MONASH University

Assessment 2: Relational and non-relational database manipulation

Word limit: N/A
Weighting: 50%
Individual/Team: Individual

Please Note that this assessment is due on 11.55pm AEST Wednesday 16 April 2025, this is to allow students in foundational units of the ITO programs the additional support in completing their first final assessments. This may not occur in all of your ITO units in the future, so please ensure any future units you undertake you are checking the submission dates.

Generative AI tools cannot be used in this assessment task

In this assessment, you must **not** use generative artificial intelligence (AI) to generate any materials or content in relation to the assessment task. This whole assessment task requires you to demonstrate human knowledge and skill acquisition without the assistance of AI.

After you have read this information, head over to the Assessment 2 Q&A category in Ed to ask any questions and see what your peers are saying about this assessment.

Assessment overview

For this assessment, you will populate resultant tables with appropriate test data and write the SQL queries specified for a brief for Monash Automotive.

This assessment supports unit learning outcomes 1, 3 and 4.

Assessment details

The following is a PDF version of the case study for this assessment, for offline use:

- Assessment 2: Relational and non-relational database manipulation – Monash Automotive (MA) Brief (PDF 598 KB).

Consider the following information:

Monash Automotive (MA) brief

Monash Automotive (MA) is a small business that services automotive vehicles from a single location. MA wishes to create a computer-based system to manage its service data, parts inventory, customers, and payments. After speaking to management, the following scenario describes what they would like to happen:

Customers drop their vehicles off at the Monash Automotive service centre, where a reception staff member makes a service entry into the system. Each service is assigned a unique service number, which the system should automatically generate. A customer indicates to the reception staff how they intend to pay for the service (Cash/Card/EFT). The customer will provide instructions for the work which needs to be carried out. For this simplified system, we will assume all services take place in one day.

MA mechanics will then work on the vehicle, a cumulative total of the hours spent on the service is recorded in the workshop, and a final service labour charge entry is struck when the service is completed. All parts required for the service will be obtained from the MA Spare Parts division as the work is carried out.

The mechanic will request an item/s from the Spare Parts division attendant, who will make a part charge entry, which will record the part number, the service number, the job number within that service, the number of items used and the total line cost for this/these item/s.

When all required work has been carried out, the total labour and total parts cost is updated in the service entry. The last service entry update carried out is to update the ready-for-pickup time, this signifies that the service has been closed. When customers pick up their vehicle, they are required to pay the full amount for the service.

To better profile MA's activities, the management would like to be able to analyse service activity by vehicle make, model and age.

A customer may sell/dispose of a vehicle—in such a situation the ownership will be transferred to the new owner. If the vehicle is no longer currently owned by an MA customer the vehicle customer number will be empty. Service records prior to the sale must be maintained as connected to the owner at the time of the service.

Monash Automotive Vehicle Service Invoice

<table>
<tr><td>Customer No: 1456</td><td rowspan="2">Customer Name & Address
Wiley E Coyote
123 Desert Chase
NeverEnding, Vic
3234</td><td>Service No: 1234</td></tr>
<tr><td>Reg. No.
WILDTNG</td><td>Year/Make/Model
2020/Ford Falcon/XR6 Sports</td></tr>
<tr><td>Vehicle Vin:
6WFFPAAB662573456</td><td>Contact Phone:
99032626</td><td>Kilometres
142,000</td></tr>
<tr><td colspan="2">Date of Service: 08/02/2024

Time Delivered: 8:30 AM

Time Promised: 4:00 PM</td><td>Payment:

Cash/Card/EFT</td></tr>
<tr><td colspan="3">Driver Instructions: [Maximum of 200 characters]</td></tr>
</table>

Standard 80,000 km service

Check/repair central locking unit

Wheel alignment

Part Charges

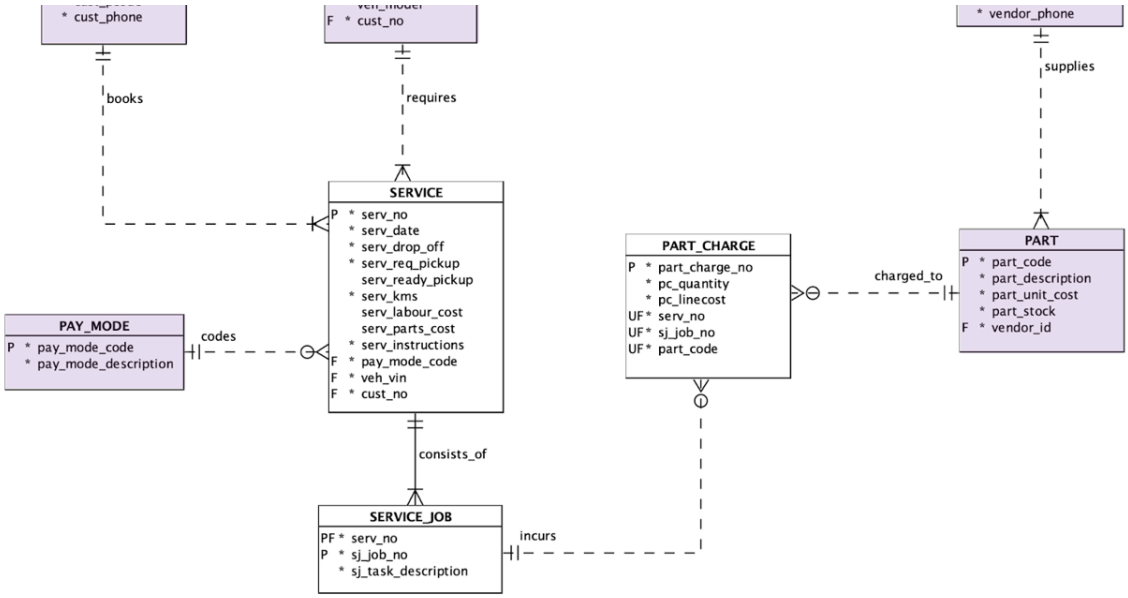
Job	Qty	Part Code	Description	Unit Price	Amount
Job# 1	1	FOA1MC	Oil Filter	15.58	15.58
Job# 1	1	FOAWV6	Refill Wiper Blades	12.80	12.80
Job# 1	2	FOAX200	Fuel Filter	25.89	51.78
Job# 2	1	FOBAC15	Transmitter SMT/Lock	176.58	176.58
Job# 3					
		Total - Parts			**256.74**

Total Labour	324.00
Total Parts	256.74
Amount Due	580.74

Note that every service has at least a job number 1, subsequent jobs required (based on the issues flagged by the customer) are numbered 2, 3 etc. The driver instructions are supplied when the vehicle is dropped off and mapped across to the jobs to be completed. For example, in the above, 'Standard 80,000 km service' is mapped to job 1, 'Check/repair central locking unit' is mapped to job 2 and 'Wheel alignment' is mapped to job 3

Data model

After some discussion with Monash Automotive, the following data model has been selected, as a starting point, to meet their basic needs:



Monash Automotive logic model (2022) Monash Online

Text alternative

Important points

1. The ass2-student.zip archive also contains several files for you to type your answers in, you should ensure these files are regularly pushed to the GitLab server so a clear development history is available for the marker to verify (a minimum of fourteen pushes are required—two per file used). In each file, you **must** fill in the header details with your name and student ID before beginning any work. Your SQL script files must not include any SPOOL or ECHO commands. Although you might include such commands when testing your work they must be removed before submission (a grade penalty will be applied if your documents contain spool or echo commands).

2. You are free to make assumptions, if needed. However, your assumptions must align with the details here and in the Ed Assessment 2 forum and must be clearly documented (see the required submission files).

   Remember you must keep up to date with the Ed Assessment 2 forum where further clarifications may be posted (this forum is to be treated as your client). Please be careful to ensure you do not post anything which includes your reasoning, logic or any part of your work to this assessment forum as doing so violates Monash plagiarism/collusion rules.

3. SQL views **must not** be used in arriving at any solutions for the tasks you are required to complete as part of this assessment.

4. In handling dates, the default date format must not be assumed; you must make use of the TO_DATE and TO_CHAR functions where appropriate.

5. ANSI joins must be used where the joining of tables in SQL is required.

6. In completing the following tasks, you must design your test data so that you always get output for the SQL scripts/queries specified below – this may require you to add further data as you move through completing the required tasks. Such extra data must be added as part of Task 2 (i.e. as part of your load of test data). Queries that are correct but do not produce any output ('no rows selected' message) using your test data will lose 50% of the marks allocated, so you should carefully check your test data and ensure it thoroughly validates your SQL queries.

   In answering the following tasks, you are ONLY permitted to use the RA/SQL/NoSQL structures and syntax which have been covered within this unit:

   - Module 2 The Relational Model
   - Module 3 Creating and Altering the Database Structure (DDL)
   - Module 4 DML & Transactions and SOL Part I

- Module 5 SQL Part II and III and the Oracle Common Functions document
- Module 6 Non-Relational Databases

SQL syntax and commands outside of the covered work, as detailed above, will not be accepted/marked.
You must NOT use PL/SQL commands such as BEGIN/END nor SQL structures such as WITH since these were not covered in the unit.

Assessment Tasks

Task 1: Relational Database Queries – Relational Algebra (6 marks)

Your answers for this task (Task 1) must be written in an MS Word document or Google document. To push the document to GitLab (a minimum of two pushes as you work) download or print the document as a pdf file and name the file as **T1-ma-ra.pdf**.

For this task, you are required to write the **relational algebra operations** for the following queries (your answer must show an understanding of query efficiency). Please ensure you copy and paste each of the questions below (a) - (c) into your answer document.

List of symbols for copying/pasting as you enter your answers below:

project: π, select: σ, join: ⋈, intersect: ∩, union: ∪, minus: -

Example

Question:

List the vehicle vin and make for all vehicles which were manufactured in the year 2021.

Answer:

R = π veh_vin, veh_make (σ veh_year= "2021" VEHICLE)

OR

R1 = σ veh_year= "2021" VEHICLE

R = π veh_vin, veh_make (R1)

(a) List the part code, part description for all parts supplied by the vendor named "Supercheap Auto".

[1 mark]

(b) List the part code and part description for all parts which have not been used in any service.

[2 marks]

(c) List the customer's name, phone number and vehicle registration number (rego) for all owners of vehicles who had their vehicle serviced on 22/02/2024 and where the service kilometres were greater than 80,000 km.

[3 marks]

Task 2: Data Insert (10 marks):

You may need to rerun the schema (ma-schema-insert.sql), especially when you have been experimenting with your solutions and may have corrupted the database unintentionally. If you suspect that there might be such problems, simply rerun the schema. The schema includes the appropriate drop commands at the head of the file.

Load selected tables with your own additional test data:

Using the supplied **T2-ma-insert.sql** script file. Add the SQL commands which will insert, as a minimum, the following sample data to represent completed services:

- 10 SERVICEs entries
  - Carried out on at least 6 different vehicles
  - Have at least 2 service entries that involve more than one job
  - Have at least 2 service entries that were ready for pickup before the requested pickup date/time and at least 2 that were ready after the requested pickup day/time
- 15 SERVICE_JOBs, and
- 10 PART_CHARGEs

Your inserted data must conform to the following rules:

1. You may treat all of the data that you add as a single transaction since you are setting up the initial test state for the database.
2. The primary key values for this data should be hardcoded values (i.e. **NOT** make use of sequences) and must consist of values below 200.
3. Dates for services must be chosen between the 1st January 2023 and the 1st June 2023.

For this task only, you are permitted to manually look up and include values for the loaded tables/data directly and carry out external calculations (for example part_charge.pc_linecost) where required.

You are **not** required to adjust the part_stock for any part which is used in a service. You may assume that the preloaded values represent the part stock after all of your data has been loaded.

Your parts used should be reasonably related to the task, for example, a broken tail light should not use a brake pad.

Note: In carrying out task 2 you must not modify any data or add any further data to the tables which were previously populated by the supplied schema file.

[10 marks]

For all subsequent tasks (Task 3 onwards) you are not permitted to:

- manually lookup an attribute/s in the database to obtain *any* value,
- manually calculate values (including dates/times) external to the database, e.g. on a calculator and then use such values in your answers. All necessary calculations must be carried out as part of your SQL code, or
- assume any contents in the database—rows in a table are potentially in a constant state of change

You must only use the data provided in the text of the questions. Where a particular case (upper case, lower case, etc.) for a word is provided you must only use that case. You may divide names such as Wiley Coyote into the first name of Wiley and the last name of Coyote if required.

Note: Failure to adhere to this requirement will result in a mark of 0 for the relevant question.

Task 3: Data Manipulation (10 marks)

For the following tasks, **your SQL must correctly manage transactions and use sequences to generate new primary keys for numeric primary key values** (under no circumstances may a new primary key value be hard-coded as a number or value). Your answers for these tasks must be placed in the supplied SQL Script **T3-ma-dml.sql.**

(a) Create a set of sequences that will allow you to enter data into the SERVICE and PART_CHARGE tables—all such sequences must start at 1000 and go up in steps of 10 (i.e. the first value is 1000, the next 1010 etc.).

[1 mark]

Task 3 (b) - (e) are potentially related questions. Where appropriate, you can use the information given in any of the parts to

answer any question e.g. use the information in (c) to help answer (d)

(b) The customer with phone number '6715573197', has called in and informed Monash Automotive that they have changed the registration number of the sole vehicle they own to 'GDD132'. You may assume that no other customer has this phone number.

[2 marks]

(c) Customer number 1030, Farrel Grazier has brought her Mazda CX-5 in for a service (the date should be treated as 21 March 2024). She brought it in at 8:30 am and required the vehicle back by 12 noon. The vehicle has completed 12,000 km and she will be paying by 'cash' if any charges are involved (this may be a warranty repair). Her reason for bringing the vehicle in is that the 'Rear seat belts are not properly retracting'. You may assume that Farrel Grazier only owns one Mazda CX-5

[2 marks]

(d) One of the MA mechanics starts the service job for the service in (c) above, they determine that the retraction issue was due to the belt mechanism being jammed. They remove the material causing the jam, so no parts are required. MA decides, in the interests of customer goodwill, that there will be no labour cost. The vehicle is ready to be picked up at 9:10 am.

Make these changes to the data in the database. These changes must be treated as a single transaction.

[3 marks]

(e) Monash Automotive have decided that they no longer wish to source any parts from 'Australian Automotive Parts'. An audit of MA's part usage shows that although they have several different Australian Automotive Parts items in stock, they have never been used for any service. They will return these items to the vendor and thus remove them from the MA part stock.

[2 marks]

Task 4: SQL Queries (46 marks):

Your answers for these tasks must be placed in the supplied SQL Script **T4-ma-select.sql**

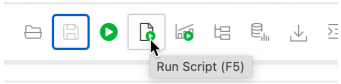
You are reminded that you must **design your test data so that you always get output for your SQL queries and thus test them appropriately**—this may require you to add further data as you move through completing the required tasks. Such extra data MUST be added as part of Task 2 (i.e. as part of your load of test data).

Where a question indicates 'Your output must have the form shown below', this means the same appearance and alignment of columns/data as the sample output shows. Your actual data may be different.

For each question you must carefully check the data and ensure that your result is correct, **do not simply assume it is correct if it runs successfully.**

In any query where the sort order (descending or ascending) is not specified, you should use the system default.

You should complete these queries using Oracle Database Actions (the ORDS server) so you are able to correctly align columns when needed (Visual Studio Code Oracle Explorer uses proportional fonts which does not give you the true alignment). Highlight the query you wish to run in ORDs and select the Run Script option:



Run script (2024) created by Monash Online

(a) Report the service number, vehicle vin, registration number, make, the service job numbers and service job descriptions for all services which have been completed.

The output should be in service number order, and within service number ordered by service job number. The most recent service number should be shown first.

[5 marks]

(b) For every part stocked by Monash Automotive list the part code, description, vendor ID and vendor name in a single VENDOR column separated by—(see below), the current stock held by MA and the total value of the parts stock currently held.

The output should be in part code order.

The stock value should be output in the form $123.45. Typical output will have the form:

```
PART_CODE PART_DESCRIPTION                      VENDOR                   PART_STOCK STOCK_VALUE
--------- ------------------------------------- ------------------------ ---------- -------------
FR8HDC    Bosch Resistor Spark Plug             30-Supercheap Auto               52       $675.48
GEN123    Rear Tail Light set                   20-Automotive SuperStore         10       $349.50
KCA415    Whiteline Chamber Bolt Kit            20-Automotive SuperStore         27       $776.52
N32780    Protex Brake Shoes                    30-Supercheap Auto               45      $2744.55
```

Typical output 1 (2024) created by Monash Online

[5 marks]

(c) For every part stocked by Monash Automotive list the part code, the part description, the name of the vendor who supplies the part and an indicator if the part has been used (or not used) in a service. The indicator should say 'Used in at least one service' or 'Not used in any service'.

The output should be in part code order.

Typical output would have the form:

```
PART_CODE PART_DESCRIPTION                      VENDOR_NAME           PARTUSAGE
--------- ------------------------------------- --------------------- ----------------------------
DB1849    Bendix Brake pad set                  Automotive SuperStore Not used in any service
FR8HDC    Bosch Resistor Spark Plug             Supercheap Auto       Not used in any service
GEN123    Rear Tail Light set                   Automotive SuperStore Used in at least one service
KCA415    Whiteline Chamber Bolt Kit            Automotive SuperStore Not used in any service
N32780    Protex Brake Shoes                    Supercheap Auto       Used in at least one service
NKR6T1    NGK Standard Spark Plug               Automotive SuperStore Used in at least one service
```

Typical output 2 (2024) created by Monash Online

[8 marks]

(d) For every part stocked by Monash Automotive list the part code, part description, the quantity of these items which have been charged out via a part charge and the total amount of such charges. In arriving at your solution, it is important to note that the current unit cost listed in the part table may not be the price the item was charged out at due to part price variations.

Your output should be listed with the part which has been used the most times first.

Typical output would have the form:

```
PART_CODE PART_DESCRIPTION                      QUANTITY_USED TOTAL_CHARGES
--------- ------------------------------------- ------------- --------------
GEN123    Rear Tail Light set                               2          69.90
341490    Castrol GTX Ultra Clean Engine Oil 5 lt           2          87.70
WA5045    Wesfil Air Filter                                 1         128.00
N32780    Protex Brake Shoes                                1          60.99
ONE2-5    Nulon One Coolant                                 1         158.00
TPS146    Tridon Oil Pressure Sensor                        1         146.95
T23000    Gates Timing Belt                                 1          75.99
```

```
TIM333    Motorkool A/C Compressor                          1         207.00
CTG009    Protorque Injector Pump                           0           0.00
```

Typical output 3 (2024) created by Monash Online

[8 marks]

(e) For all completed services in which the vehicle was ready for pickup later than the customer's requested pickup time, list the customer number, customer name, the service number, the required pickup time, the time the vehicle was ready for pickup and how late the delivery was in hours and minutes in the form 1 hr 15 mins.

The output should show the longest delayed delivery first.

Typical output would have the form (you are required to use the format, output positions and column headings as shown in the following):

```
CUST_NO CUSTOMER NAME  SERV_NO REQUIRED PICKUP TIME READY FOR PICKUP TIME LATE DELIVERY
------- -------------- ------- -------------------- -------------------- -------------
   1000 Andres Syphas      100              10:00 am              11:00 am 1 hr 0 mins
   1040 Angie Eouzan       104              05:00 pm              05:30 pm 0 hr 30 mins
   1050 Butch Japp         105              05:00 pm              05:10 pm 0 hr 10 mins
```

Typical output 4 (2024) created by Monash Online

[8 marks]

(f) List the customer number, customer name and total amount they have been charged for parts across all their completed services where the amount they have been charged is greater than the average amount charged for parts across all completed services for all customers.

For example, on average customers may have been charged $234.56 for all part charges across all their services.

This report will then list those customers whose total part charges across all their services exceeds $234.56 (note these figures are quite small here due to the small amount of data in our model system). The customer with the highest part charges should be listed first. Where two customers have been charged the same total parts charge, order them by the customer's name.

Typical output would have the form:

```
CUST_NO CUST_NAME      TOTAL_PART_PAYMENT
------- -------------- ------------------
   1030 Farrel Grazier         $423.99
   1080 Fredra Doulton         $423.38
   1040 Angie Eouzan           $375.34
```

Typical output 5 (2024) created by Monash Online

[12 marks]

Task 5: Design Modifications (13 marks):

Your answers for these tasks must be placed in the supplied SQL Script **T5-ma-mods.sql**

These tasks should be attempted only after tasks 2, 3 and 4 have been completed. They are to be completed on the 'live' database i.e. the database with the data loaded from your previous work. Also, remember as a live database other users are using the database as you make these changes

In completing this task, you **must**:

- if you need to add new columns, tables or related constraints, follow the naming conventions used in the data models and schema file that have been provided,
- provide column comments for any new columns that you add, and
- correctly manage any transactions used as part of your solution

correctly manage any transactions used as part of your solution.

(a) Monash Automotive (MA) would like to introduce a customer spare parts division where the parts which the business carries, and currently uses for vehicle services, can be sold directly to customers for their personal use (i.e. not connected to a service).

The intention is to issue each customer who makes a spare parts purchase with an invoice of the form shown below. Mock data has been added to illustrate the contents of the form, note, that you are **not required to add data as part of this task**

Monash Automotive Spare Parts Sale Receipt

Customer No: 1456	**Customer Name & Address** Wile E Coyote 123 Desert Chase NeverEnding, Vic 3234	**Sale No:** 567 **Sale Date**: 01/07/2023
Customer Phone 99032626		

Parts Purchased

Part Code	Description	Unit Price	Qty	Amount
NOS999	Nitrous Oxide System	1450.00	1	1450.00
FDB605	Ferodo Brake Pads	45.00	2	90.00
			Total Paid	**1540.00**

MA wish to record, for each sale made, the customer who made the purchase, the sale number (which should be auto-generated by the system, starting with 100), the date of the sale and the total paid for the sale, up to 99999.99. For a given sale, they wish to record for each part sold, the part code, the unit price that was charged (up to 9999.99) and the quantity of the part purchased, up to 99.

Change the database to allow MA to record these, non-service related, customer spare parts sales.

[5 marks]

(b)

Monash Automotive is experiencing an issue where stocks of some parts are allowed to get unreasonably low and as a consequence cause cancellation of service jobs in the workshops.

MA would like to include a re-order level for parts to indicate when a re-order is required. The spare parts staff will close each day with an SQL query which will list those parts which need to be ordered due to being below their re-order level (you do not need to provide this SQL as part of your solution). The output will be used to re-order those parts which have fallen below their restock level during the current day (MA order on a daily basis from their vendors). MA also wish to note the date at which a particular part was restocked, this is normally several days after the part was ordered. Your solution needs to deal with the fact that the parts staff need to be able to work out which part has been reordered already and which part needs ordering based on today's usage.

An analysis of the current part stock levels for all parts has resulted in a decision that this reorder stock level for a particular part should be set initially, for the current parts held, at half of the current stock of the part. For example, if the current stock level of ACFD22 "Sterling Air Cleaner Hose" is 5, the reorder level for ACFD22 should be set at 3 (ie. 2.5 rounded up). For this initial setup, you may set all parts restock date as 01 Jan 2024.

All new parts added to the parts table must be required to provide a re-order stock level which the user entering the data must assign, and a restocked date. This re-order stock level and restocked date are entered manually by the user adding the new part.

Change the database to satisfy this requirement.

[8 marks]

Task 6: Non Relational Database Queries – MongoDB (10 marks)

Your answers for this task (Task 6) must be placed in the supplied sql file **T6-ma-json.sql** and the supplied MongoDB VS Code playground file **T6-ma-mongo.mongodb.js**

(a) Write an SQL statement in **T6-ma-json.sql** to generate a collection of JSON documents using the following structure/format from the Monash Auto tables used in tasks 4 and 5 (code/run this after task 5). Each document in the collection represents a vehicle and contains the service record for that vehicle. The document identifier (_id) is the customer number joined with the vehicles VIN separated by an underscore e.g. '1000_19XFB2E59DE245929':


```
{
    "_id": "1000_19XFB2E59DE245929",
    "customer": {
        "custno": 1000,
        "name": "Andres Syphas",
        "phone": "9571953915"
    },
    "rego_number": "J9Z085",
    "make": "Nissan",
    "model": "Maxima",
    "year": "2019",
    "noservices": 1,
    "booked_services": [
        {
            "servno": 108,
            "servdate": "12-May-2023",
            "labourcost": 350,
            "partcost": 71.28,
            "totalcost": 421.28
        }
    ]
},
{
    "_id": "1000_WBAWR33598P984354",
    "customer": {
        "custno": 1000,
        "name": "Andres Syphas",
        "phone": "9571953915"
    },
    "rego_number": "I6W872",
    "make": "Toyota",
    "model": "Hilux",
    "year": "2018",
    "noservices": 2,
    "booked_services": [
        {
            "servno": 100,
            "servdate": "01-Jan-2023",
            "labourcost": 350,
            "partcost": 117.26,
            "totalcost": 467.26
        },
        {
            "servno": 114,
            "servdate": "01-Jun-2023",
            "labourcost": 350,
            "partcost": 58.99,
            "totalcost": 408.99
        }
    ]
},
```


1000_19XFB2E59DE245929 (2024) created by Monash Online

Note only *sample* partial (two) documents are shown.

[4 marks]

Answer the following in the playground file **T6-ma-mongo.mongodb.js**

You must not add any further comments to the supplied MongoDB script file nor remove/rename any comments (indicated by //)
Only use the solution space to enter your answer - ENSURE every statement ends with a ;

(b) Create a new collection named **services** and insert *all documents generated in Task 6 (a)* above into MongoDB in this collection.

[1 mark]

List all documents which have been inserted

[1 mark]

(c) List the customer number, customer name, rego number and number of services for all vehicles of make "Mazda" which have had more than one service. You may use a different make, other than Mazda, to suit your inserted data to show vehicles with more than one service.

[2 marks]

(d) Customer number '1040' who currently owns the vehicle identified by vin '5XXGM4A74DG668202' has just had a new service completed.

1. Use an appropriate db.find command which shows only the services for this customer and vehicle to show their service record before this service is added.
2. The service number was 2000, the date serviced '12-MAR-2024', the labour cost was 501.10, the parts cost 123.45 and the total cost was 624.55.
3. Write the necessary MongoDB commands to add this new service into the collection.
4. Use an appropriate db.find command which shows only the services for this customer and vehicle after making the change so that you show/confirm the change that was made.

[2 marks]

Supporting resources

- Citation using APA 7th ed. style (Monash University Library, 2020).

Monash University Library has further information about referencing on their Citing and referencing page.

Submission details overview

Please note, that if you need to resubmit, you cannot depend on staff availability, for this reason, please be very careful with your submission. It is strongly recommended that you submit several hours before this time to avoid such issues.

For this assessment there are seven files you are **required** to submit:

- T1-ma-ra.pdf
- T2-ma-insert.sql
- T3-ma-dml.sql
- T4-ma-select.sql
- T5-ma-mods.sql
- T6-ma-json.sql
- T6-ma-mongo.mongodb.js

If you need to make any comments to your marker please place them at the head of each of your solution scripts/answers in the 'Comments for your marker' section.

Do not zip these files into one zip archive, submit seven independent SQL scripts.

Note: Late submission will incur penalties at the rate of -4.5 marks for every 24 hours (or part thereof) the submission is late.

The seven files must also exist in your FITGitLab server repo and show a clear history of development (a minimum of two pushes per file).

Please note *we cannot mark any work on the GitLab Server*, you need to ensure that you submit correctly via Moodle since it is only in this process that you complete the required student declaration without which work cannot be assessed.

It is your responsibility to ensure that the files you submit are the correct files—we strongly recommend after uploading a submission, and prior to submitting, that you download the submission and double-check its contents.

Submission status

Your assessment MUST show a status of '**Submitted for grading**' before it will be marked.

If your submission still shows a status of '**Draft (not submitted)**', it will **not** be assessed and will incur late penalties after the due date/time.

Assessment criteria

Where SQL is involved there are often several alternative approaches possible, such alternatives will be graded based on the code successfully meeting the brief's requirements. If it does, the answer will be accepted and graded appropriately—efficiency will not be graded.

Submissions will be graded on the following:

- Task 1: Relational Algebra Queries (6 marks)
- Task 2: Data Insert (10 marks)
- Task 3: Data Manipulation (10 marks)
- Task 4: SQL Queries (46 marks)
- Task 5 Design Modifications (13 marks)
- Task 6 Non Relational Database Queries – MongoDB (10 marks)
- Correct use of Git (5 marks)

Assessment 2 marking criteria

Marking Criteria	**Items Assessed**
Task 1 Relational Algebra Queries 6 marks	
	Maximum 6 marks - Satisfy brief requirements: • Marks awarded, as listed, (a)–(c) for relational algebra which meets the expressed requirement • Marks awarded for correct use of RA operations and the use of appropriate symbols • Mark penalty applied if your answer does not show an understanding of query efficiency i.e. you must not make use of unnecessary joins, nor carry attributes and tuples up through the query which are not necessary • 0 marks will be awarded if SQL select statements are provided as solutions for Task 1
Task 2 Data Insert 10 marks	
Insert of required	**Maximum 5 marks - Insert of data:**

items of test data	• Marks awarded for correct insert of required data • Marks awarded for correct management of transactions
Insert valid test data	**Maximum 5 marks - Valid data inserted:** • Marks awarded for validity of data inserted ◦ meets the requirements expressed in the assessment brief • Marks awarded for correct management of dates when inserting
Task 3 Data Manipulation 10 marks	
	Maximum 10 marks - Satisfy brief requirements: • Marks awarded (a) - (c) for SQL code which meets the expressed requirement

Assessment declaration and statement of authorship

By submitting my assessments I declare that:

- This is an original piece of work and no part has been completed by any other person.
- I have read and understood the guidelines on the Assessment and academic integrity policy, and no part of this work has been copied or paraphrased from any other source except where this has been clearly acknowledged in the body of the assessment and included in the reference list.
- I have retained a copy of this assessment in the event of it becoming lost or damaged.

I agree and acknowledge that:

- I have read and understood the declaration and statement of authorship.
- I accept that use of my Monash Online account to electronically submit this assessment constitutes my agreement to the Monash Online Assessment Declaration.
- If I do not agree to the Monash Online Assessment Declaration in this context, the outcome of my assessment may not be valid for assessment purposes and may not be included in my aggregate score for this unit.
- I am aware that it is not acceptable to resubmit the same piece of work (in part or as whole) for multiple assessments without permission from the Unit Coordinator.

Prior to submitting your assessment, please refer to the Academic integrity Monash Library modules to ensure the integrity of your submission.

Further information relating to the penalties for plagiarism, which range from a formal caution to expulsion from the University, is contained on the Academic integrity and what plagiarism, collusion and cheating mean page.