### DATABASE FOUNDATIONS FOR BUSINESS ANALYTICS (BUAN 6320)

**GROUP 14** 

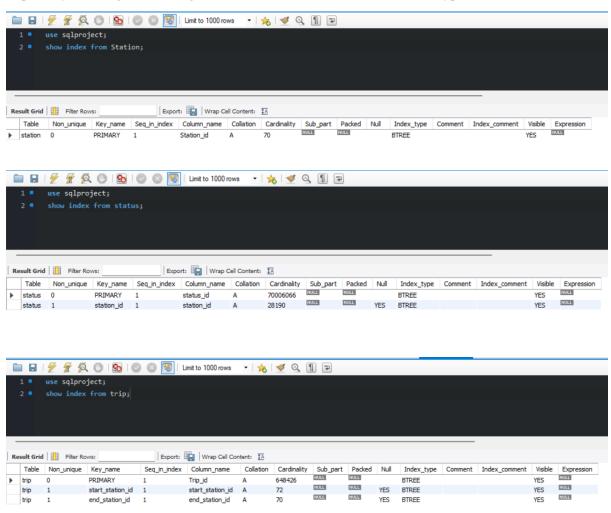
**Project 2 Report** 

Submitted By: Sahejbir Singh Kumar
Indrajeet S Thakare
Bala Krishna Bobbili
Mano Snigdha Devara
Shreya Shamarthi
Harshitha Reddy Nandikonda

### **Part 1: Indexing and Query Timing**

# 1.1. List all the current indexes in your database and the columns they are associated with along with the index type.

In primary indexing, the index gets stored into B-Trees. Hence, the index type is B-Trees.



## 1.2. Explain what is in common between these columns (why these columns are indexed automatically by the database management system).

Database management system follows Primary Indexing. Indexes have been automatically created for all the columns in out database. The primary key is automatically indexed because the primary key, index gets stored into B-Trees. Since the primary keys are unique, database management system manages uniqueness by Primary Key Indexing.

1.3. Make a copy of your database and delete all the indexes there (you might need to delete foreign keys before you can delete some of the indexes) – now you have two databases: database A with indexes and database B without any indexes.

We created a new database (Database B) and copied the data.

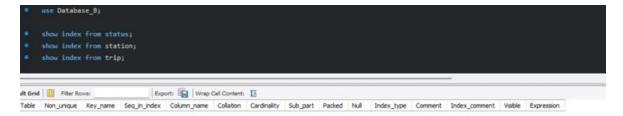
```
create table status as (select * from database_a.status);
create table station as (select * from database_a.station);
create table trip as (select * from database_a.trip);
```

We dropped the indexes from the newly created database (Database B) using:

DROP INDEX 'PRIMARY' ON STATUS.

Dropped indexes similarly in trip and station table.

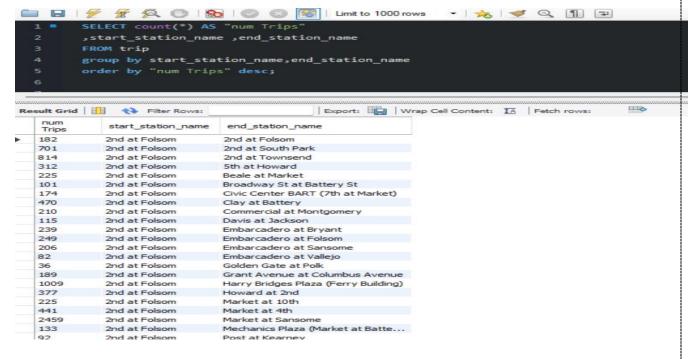
Below is the snapshot of database B relations having no indexes.



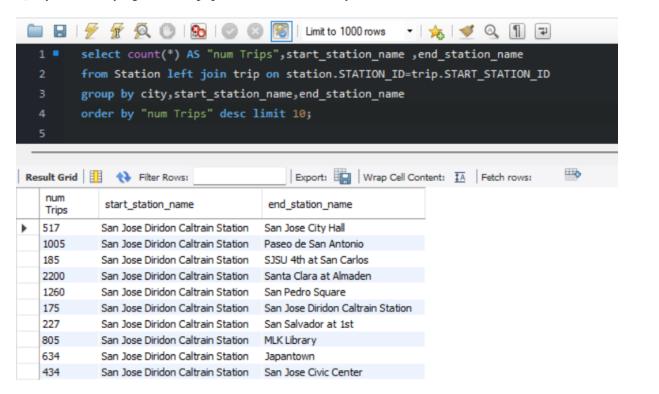
Now we have two same databases – database A (with indexes) and database B (without indexes).

### 1.4. Write at least 5 queries (with JOINs between your tables).

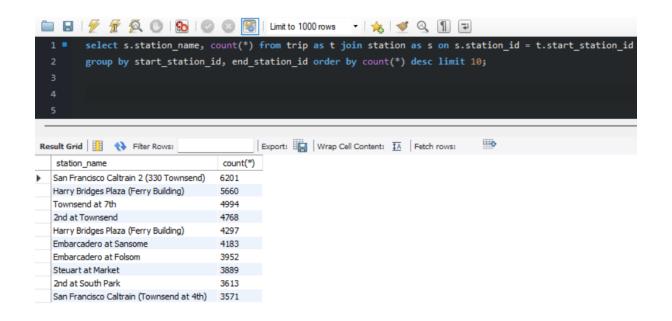
Query I: Identifying the most popular routes.



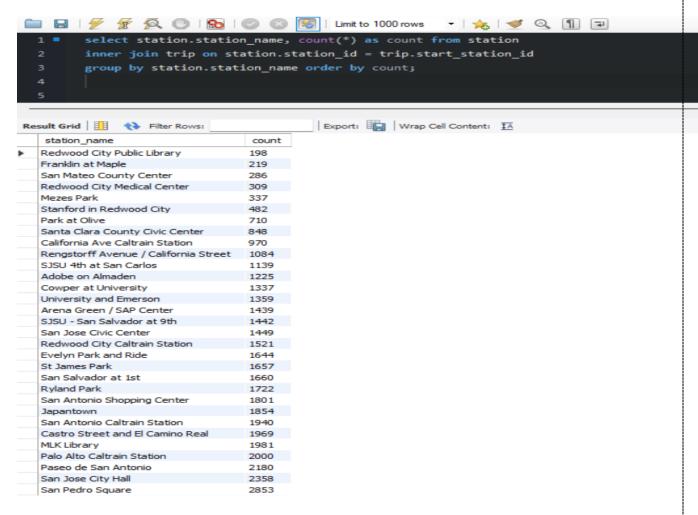
Query II: Identifying the most popular routes in each city.



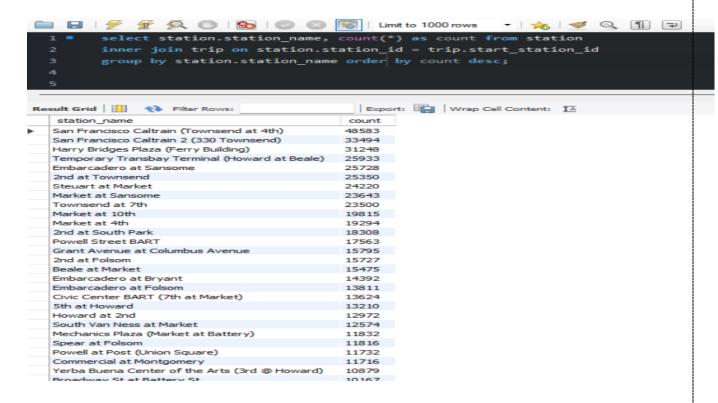
Query III: Identifying the stations from where most trips started.



Query IV: Identifying the stations that have been preferred the least number of times by customers

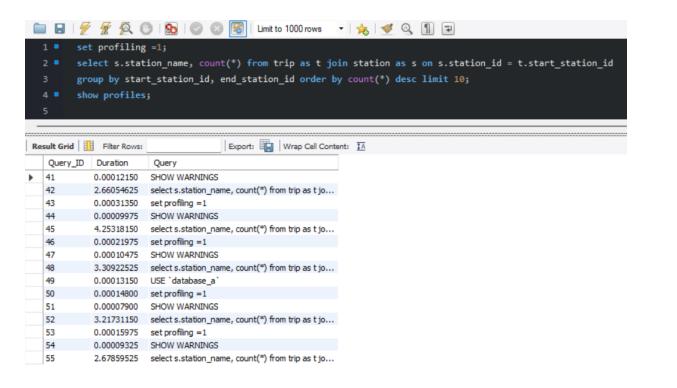


Query V: Identifying the stations that have been preferred the most number of times by customers.



# 1.5. Execute and time these queries on both databases and report your findings (repeat timing for each query at least 10 times and average the times).

Queries have been timed as per the below command:



On running these on all queries we get the below average query runtimes:

DATABASES	A	В
QUERY I	8.54602050	11.35684654
QUERY II	3.76957150	5.71239100
QUERY III	4.00908500	4.45617350
QUERY IV	4.21591450	5.00045725
QUERY V	3.57542004	4.14751100

We can clearly see that the database having indexes has faster query runtime.

### 1.6. Select some columns from database A (columns that are not already indexed) and create index on them.

We have created indexes in Database A. Snapshots have been provided below:

```
41 * create index city_idx on station (city);
42 * create index bikes_available_idx on status (bikes_available);
43 * create index subscription_type_idx on trip (subscription_type);
44
```

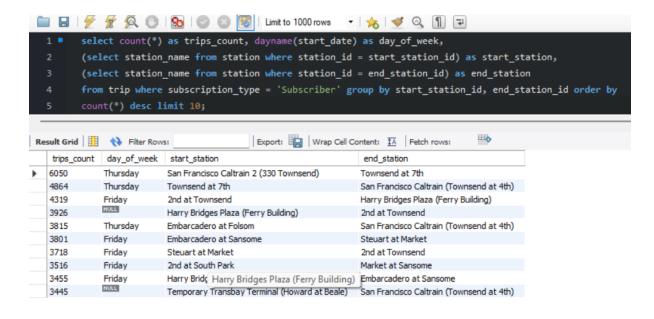
city has been indexed from station table.

bikes available has been indexed from status table.

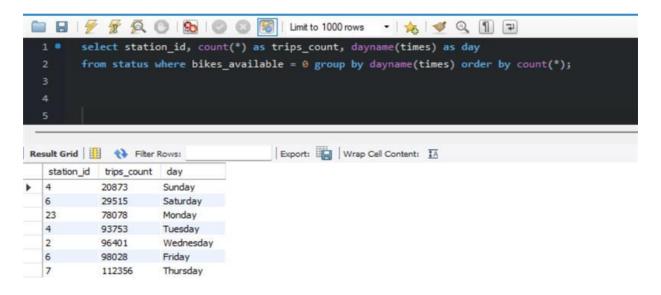
subscription\_type has been indexed from trip table.

### 1.7. Write a query for each column – the query should include the column in the WHERE clause in a condition.

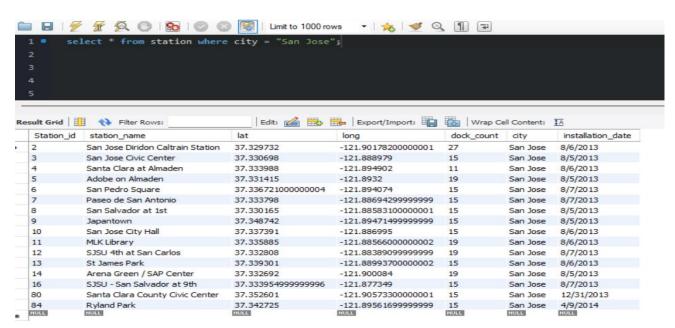
Query I: Identifying which day of the week has the highest number of trips and the start and end station associated with it.



Query II: Identifying the busiest station id for the whole week.



Query III: Getting the details of stations for San Jose city.



# 1.8. Execute and time these queries on both databases and report your findings (repeat timing for each query at least 10 times and average the times).

Following the steps done in 1.5, we got the runtime of the "where" queries. Below are the results:

DATABASE	A	В
QUERY I	0.78401250	1.28210207

QUERY II	29.21545304	55.73942405
QUERY III	0.00351800	0.00853052

### 1.9. Make a conclusion based on your findings in this part.

We can clearly see the advantage of indexes in as per our observations from the two databases – one with indexes and vice-versa. Using indexes results in query optimization i.e indexes make execution of queries much faster. Runtime for queries for database A is much less than database B. This is due to the absence of indexes in database B. Indexes expedited the query execution in database A.

### Part 2: MongoDB and MQL

### 2.1. Explore your dataset and familiarize yourself with the dataset and its content.

We have chosen the dataset – PakWheels. The dataset contains Pakistan's largest PakWheels automobile listings. We have explored the dataset and its content and got familiar with it.

# 2.2. Explain why it is better to use non-relational databases such as MongoDB to work with such a dataset (explain in the context of your dataset).

Non-relational databases are better to use for unstructured data, they are suitable for both operational and transactional data. This is because non-relational databases eliminated the limitations that come with relational databases like scalability, flexibility, and performance issues. MongoDB is a non-relational database that offers scalability, high performance, reliability, and flexibility. Data is stored in the form of documents. There are no tables (rows and columns), the data varies from record to record.

In our dataset, let us consider the "features" variable which is an array. One record has contained an array of 4 features, and one has an array of 12 features. This is the flexibility provided in the non-relational databases. Also, variable "bodyType" is not present in all records. This is not possible in relational databases.

```
v features: Array
                                               0: "AM/FM Radio"
                                               1: "Air Bags"
                                               2: "Air Conditioning"
                                               3: "Alloy Rims"
                                               4: "Cassette Player"
                                               5: "Keyless Entry"
v features: Array
                                               6: "Power Locks"
    0: "AM/FM Radio"
                                               7: "Power Mirrors"
    1: "Alloy Rims"
                                               8: "Power Steering"
    2: "Cassette Player"
                                               9: "Power Windows"
    3: "Immobilizer Key"
                                               10: "Sun Roof"
```

Another advantage of non-relational databases is they allow sub-documents. In our dataset, myriad variables like "brand", "vehicleEngine", "extraFeatures" have sub-documents which contain their own information (like nested documents). This is not possible in relational databases.

## 2.3. Import your dataset into MongoDB using the following process (do not use MongoDB Compass as it has a 16MB limit for JSON files).

The instructions have been followed to import the dataset.

## 2.4. List some of the attributes (field/properties) of your database which are common among all documents.

Some attributes which are common in all documents – "itemCondition , "model", "manufacturer", "fuelType", "sellerLocation" "name", "decription", modelDate", "color" etc.

### 2.4.1. For these fields, provide some of the values they contain in the database.

```
db.PakWheels.distinct("model")
 '1 Series',
                                '300 C',
 '250 D'.
                   '3 Series',
 '370Z',
                                                 db.PakWheels.distinct("modelDate")
                   'A Class',
                               'A6',
                   'AD Van'.
                                                     1963, 1964, 1965, 1966, 1967, 1968, 1969,
                   'Airwave',
                                'Allion',
                   'Alto Lapin', 'Anglia',
                                                     1991, 1992, 1993, 1994, 1995, 1996, 1997,
 'Autobiography',
                  'Avanza',
                                'Avensis',
                   'B2200',
                                'BJ40',
 'Azwagon',
                                                     2005, 2006, 2007, 2008, 2009, 2010, 2011,
                                                     2012, 2013, 2014, 2015, 2016, 2017, 2018,
 'Beetle',
                   'Bego',
                                'Belta',
 'Besturn',
```

```
db.PakWheels.distinct("itemCondition")

{ [ 'used' ]
} db.PakWheels.distinct("fuelType")

{ [ 'CNG', 'Diesel', 'Electric', 'Hybrid', 'Lpg', 'Petrol' ]
} db.PakWheels.distinct("@type")

{ [ 'Car' ]
} db.PakWheels.distinct("vehicleTransmission")

{ [ 'Automatic', 'Manual' ]
```

```
db.PakWheels.distinct("color")
                                                      db.PakWheels.distinct("manufacturer")
                                                         'BMW',
                                                                      'Bentley',
                                                                                    'Cadillac',
                                                                                                   'Changan',
                                                         'Chery',
                                                                                                   'Citroen',
  '14/ 16',
                                                                                    'Daehan',
                                                                                                   'Daewoo',
  '3 shaded german colour',
                                                                                                   'Dongfeng',
                                                         'Daihatsu',
                                                                      'Datsun',
                                                                                    'Dodge',
  'AQ Jade',
                                                                      'Fiat',
                                                                                                   'GMC',
  'AQ.JADE',
                                                         'Geely',
                                                                                                   'Hino',
  'AQUA BLUE',
                                                         'Honda',
                                                                      'Hummer',
                                                                                   'Hyundai',
                                                                                                   'Isuzu',
  'AS YOU CAN SEE',
  'AUTOMATIC '.
                                                         'JAC',
                                                                      'JW Forland', 'Jeep',
  'Ac installed working',
                                                         'Land Rover',
                                                                                                   'MINI',
                                                         'Master',
                                                                                    'Mercedes Benz', 'Mitsubishi',
                                                         'Nissan',
                                                                      'Others',
                                                                                                   'Plymouth',
  'Any colour ',
  'Aqua Blue',
                                                         'Porsche',
                                                                      'Prince',
                                                                                                   'Range Rover',
  'Aqua Marine',
                                                                      'Roma',
                                                                                    'Saab',
                                                                                                   'Skoda',
                                                                                                   'Suzuki',
                                                         'Sogo',
                                                                                                   'Vauxhall',
                                                         'Volkswagen', 'Volvo',
  'Ash White',
  'Attitude Black'
```

### 2.5. List some of the attributes (field/properties) of your database which are not common among all the documents.

There are two variables which are not common in all the documents – "bodyType" and "features". "bodyType" is present in only few documents and "features" has different number of elements stored as an array among different documents.

### 2.5.1. For these fields, provide some of the values they contain in the database.

```
db.PakWheels.distinct("bodyType")
⟨[
                                         'AM/FM Radio',
                 'Compact sedan',
   'Compact SUV',
                      'Coupe',
   'Convertible',
                                         'CD Player',
                      'Double Cabin',
   'Crossover',
                                        'Cassette Player',
   'Hatchback',
                      'High Roof',
                                         'CoolBox',
                                         'Cruise Control',
   'MPV',
                      'Micro Van',
                                         'DVD Player',
                                         'Front Camera',
   'Mini Van',
                      'Mini Vehicles',
                                         'Front Speakers',
                      'Pick Up',
                                         'Heated Seats',
   'Off-Road Vehicles',
   'SUV',
                      'Sedan',
                                         'Keyless Entry',
                                         'Navigation System',
   'Single Cabin', 'Station Wagon',
                                         'Power Locks',
                                         'Power Mirrors',
   'Subcompact hatchback', 'Truck',
                                         'Power Steering',
   'Van'
                                         'Rear AC Vents',
                                         'Rear Seat Entertainment
```

# 2.6. Write at least 5 queries using the key-value pairs you found in the previous steps to narrow down the result (provide the queries and results in your report).

Query I: Black coloured automobiles with price greater than 1000000, and body type as SUV or Compact SUV.

```
db.PakWheels.find({'price': {$gt: 1000000},

$or: [{'extraFeatures.BodyType': 'SUV'}, {'extraFeatures.BodyType':
'Compact SUV'}],
'extraFeatures.Color': 'Black'}).limit(2).pretty()
```

Output:

Query II: Black or White coloured automobiles (cars) of either Suzuki or Toyota brand with seller location - Hyderabad.

```
db.PakWheels.find({'sellerLocation': {$regex: 'Hyderabad'},
$or: [{'extraFeatures.Color':'White'}, {'extraFeatures.Color':'Black'}],
$or: [{'brand.name':'Suzuki'}, {'brand.name':'Toyota'}]}).limit(3).pretty()
```

#### Output:

Query III: Automobiles released after 2010, fuel type being either Electric or Hybrid.

```
db.PakWheels.find({'modelDate': {$gt: 2010},
$or: [{'fuelType': 'Hybrid'}, {'fuelType': 'Electric'}]}).limit(5).pretty()
```

#### Output:

Query IV: Unregistered automobiles which are posted from website and having the following extra features – unregistered, assembly: imported.

```
db.PakWheels.find({'postedFrom':'Added via Website',
   'extraFeatures.RegisteredIn':'Un-
Registered','extraFeatures.Assembly':'Imported'}).limit(1).pretty()
```

#### Output:

Query V: Manual automobiles (cars) launched before 2010 with fuel type being one of them – Petrol/Diesel/LPG.

#### Output:

## 2.7. Write at least 5 update queries to update some of the values in your database (provide the queries and results in your report).

Query I: Update one car color to Red, for Suzuki, models from one of Mehran/MR Wagon/Wagon R, having Grey color.

```
db.PakWheels.updateOne({'manufacturer': 'Suzuki', 'color': 'Grey', $or:
   [{'model': 'Mehran'}, {'model': 'MR Wagon'}, {'model': 'Wagon R'}]
}, {$set: {'color': 'Red'}})
```

#### Output:

```
> db.PakWheels.updateOne({'manufacturer': 'Suzuki','color': 'Grey', Sor: [{'model': 'Mehran'}, {'model': 'MR Wagon'}, {'model': 'Wagon R'}]
}, {Sset: {'color': 'Red'}})
{{ acknowledged: true,
   insertedId: null,
   matchedCount: 1,
   modifiedCount: 1,
   upsertedCount: 0 }
```

Query II: Update the "Posted From" information from NA/NULL to "Added via Phone" for Suzuki or Toyota cars.

Output:

Query III: For cars sold in Islamabad with imported assembly and engine displacement of 3500cc, update the assembly to Local.

```
db.PakWheels.updateOne({'sellerLocation': {$regex: 'Islamabad'},
   'vehicleEngine.engineDisplacement':'3500cc',
   'extraFeatures.Assembly':'Imported'}, {$set:
   {'extraFeatures.Assembly': 'Local'}})
```

Output:

```
db.PakWheels.updateOne({'sellerLocation': {$regex: 'Islamabad'},
   'vehicleEngine.engineDisplacement':'3500cc', 'extraFeatures.Assembly':'Imported'}, {$set:
   {'extraFeatures.Assembly': 'Local'}})

{   acknowledged: true,
   insertedId: null,
   matchedCount: 1,
   modifiedCount: 1,
   upsertedCount: 0 }
```

Query IV: Unset the body type for cars having fuel type - CNG, Engine Displacement – 1500cc, prices – greater than 1000000

```
db.PakWheels.updateMany({'fuelType': 'CNG',
  'vehicleEngine.engineDisplacement': '1500cc', 'price': {$gt:
  1000000}}, {$unset: {'bodyType': 1}})
```

Output:

```
db.PakWheels.updateMany({'fuelType': 'CNG', 'vehicleEngine.engineDisplacement': '1500cc', 'price': {$gt:
1000000}}, {$unset: {'bodyType': 1}})

{    acknowledged: true,
    insertedId: null,
    matchedCount: 9,
    modifiedCount: 9,
    upsertedCount: 0 }
```

Query V: For Toyota cars released before 2010 having automatic transmission, set the vehicle transmission to manual.

```
db.PakWheels.updateMany({'modelDate': {$1t: 2010}, 'vehicleTransmission':
   'Automatic', 'brand.name':
   'Toyota'}, {$set: {'vehicleTransmission': 'Manual'}})
```

Output:

```
> db.PakWheels.updateMany({'modelDate': {$1t: 2010}, 'vehicleTransmission': 'Automatic', 'brand.name':
    'Toyota'}, {$set: {'vehicleTransmission': 'Manual'}})

< { acknowledged: true,
    insertedId: null,
    matchedCount: 3389,
    modifiedCount: 3389,
    upsertedCount: 0 }</pre>
```

## 2.8. Write at least 5 queries to insert new documents into your database (provide the queries and results in your report).

#### Query I:

#### Query II:

```
description of the process of t
```

#### Query III:

```
db.PakWheels.insertOme((
    "@type": "Car",
    "brand": ("@type": "Brand", "name": "Mazda"),
    "model1:"CS 5",
    "itemCondition": "used",
    "model1:"CS 5",
    "itemCondition": "used",
    "modelbate": 2017,
    "manufacturer": "Mazda",
    "fuelType": "CNC",
    "wehicleTransmission": "Manual",
    "color": "Grey",
    "bodyType": "Compact Sedan",
    "wehicleEngine": ("@type": "EngineSpecification", "engineDisplacement": "1000cc"), "mileageFromOdometer": "36,000 km",
    "postedFrom": "Added via Website",
    "extraFeatures": ("RegisteredIn": "Jauharabad", "Color": "Grey", "Assembly": "Imported", "EngineCapacity": "1300 cc", "BodyType": "Compact Sedan", "LastUpdated: ": "Mar 16, 2017", "AdRef#": "4937978"
    "features": ("RegisteredIn": "Jauharabad", "Color": "Grey", "Assembly": "Imported", "EngineCapacity": "1300 cc", "BodyType": "Compact Sedan", "LastUpdated: ": "Mar 16, 2017", "AdRef#": "4937978"
    "features": ("RegisteredIn": "Jauharabad", "Color": "Grey", "Assembly": "Imported", "EngineCapacity": "1300 cc", "BodyType": "Compact Sedan", "LastUpdated: ": "Mar 16, 2017", "AdRef#": "4937978"
    "features": ("RegisteredIn": "Jauharabad", "Color": "Grey", "Assembly": "Imported", "EngineCapacity": "1300 cc", "BodyType": "Compact Sedan", "LastUpdated: ": "Mar 16, 2017", "AdRef#": "4937978"
    "features": ("RegisteredIn": "Jauharabad", "Color": "Grey", "Assembly": "Imported", "EngineCapacity": "1300 cc", "BodyType": "Compact Sedan", "LastUpdated: ": "Mar 16, 2017", "AdRef#": "4937978"
    "features": ("RegisteredIn": "Jauharabad", "Color": "Grey", "Assembly": "Imported", "EngineCapacity": "1300 cc", "BodyType": "Compact Sedan", "LastUpdated: ": "Mar 16, 2017", "AdRef#": "4937978"
    "features": ("Assembly ": "LastUpdated: ": "Mar 16, 2017", "AdRef#": "4937978"
    "features": "Grey", "Assembly ": "Imported", "EngineCapacity": "1300 cc", "BodyType": "Compact Sedan", "LastUpdated: ": "Mar 16, 2017", "AdRef#": "4937978"
    "features": ("Assembly ": "LastUpdated: ": "Mar 16, 2017", "AdRef#": "4937
```

#### Query IV:

```
db.PakWheels.insertOne({
    "@type": "Car",
    "brand": ("@type": "Brand", "name": "Nissan"),
    "model": "Sunny",
    "itemCondition": "used",
    "modelDate": 2012,
    "manufacturer": "Nissan",
    "fuelType": "Petrol",
    "vehicleTransmission": "Manual",
    "color": "Black",
    "bodyType": "Sedan",
    "vehicleEngine": ("@type": "EngineSpecification", "engineDisplacement": "1498cc"},
    "extraPeatures": ("RegisteredIn": "Hyderabad", "Color": "Black", "Assembly": "Imported", "EngineCapacity": "1498 cc", "AdRef#": "8136978"},
    "features": ("Rutomatic Climate Control", "Air Bags", "Anti Lock Braking System"],
    "price": 5914000,
    "priceCurrency": "PKR"
})

{    acknowledged: true,
    insertedId: ObjectId("639717c493649e47e756b611") }
```

#### Query V:

## 2.9. Write at least 5 delete queries to remove documents from your database (provide the queries and results in your report).

#### Query I:

```
> db.PakWheels.deleteOne({"color":"Grey"})

< { acknowledged: true, deletedCount: 1 }</pre>
```

#### Query II:

```
> db.PakWheels.deleteMany({$or:[{"manufacturer":"Audi"},{"manufacturer":"Nissan"}]})
< { acknowledged: true, deletedCount: 1256 }</pre>
```

#### Query III:

```
db.PakWheels.deleteOne({'modelDate': {$lt:2010},'vehicleTransmission':'Manual', 'brand.name': 'Suzuki'})

{ acknowledged: true, deletedCount: 1 }
```

#### Query IV:

```
> db.PakWheels.deleteMany({"bodyType":"Sedan"})

< { acknowledged: true, deletedCount: 22123 }</pre>
```

### Query V:

```
b db.PakWheels.deleteMany({'postedFrom': 'Added via Phone',$or: [{'brand.name':'Suzuki'}, {'brand.name': 'Toyota'}], 'features':[]})

{ acknowledged: true, deletedCount: 2 }
```