FIT9132 2022 S1 Exam Feedback

The S1 2022 exam was a "Closed Book with Specifically Permitted Items" exam run on the University eExam platform. 290 students sat the exam, 26 students were not present for the exam. Of the 26 students who were not present, 14 were granted a DEF exam.

The final exam result was scaled to a mark out of 75 to compensate for the number of students who did not attempt, or partially attempted questions and for issues related to the online access to the database servers.

72 exam papers were second marked due to a failure to meet the exam hurdle requirement of 45% against the scaled mark. The papers were remarked by the lecturing team.

The scaled average exam mark for all students who sat the exam was 60.9% (2021 S1 60.1%, S2 57.6%), the average overall FIT9132 unit mark for all students was 62.6 % (2021 S1 65.5%, S2 62.5%)

Q1 Relational Model 5 marks
Average mark awarded: 2.2 mks

Students needed to clearly explain the terms super key, candidate key, primary key, surrogate key and functional dependency plus provide suitable examples from the provided SHOW_TICKET relational data (0.5 mk explanation, 0.5 mk example = 1 mk x 5 = 5 mks).

The common errors made here

- *failure to explain the listed terms correctly-*
    - *Super key → any combination of attribute/s which shows uniqueness property*
    - *Candidate key → minimal superkey*
    - *Primary key → chosen candidate key*
    - *Surrogate key → an attribute that is introduced to replace a composite PK*
    - *Functional dependency → an attribute/s that determines one single value of another attribute/s at any given time*
- *failure to provide suitable examples for each term*

Q2 Relational Algebra 10 marks
Average mark awarded: 5.7 mks

Students were required to *clearly* demonstrate an understanding of the efficiency of their solution by limiting the number of rows and columns which were passed up through the query.

(a) A select of PART for partstock < 100 followed by a project of the required attributes (0.5 mk select, 0.5 mk project). Correct symbols and relational algebra form was required. (1 mk total)

(b)
(i) Select or ORDERS between 01-05-2022 and 31-05-2022 followed by project of custno and slsrno => 1 mk
(ii) Join of (i) with projected slsrno, slsrname and slsrphone from SLSREP to determine details for these orders, followed by project of custno, slsrname, slsrphone => 1.5 mks
(iii) Join of (ii) with projected custno, custname from CUSTOMER followed by a project of the required attributes => 1.5 mk (4 mks total)

(c)
(i) select from CUSTOMER to obtain Jack Witherheads details, followed by project of custno => 1 mk
(ii) Join of (i) with projected ordno, custno from ORDERS, followed by project of ordno to obtain the customers orders => 1 mk
(iii) Join of (ii) with projected partno, orderno from ORDERLINE to find parts ordered by customer => 1 mk
(iv) subtract of (iii) from project partno from PART to find part nos not ordered by customer => 1 mk
(v) join of (iv) with projected partno, partdesc to find details of parts not ordered by customer => 1 mk (5 mks total)

Common errors made here included:

- *trying to join relations with no matching attributes*
- *not selecting only the necessary tuples before joining/Not projecting only the necessary attributes before joining (ie. failure to identify efficiency)*
- *incorrect/misspelt attribute names used*

For all SQL questions, for *all* students, the logic of the submitted answer was marked so as to allow for students who were unable to connect to the database server.

Q3. Basic SQL 8 marks
Average mark awarded: 5.4 mks

- correct attributes 2 mks
- appropriate headings 1 mk
- join 4 mks (join to species must be on both spec_genus and spec_name)
- order by 1 mk

Common errors made here:

- *incorrect join conditions between animal, species and breeding_event*
- *column alias not used or where not meaningful*

Q4. SQL Intermediate 14 marks
Average mark awarded: 4.8 mks

- Calculation of average (sum, divide by number of centres, use of distinct) 6 mks
- Format of output (lpad, to_char, As) 5 mks
- JOIN - outer join required (ALL centres) 3 mks

Common errors made here:

- *incorrect approach to work out average eg worked out for each centre, rather than the one overall average*
- *incorrect format for output, incorrect use of lpad*
- *did not use outer join*

Q5. SQL Intermediate 14 marks
Average mark awarded: 6.6 mks

- All attributes listed 1 mk
- Determination of count of animals critically endangered
    - nested select 3 mks
    - upper(redlist_category) = 'CRITICALLY ENDANGERED' 3 mks
    - join 2 mks
    - count 2 mks, AS 1 mk
- FROM 1 mk
- ORDER BY 1 mk

Common errors made here:
- *incorrect approach to work out count for critically endangered*
- *incorrect join*
- *did not use upper*

Q6. Advanced SQL 19 marks
Average mark awarded: 9.9 mks

- select attribute list includes all required 1 mk
- expansion of animal sex (case or decode) 2 mks, AS 1 mk
- count 2 mks, AS 1 mk
- join 1mk
- group by 3 mks
- having count(*) > 2 mks
    - determination of average exchanges 5 mks
- order 1 mk

Common errors made here:
- *omitted expand sex*
- *incorrect determination of average exchanges*
- *using where instead of having*

For all NoSQL questions, for *all* students, the logic of the submitted answer was marked so as to allow for students who were unable to connect to the respective database server.

Q7. Generate JSON 8 marks
Average mark awarded: 5.5 mks

- Select main JSON_OBJECT 2 mks
- code for embedded JSON_OBJECT 2 mks
- use of to_char for exchange date 1 mk
- join 2 mks
- group by 1 mk

Common errors made here:
- *incorrectly formatted JSON syntax*
- *missing attributes*
- *incorrect join*
- *incorrect group by*

Q8. MongoDB 7 marks
Average mark awarded: 4.0 mks

(a)

- 2 x 0.5 mk condition ($and allowed)

  => 1 mk total

Common errors made here:
- *incorrect syntax*
- *incorrect use of parameters*

(b)

- 0.5 mk find
- 0.5 $OR, 1 mk conditions
- Control of output - suppress _id 0.5 mk, list other attributes 0.5 mk

  => 3 mks total

Common errors made here:
- *incorrect syntax*
- *incorrect use of parameters*
- *not restricting to both to or from SAF30*

(c)

- db.animal.update 0.5 mk
- condition for update 0.5 mk
- push to add new details for exchange 0.5 mk
- 3 x 0.5 mks for new values

  => 3 mks total

Common errors made here:
- *incorrect syntax*
- *incorrect use of parameters*
- *not selecting the document for update correctly*
- *not adding exchange correctly*

Q9. Lock Table 5 marks
Average mark awarded: 3.6 mks

Correct cell values (3 mks) - S, X or WAIT required in each appropriate cell
- 0.5 mk for each error in table to a maximum of 6 errors (0/3)

Overall this question was well answered. Common errors made here:
- *mixed up type of lock applied (S vs X)*
- *wrote Wait Tm rather than Tn Wait Tm as required,*
- *incorrectly applied X lock where resource was already S locked by a different transaction*

YES deadlock exists = 1 mk
Because T1 waits for T2 and T2 waits for T1 = 1 mk

Common error made here:
- *lack of explanation or incorrectly explained reason for deadlock*

Q10. Transaction-1 5 marks
Average mark awarded: 1.2 mks

- Correct explanation of issue 2 mks
- Correct explanation of what a lost update is 2 mks
- Oracle uses Locks to solve issue 1 mk

Overall poorly attempted. Common errors made here:
- *not attempted*
- *Not providing the values or identifying the issue with this sequence of actions (only general answer used)*
    - *Time 0: User A READS*
    - *Time 1: User B READS*
    - *Time 2: User A UPDATES price*
    - *Time 3: User A WRITES to database*
    - *Time 4: User B UPDATES price*
    - *Time 5: User B WRITES to database*
        - *LOST UPDATE at time 3 grant_amount of 302500 overwritten by write at time 5*

Q11. Transaction-2 5 marks
Average mark awarded: 1.4 mks

- Stage 1 - prepare redo and undo lists 1 mk
- Stage 2
    - undo incomplete/rolled back transactions 1 mk
    - starting from newest 1 mk
- Stage 3
    - redo committed transactions 1 mk
    - starting from oldest 1 mk

Overall poorly attempted. Common errors made here:
- *not attempted*
- *incorrectly read as rollback/rollforward for each transaction rather than responding to the recovery process which is required*