

NATIONAL UNIVERSITY OF SINGAPORE

SCHOOL OF COMPUTING

ASSESSMENT 2 FOR
Semester 2 AY2015/16

CS4221 - DATABASE DESIGN

April 2016

Time Allowed: 2 Hours

INSTRUCTIONS TO CANDIDATES

1. This assessment paper contains **FIVE (5)** questions and comprises **EIGHT (8)** printed pages, including this page.
2. Answer **ALL** questions as indicated in the space provided on the script.
3. This is a **CLOSED BOOK** assessment.
4. One double sided handwritten or printed cheat sheet is allowed.
5. Your group notes on Data Warehousing are allowed.
6. Please write your Matriculation Number below.

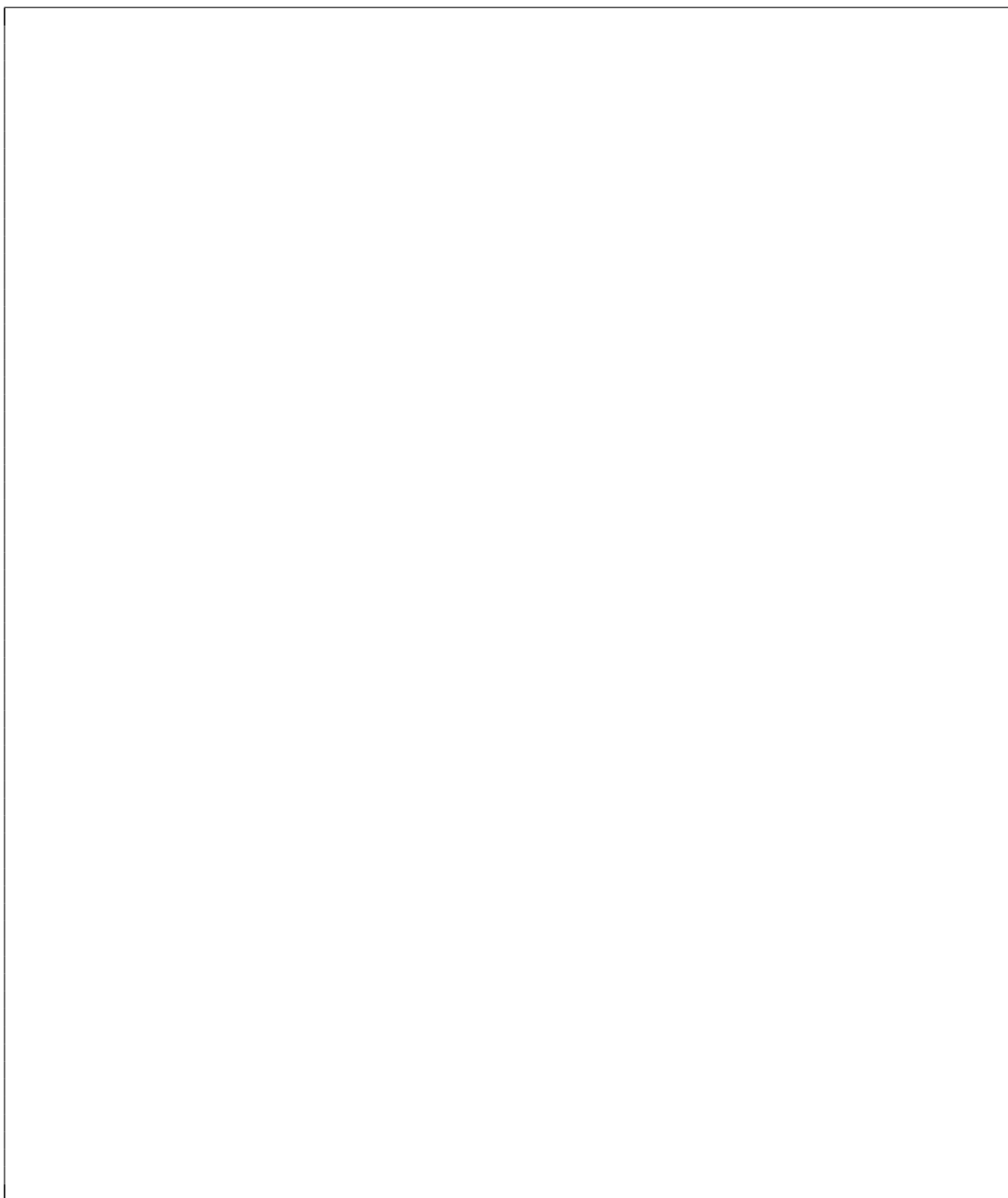
MATRICULATION NUMBER: _____

EXAMINER'S USE ONLY		
Question	Marks	Score
1	3	
2	3	
3	3	
4	8	
5	8	
TOTAL	25	

Question 1 [3 marks]

We are interested in the sales of residential properties in Singapore by agents. We look at the cost of purchasing from the buyer's point of view. Different types of residential properties are sold in Singapore: condominiums, walk-ups, cluster houses, executive condominiums, landed houses, HDB apartments. A property has a unique address, postal code, and district. A property is typically described by its floor area in square feet and square meters, number of bedrooms, number of bathrooms, floor level or number of floors, whether it has air conditioning and amenities (swimming pool, security, etc.) and by its development information (developer, tenure, Temporary Occupation Permit year, etc.). For a given transaction we know the purchase price and the fees (we ignore the breakdown of the various duties and fees into buyers agent fee, stamp duty, etc., in this exercise). Real estate market analysis typically also looks at purchase price per square foot and per square meter. The details about the agent who sold the property include her name, agency, Council for Estate Agencies registration number, and contact information.

- (a) Design a warehouse schema for the analysis of this property-for-sale listing process. Draw the logical diagram (tables, attributes, primary keys, and foreign keys). (3)



Question 2 [3 marks]

PostgreSQL does not create indexes for foreign key constraints.

- (a) Explain what happens to foreign key constraints in most designs in terms of indexes. Discuss the consequences on updates and query performance. Discuss the advantages and disadvantages of PostgreSQL not creating indexes for foreign key constraints. (3)

Question 3 [3 marks]

PostgreSQL provides four isolation levels: Read Uncommitted, Read Committed, Repeatable Read, and Serializable. The default is Read Committed.

(a) Fill in the following table with Possible (P) / Not Possible (N)

(3)

Isolation Level	Dirty Read	Non-repeatable Read	Phantom Read
Read uncommitted	.	.	.
Read committed	.	.	.
Repeatable read	.	.	.
Serializable	.	.	.

Question 4 [8 marks]

The goal of this exercise is to identify examples of Web applications that would benefit of various NoSQL data management technologies with different data models and query languages. For every technology listed below, provide a realistic example of a Web-based service (it can be an existing one such as Google Search, Amazon, or Facebook or an imaginary one), for which this specific NoSQL technology would be well-adapted. In one or two paragraphs, explain why the technology would suit this service, and discuss the limitations and disadvantages that arise from the use of this technology and why they would not be critical to the service.

- (a) A native XML database such as eXist or BaseX.

(2)

- (b) A distributed column store such as BigTable or HBase.

(2)

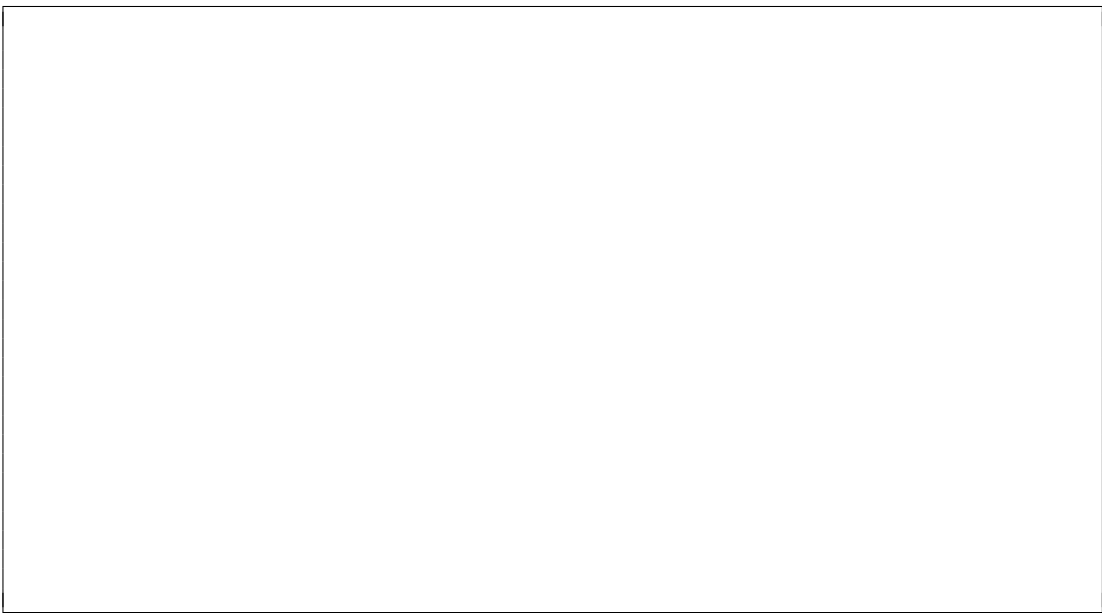
(c) A graph database such as Neo4J.

(2)



(d) A document store such as MongoDB or CouchDB.

(2)



Question 5 [8 marks]

Consider the following PHP fragment extracted of a Web comment system. The code adds a comment to a MySQL database if one is provided as an HTTP parameter (through a Web form on the same page, not shown here), and display all comments in the current thread, the thread identifier being also provided by an HTTP parameter. The authenticated user is extracted from a session variable, attached to the user's cookie. It is not necessary to know the detailed syntax of PHP or of the `mysqli` library to follow the code. As written, the code is syntactically correct, runs apparently correctly, and allows a user to add and list comments as intended.

```
<?php
    $connection = mysqli_connect("localhost","dbuser","dbpassword","dbname");
    $thread = $_GET["thread"];
    $user = $_SESSION["user"];

    if(isset($_GET["comment"])) {
        /* Add the comment to the database */
        $prepared = mysqli_prepare($connection,
            "INSERT INTO Comments (date, thread, user, text) VALUES (NOW(),?,?,?)");
        mysqli_stmt_bind_param($prepared, "iss", $thread, $user, $_GET["comment"]);
        mysqli_execute($prepared);
    }

    /* Display all comments for this thread */
    $result = mysqli_query($connection,
        "SELECT user,text,date FROM Comments WHERE thread=$thread ORDER BY date DESC");

    while($row = mysqli_fetch_object($result)) {
        echo "<p>";
        echo $row->user." wrote:<br>";
        echo $row->text;
        echo "</p>";
    }
?>
```

- (a) Does the code contain a *Cross-Site Request Forgery (XSRF)* vulnerability? If not, explain why. If the code does contain such a vulnerability, give an example of a possible attack, of what the impact would be, and of how to remedy this vulnerability. (2)

- (b) Does the code contain a *Race Condition* vulnerability? If not, explain why. If the code does contain such a vulnerability, give an example of a possible attack, of what the impact would be, and of how to remedy this vulnerability. (2)

- (c) Does the code contain a *SQL Injection* vulnerability? If not, explain why. If the code does contain such a vulnerability, give an example of a possible attack, of what the impact would be, and of how to remedy this vulnerability. (2)

- (d) Does the code contain a *Cross-Site Scripting (XSS)* vulnerability? If not, explain why. If the code does contain such a vulnerability, give an example of a possible attack, of what the impact would be, and of how to remedy this vulnerability. (2)