropriate constraints on fields (5 marks)

You do not need to use SQL to apply constraints other than NOT NULL or keys. However, you should include in your Word document a brief description of each additional constraint you have implemented, for example:

Tablename, constraint on *fieldname*, value must be less than 100

- b. **Test** your database (20 marks)

Check that your database tables are normalized to 3NF and justify your conclusions.

Insert data into your tables. You should have 20 toppings, with 4 different categories, and you should have at least 4 rows in every other table. Choose test data to demonstrate that all relationships are working correctly.

Test each key and constraints by attempting to enter a row which should not be allowed, and record the result for each in your Word document as follows, under the heading **Part 1b**.

- description of key or constraint, for example *Customers primary key*
- test data: field values for an invalid row
- description of result, for example error, *row not inserted*

Task 2: Queries (20 marks)

Create and **save a query** in your database for each of the following. For example, the first query should be called *Query 2a*. You should create your queries in **SQL View**. You should copy the SQL and the results for each query to your Word document under the heading **Part 2**.

***NOTE:** you can copy the results of a query by clicking on the top left corner of the results view to select the whole grid, and selecting Copy from the Edit menu. The results can then be pasted into Word.*

- a. **List all details of all toppings in a specified category in ascending order of toppingName**
- b. **List the distinct categories of the toppings**
- c. **Find the number of orders made between two specified dates**
- d. **List the names of all pizzas with a specified word in the name**
- e. **Increase the price of all bases by 10%**
- f. **List the average fat content in each topping category in ascending order**
- g. **With a single query, delete a specified store and any related orders** (you should make sure the relationship between the tables has the appropriate properties)
- h. **List the name, and total price of all toppings for each pizza**
- i. **For a town specified when the query runs, list the date, pizzalD and discount for all orders placed in that town**
- j. **List the name, category, calories, salt and fat for the topping (or toppings) with the highest fat content**

Task 3: Forms/Reports (10 marks)

Create the following **form and report** in your database. To gain full credit these should be attractively formatted and should display information in the most appropriate way.

- a. **a master-detail form to display information about stores and orders. The subform should list orders (showing *orderDate*, *pizzalD*, *discount* and *completed*) while the main form should list store information.**
- b. **A report listing toppings, grouped by category, showing average calories, fat and salt for each category and also the overall averages for calories, fat and salt**