

WARNING

This material has been reproduced and communicated to you by or on behalf of *Charles Darwin University* in accordance with section 113P of the *Copyright Act 1968 (Act)*.

The material in this communication may be subject to copyright under the Act.
Any further reproduction or communication of this material by you may be the subject of copyright protection under the Act.

Do not remove this notice



Family Name					
Given Name/s					
Student Number					
Teaching Period	Semester 1, 2019				

HIT326 – Database-Driven Web Applications	DURATION	
	Reading Time:	10 minutes
	Writing Time:	120 minutes
INSTRUCTIONS TO CANDIDATES		
<p>The examination has ONE section. ALL questions must be answered on the Answer Booklet provided.</p> <p>Please ensure that your name and student number are clearly indicated both on your Answer Sheet and at the top of this examination paper.</p> <ul style="list-style-type: none"> • Read ALL questions carefully. • Do not commence writing until instructed to do so. • Total marks – 60 marks. 		
EXAM CONDITIONS		
<p><u>You may begin writing from the commencement of the examination session.</u> The reading time indicated above is provided as a guide only.</p>		
This is a CLOSED BOOK examination		
No calculators are permitted		
No handwritten notes are permitted		
Hard copy, unannotated English translation dictionary only		
ADDITIONAL AUTHORISED MATERIALS	EXAMINATION MATERIALS TO BE SUPPLIED	
No additional printed material is permitted	1 x 20 Page Book 1 x Scrap Paper	

**THIS EXAMINATION IS PRINTED
DOUBLE-SIDED.**

**THIS PAGE HAS BEEN INTENTIONALLY
LEFT BLANK.**

Question 1: Web and Database Concepts

(6 + 4 + 4 + 6 = 20 marks)

Q1a.

Discuss the mechanisms behind the HTTPS protocol and certificates and explain when they should be implemented in Web applications.

(Marks: 6)

Q1b.

Describe both the relation and difference between sessions and cookies.

(Marks: 4)

Q1c.

Explain CRUD in the context of typical HTTP requests and SQL operations.

(Marks: 4)

Q1d.

Discuss the application of folksonomies in modern web applications. Give an example on how folksonomy as a tagging system is implemented in databases. Write SQL or draw diagram if necessary.

(Marks: 6)

Question 2: PHP and MySQL

(4 + 6 + 4 + 6 = 20 marks)

Q2a.

Suppose a PHP script contains an HTML form using the POST method. The form has a studentID input box defines as:

```
<input id="sid" type="text" name="studentID" />
```

Write PHP code that checks if a value has been set for the studentID, and if so adds the input value to a session variable with the same name as the key. Note that your code should also include statements for starting and closing a session.

(Marks: 4)

Q2b.

Consider the following PHP array.

```
$messages = array();
```

```
$messages[] = "E2134";
```

```
$messages[] = "Database problem";
```

```
$messages[] = "Error: database login failed.";
```

Write a PHP function (with a sensible name) that takes an array (like the \$messages above) as a parameter, changes the array's first element to string "E1234", and prints the following with a loop:

E1234

Database problem

Error: database login failed.

(Marks: 6)

Q2c.

Assume the following variables have been declared and initialised

```
$dbName = "mydb";
```

```
$host = "localhost";
```

```
$username = "username";
```

```
$password = "password";
```

Using the above variables, write a single line PHP statement that will start a connection to the server, open the database **mydb** and store the connection in variable **\$dbConnect**.

(Marks: 4)

Q2d.

Suppose a connection has been successfully made to the database in question **Q2c** and stored in the variable **\$dbConnect**. The **mydb** database has a table called **results** with the following fields: **student_id**, **unit_id** and **result**.

Write PHP code that first retrieves the **student_id** and **result** for all students with a **unit_id** of HIT326, then store the result set in a variable called **\$unit_results**, and lastly close the database connection.

(Marks: 6)

Question 3: Design Patterns

(6 + 4 + 4 + 6 = 20 marks)

Q3a.

Discuss the “Model-View-Controller” (MVC) design pattern and use an example or diagram to show to what extent it can be used to organize and separate the application code.

(Marks: 6)

Q3b.

Write two regular expression patterns with the first checks if a whole password contains between 8 and 255 letters, numbers or punctuation marks; and the second checks if a password contains at least one uppercase letter.

(Marks: 4)

Q3c.

Briefly explain what “SQL Injection attacks” are and how you can defend an application against them. Also provide an example to illustrate your answer.

(Marks: 4)

Q3d.

Draw a diagram that describes the Ajax Web application model. Briefly describe the advantage of Ajax application model compared with the traditional HTTP model.

(Marks: 6)

END OF THE EXAMINATION.

**THIS PAGE HAS BEEN INTENTIONALLY
LEFT BLANK.**