



## 12 Pages

1. Please give the information requested in the boxes below.
2. Enter your name in the space provided to the right, fold over and seal down (using the label provided) to hide your name.
3. The answers must be legibly written. The Examiners can either ignore illegible script or have it typed and the expense of typing charged to the candidate.
4. Write your answers in ink.
5. Begin each answer on a fresh page. Do all rough work in the answer book and cross it through.
6. No part of this book is to be torn out; it must be handed in complete.
7. If more books than one are used, fasten them together using the tags available from the invigilator(s).
8. **Every book must be handed in. In no circumstances may answer books, used or unused, be taken out of the examination venue.**

(see note 2 below)

SURNAME: Whiteley

FORENAME(S): Sol

|   |                                  |                                   |
|---|----------------------------------|-----------------------------------|
| 1 | Student Number                   | 149357                            |
| 2 | Title of Examination Paper       | DATABASE<br>MANAGEMENT<br>SYSTEMS |
| 3 | Module Code                      | ACCA5014                          |
| 4 | Date and Time of Examination     | 09/05/19<br>9:30                  |
| 5 | Examination Venue                | Pembrokeshire<br>College          |
| 6 | Seat Number                      | 17                                |
| 7 | Number of Answer Books handed in |                                   |

| List the numbers of the questions you have attempted | Office Use |
|------------------------------------------------------|------------|
| 1                                                    | 22         |
| 3                                                    | 16         |
| 4                                                    | 16         |
|                                                      |            |
|                                                      |            |
|                                                      |            |
|                                                      |            |
|                                                      |            |
|                                                      |            |
|                                                      |            |
|                                                      |            |
|                                                      |            |
|                                                      |            |
|                                                      |            |
|                                                      |            |
| Office Use                                           | 54         |



Question 1.

a.

i) CREATE TABLE TutorBooking (

bookingRef CHAR (5) NOT NULL,  
 bookingDate DATETIME NOT NULL CONSTRAINT  
 DF\_bookingDate DEFAULT (DATENOW()),  
~~bookingDate~~ tutorNo CHAR (5) NOT NULL,  
 studentNo CHAR (5) NOT NULL,  
 subjectId CHAR (5) NOT NULL,  
 subjectLevel VARCHAR (20) NOT NULL,  
 hourlyRate ~~INT~~ INT NOT NULL  
 CONSTRAINT CHK\_hourlyRate CHECK (IN ('10', '15', '20', '25'))  
 ); ✓

ii) ALTER TABLE TutorBooking ADD  
 CONSTRAINT PK\_bookingRef PRIMARY KEY  
 (bookingRef); ✓

iii) ALTER TABLE TutorBooking ADD CONSTRAINT  
~~FK\_TutorBookingTutorNo~~ FK\_tutorNo\_TutorBooking FOREIGN  
 KEY (tutorNo) REFERENCES Tutor (tutorNo),  
 CONSTRAINT FK\_studentNo\_TutorBooking FOREIGN KEY  
 (studentNo) REFERENCES Student (studentNo),  
~~CONSTRAINT FK\_studentLevel~~ FK\_subjectId\_TutorBooking  
 FOREIGN KEY (subjectId) REFERENCES Subject  
 (subjectId); ✓

~~Atomicity~~  
~~Consistency~~  
~~Isolation~~  
~~Recovery~~

external  
 conceptual  
 internal  
 physical

iv) CREATE TABLE TutorSubject (

tutorNo CHAR (5) NOT NULL,  
 subjectID CHAR (5) NOT NULL,  
~~KSLevel~~ INT,  
~~KSLevel~~ INT,  
~~KSLevel~~ hLevel INT,  
 minHourlyRate INT NOT NULL CONSTRAINT  
 CHK-minHourlyRate CHECK (IN ('5', '10',  
 '15'),  
 maxHourlyRate INT NOT NULL CONSTRAINT  
 CHK-maxHourlyRate CHECK (IN ('20', '25', '30'),  
 CONSTRAINT PK\_TutorSubject PRIMARY KEY  
 (tutorNo, subjectID),  
 CONSTRAINT FK-subjectID-TutorSubject  
 FOREIGN KEY (subjectID) REFERENCES  
 Subject (subjectID)  
 );

TutorNo Foreign key?

7 ✓

b.

i) SELECT (tutorNo, title, first Name, Surname, email,  
~~contactEmail~~, ~~contactAddress~~, city, postcode, phoneNo)  
 FROM Tutor WHERE DBS (DBSEXP  
 (DBSEXPDate > 30/05/2019 AND DBSEXPDate  
 < 01/07/2019); ✓

2 ✓

ii) ~~UPDATE~~ UPDATE TABLE Tutor SET  
 (status = (~~Expired~~ Pending DBS Check)) WHERE ~~status~~  
~~(status) = (Expired);~~

2 ✓

iii) SELECT MIN(<sup>AS minHourlyRate</sup>minHourlyRate), MAX(<sup>AS maxHourlyRate</sup>maxHourlyRate),  
 AVG(minHourlyRate, maxHourlyRate) ~~FROM~~ <sup>AS</sup> averageHourlyRate FROM Tutor WHERE  
 TutorBooking b = (booking Date > 31/12/2018 AND  
 < 01/01/2020)  
 INNER JOIN tutorNo t = tutorNo b;

0

backups  
 accidental damage  
 purposeful damage

(22) ✓

Question 3.

- a. The first example of a threat is when a user accidentally directs a ~~threat~~ virus to the database, allowing the virus to ~~collect~~ data complete the CRUD operations to interact with the data. This can be very harmful as sensitive data can be accessed. 2 ✓
- Another example of a threat is when ~~the~~ physical damage is caused to a server containing the data, if the data isn't securely backed up and unretrievable then this could cause a loss in data. 2 ✓
- Accidental damage can also be caused if a user has too much authorisation which ~~could~~ <sup>may</sup> cause and accidentally ~~deletes~~ <sup>modifies</sup> an important query. 2 ✓

b)

i) Authentication is used to verify the identity of a user and consists of login and user type queries to create authentication.

2 ✓

ii) Access controls are used to either grant or revoke privileges for an individual user so they can perform actions like reading or updating for a specified table.

2 ✓

iii) Views are another type of method used to specify the tables each individual user can view and it is possible to setup a horizontal, vertical, or grouped or joined view.

2 ✓

c) CREATE VIEW ActiveTutors SELECT (title t, firstName t, surname t, email t, city t, subjectId b, subjectLevel subjectLevel b, minHourlyRate s, maxHourlyRate s) FROM Tutor t, TutorSubject s, TutorBooking b WHERE status t = ('Active')  
INNER JOIN tutorNo t = tutorNo b,  
INNER JOIN tutorNo t = tutorNo s,

4 ✓

Subject join?

(16)

Question 4

a) Atomicity, Consistency, Isolation and durability are all important properties of a transaction that ensures the database is consistent and reliable.

2 ✓

b) Concurrency control ensures that a transaction is committed before starting another transaction.

1 ✓

c) i)

lost update is caused when the user reads a balance <sup>from a transaction</sup> during another transaction which causes the <sup>first transactions</sup> ~~first~~ updated balance to become ~~x~~ obsolete. &

Uncommitted dependency is caused when the commit action is caused during another transaction. Inconsistent analysis is caused by the user not ~~beginning~~ <sup>processing</sup> transactions one at a time.

2 ✓

ii) ~~see other papers~~ The problem occurred because the balance has lost 100 as transaction 2 was overwritten by transaction 1. This can be prevented by committing t2 before t1 reads the balance.

7 ✓

d) i) T3 needs to be redone as we do not know if it was successful as it carried on after checkpoint.

1 ✓

ii) T4 needs to be undone as we ~~do~~ know that this transaction has failed.

1 ✓

iii) T1 and T2 are the transactions that were not affected by the crash

2

(16)

even though T1 was aborted. ~~The~~  
~~if~~ ~~Time~~ ~~if~~ ~~that's the way~~  
we know this because we can  
~~if~~ see these transactions were  
completed before the checkpoint. ✓

Student Number: .....149357.....

| Time | T <sub>1</sub>                           | T <sub>2</sub>                            | bal <sub>x</sub> (£500) |
|------|------------------------------------------|-------------------------------------------|-------------------------|
| t1   |                                          | BEGIN TRANS                               | 500                     |
| t2   | BEGIN TRANS                              | Read bal <sub>x</sub>                     | 500 ✓                   |
| t3   | Read bal <sub>x</sub>                    | bal <sub>x</sub> = bal <sub>x</sub> + 100 | 500 ✓                   |
| t4   | bal <sub>x</sub> = bal <sub>x</sub> - 50 | Write bal <sub>x</sub>                    | 600 ✓                   |
| t5   | ...                                      | COMMIT                                    | 600 ✓                   |
| t6   | Write bal <sub>x</sub>                   |                                           | 450 ✓                   |
| t7   | COMMIT                                   |                                           | 450 ✓                   |

Table 1: Table for Question 4c.ii.

## DATABASE MANAGEMENT SYSTEMS (ACCA5014) EXAM MAY 2019

### Personal Tutors Case Study

Personal Tutors is an online service designed to help registered members find tutors, communicate with them and book and pay for lessons. Tutors are selected carefully and all registered tutors must have a valid, up-to-date DBS certificate. Members can book lessons with a preferred tutor at an agreed rate. Booked lessons can be cancelled within 24 hours of the booked lesson time. All payments are handled through the platform service. Payment is taken once the lesson has taken place and payments are made to tutors at the same time.

A subset of the current database schema for the Personal Tutors platform is shown below. Please note that the database schema does not truly reflect a real world scenario as it has been simplified for examination purposes.

| Table        | Columns                                                                                                                                  |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------|
| Subject      | <u>SubjectId</u> , Subject                                                                                                               |
| Tutor        | <u>TutorNo</u> , Title, FirstName, Surname, DOB, Email, City, Postcode, PhoneNo, DBSEpiryDate, Status                                    |
| TutorSubject | <u>TutorNo</u> , <u>SubjectId</u> , KSLevel, ALevel, HELevel, MinHourlyRate, MaxHourlyRate                                               |
| Student      | <u>StudentNo</u> , Title, FirstName, Surname, ContactEmail, ContactAddress, City, Postcode, PhoneNo, CardNo, CardCVV, CardExpiry, Status |
| TutorBooking | <u>BookingRef</u> , BookingDate, TutorNo, StudentNo, SubjectId, SubjectLevel, HourlyRate                                                 |
| BookedLesson | <u>BookingRef</u> , <u>LessonDateTime</u> , DurationMins, Location, CancellationDateTime, PaymentDetails, TutorFee                       |

Table 2: Current database schema for Personal Tutors database (Note: primary key fields for each table have been underlined)

Where:

| Table        | Description                                                                                                                  |
|--------------|------------------------------------------------------------------------------------------------------------------------------|
| Subject      | Contains a list of different subjects available.                                                                             |
| Tutor        | Contains details of tutors registered with Personal Tutors.                                                                  |
| TutorSubject | Contains a list of subjects taught by a tutor.                                                                               |
| Student      | Contains details of student members registered with Personal Tutors.                                                         |
| TutorBooking | Contains details of a tutor booked by a student for a particular subject. It includes the hourly rate agreed with the tutor. |
| BookedLesson | Contains details of the individual lesson dates and times the student has booked as part of a set of tutoring sessions.      |