

CS2 Databases TEST 2
3 September 2003
Solutions

Section A – Multiple Choice [10 marks]

- 1 B Field
- 2 The correct answer is **B**. You need to include the INTO clause prior to the FROM clause. The correct statement would be SELECT * INTO NewCustomers FROM Customers WHERE CustomerID = 12345.
- 3 The correct answer is **D**. By indicating the columns immediately following the SELECT keyword, you limit the results to only those columns.
- 4 The correct answer is **B**. Use the WHERE clause to limit the rows returned to the criteria you indicate. In this example, the WHERE clause singles out the customer ID of 0001.
- 5 The correct answer is **C**. First you indicate the columns, then the target table, then the source. This results in the new table being created and populated with the results of the query.
- 6 The correct answer is **B**. When this statement is submitted, the results will be returned in descending order (largest value first), based on the Qty column.
- 7 The correct answer is **A**. To make changes to an existing row, you use the UPDATE statement. You could delete and then insert a new row, but that would require statements that you don't really need if you just use the UPDATE statement. Simply indicate the table and the values to update. You can also include the WHERE clause if needed to limit the scope of your UPDATE statement.
8. The correct answer is **D**.
- 9 The correct answers are **A and C**. Sorting is provided by the ORDER BY clause, not by the index. Indexes are largely for optimization, providing the database engine with the information it needs to fulfil your query in a timely manner.
- 10 The correct answers are **B and C**. Relational database systems, the foundation for the SQL language and its capabilities, are built to be more accessible, and are much more contained, typically having only one physical file on disk that contains the database and index files.

Section B – Short Answer [35 Marks]

Question 1 [9]

Start with : Products(Prodnum, Desc, Macnum, Setup, Prorate, Igd, Amt).

Prodnum was defined as being unique so it is an obvious key:

Products(Prodnum, Desc, Macnum, Setup, Prorate, Igd, Amt).

Remove the first repeating group of Macnum, Setup, and Prorate, as follows:

(a1) **Settings(Prodnum, Macnum, Setup, Prorate).**

We are told what the key for this relation has to be (the given dependencies).

Now remove the second repeating group of Igd and Amt:

(a2) **Ingredients(Prodnum, Igd, Amt).**

We are told what the key for this relation has to be (dependencies given).

Removing these two repeating groups (a1) and (a2) leaves:

(a3) **Products(Prodnum, Desc).**

2NF.

For both (a1) and (a2), the non-key fields are dependent on the whole compound key (we are told this in the dependencies given). (a3) does not have a compound key, so 2NF is not applicable.

3NF.

This step applies only to (a1), as this is the only relation which could have dependencies between non-key fields. However the dependencies given indicate that setup and prorate are not dependent upon one another, so 3NF is not applicable. Hence the relations (a1), (a2) and (a3) are now in 3NF. ANSWER.

Question 2 [2]

Yes, there is a problem of referential integrity. The foreign key Vend_Code in Product_table does not have matching values (for the same field name) in the Vendor_table.

Question 3 [2,2]

a) Make sure that the first two columns are not null to help maintain essential information in the database table.

CREATE TABLE statement you created should resemble the following example:

```
(  
  Product_ID   NUMERIC (10) NOT NULL,  
  Product_name char (30) NOT NULL,  
  Product_description char (50)  
)
```

b)The statement you'll need has to call out the table and column you want to index. You'll also have to remember to name the index. The correct statement is (your index name may be different):

CREATE INDEX idx_Product ON Products(ProductID)

This creates a new index on the Products table as needed.

For the second part of the exercise, you need to write the SQL that would drop the index you've created just now. The correct statement would be:

DROP INDEX idx_Product

Question 4 [1.1]

The UPDATE statement

Did you remember to use only the customer ID for the WHERE clause? Here's what the resulting statement should be:

UPDATE customer SET city='Trenton', state='NJ' WHERE customerID=5

Here's something to note: Because you're indicating more than one item to be updated, by showing both the city and the state in the same statement, the SQL engine will update both columns in the row you called out in the WHERE clause. Also, remember to include the single quotes around the city and state because they are CHAR type fields (character values).

The DELETE statement you created should reference the customer ID for Tabinda. The statement you created should resemble the following:

DELETE FROM customer WHERE customerID=3

This will remove only the row with the customer ID of 3, and will leave the remaining rows intact.

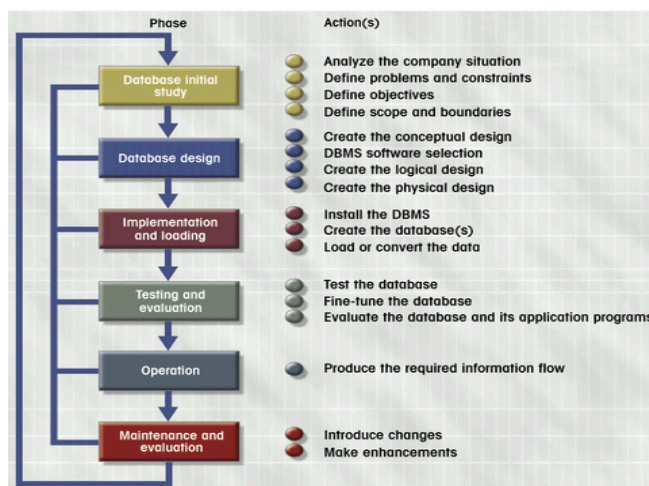
Question 5 [2]

Data is the raw facts, and the building block on which information is produced.

Information is the outcome of processing Data into a useable often summarised format which can be used for decision making. Data is given context and used in the generation of information

Question 6 [1,4]

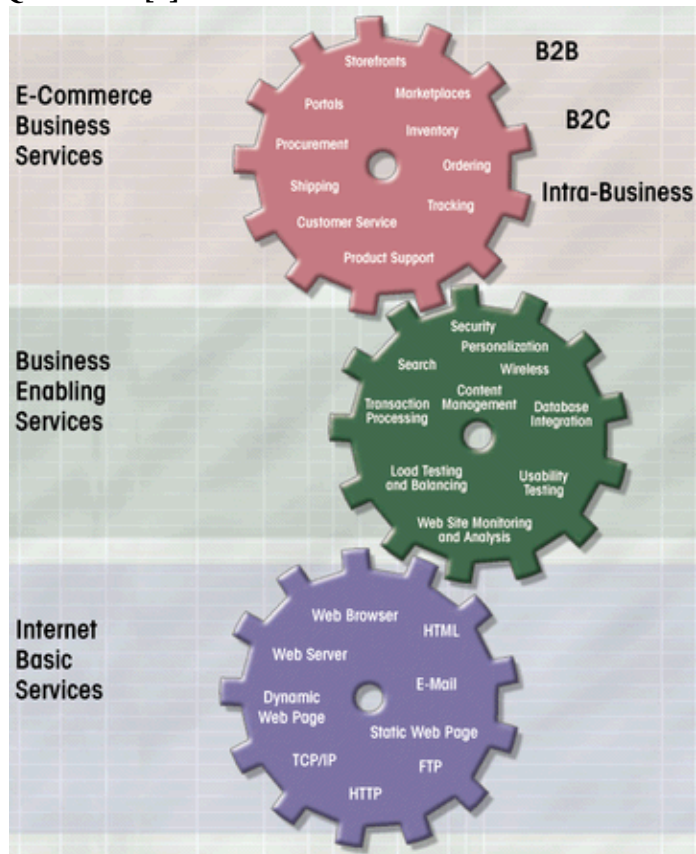
- a) DBLC – DataBase Life Cycle – The process whereby databases are developed.
- b) DBLC consists of
 - a. DB Initial Study
 - b. DB Design
 - c. Implementation and Loading
 - d. Testing and Evaluation
 - e. Operation
 - f. Maintenance and Evaluation



Question 7 [4]

- Atomicity
 - All parts of must be completed OR none
 - Indivisible Work unit
- Durability
 - On completion DB is in a consistent state
- Serializability
 - Concurrent execution of multiple transactions
 - Concurrency can mean that transactions are handled in a serial manner
- Isolation
 - Data used in a transaction cannot be used by another until the first completes

Question 8 [3]



Question 9 [1,2,1]

a) Without compression: `pg_dump dbname > outfile`

With compression : `pg_dump dbname | gzip > filename.gz`

b) Backup would be smaller, as well as consistent, which is not ensured by a file level backup

c) Smaller due to indexes not being backed up, as they would be re-created on the fly.