

# PRESENTATION OF FIRST ASSIGNMENT

---

DONE BY BORIS MENSHIKOV



# TARGETS:

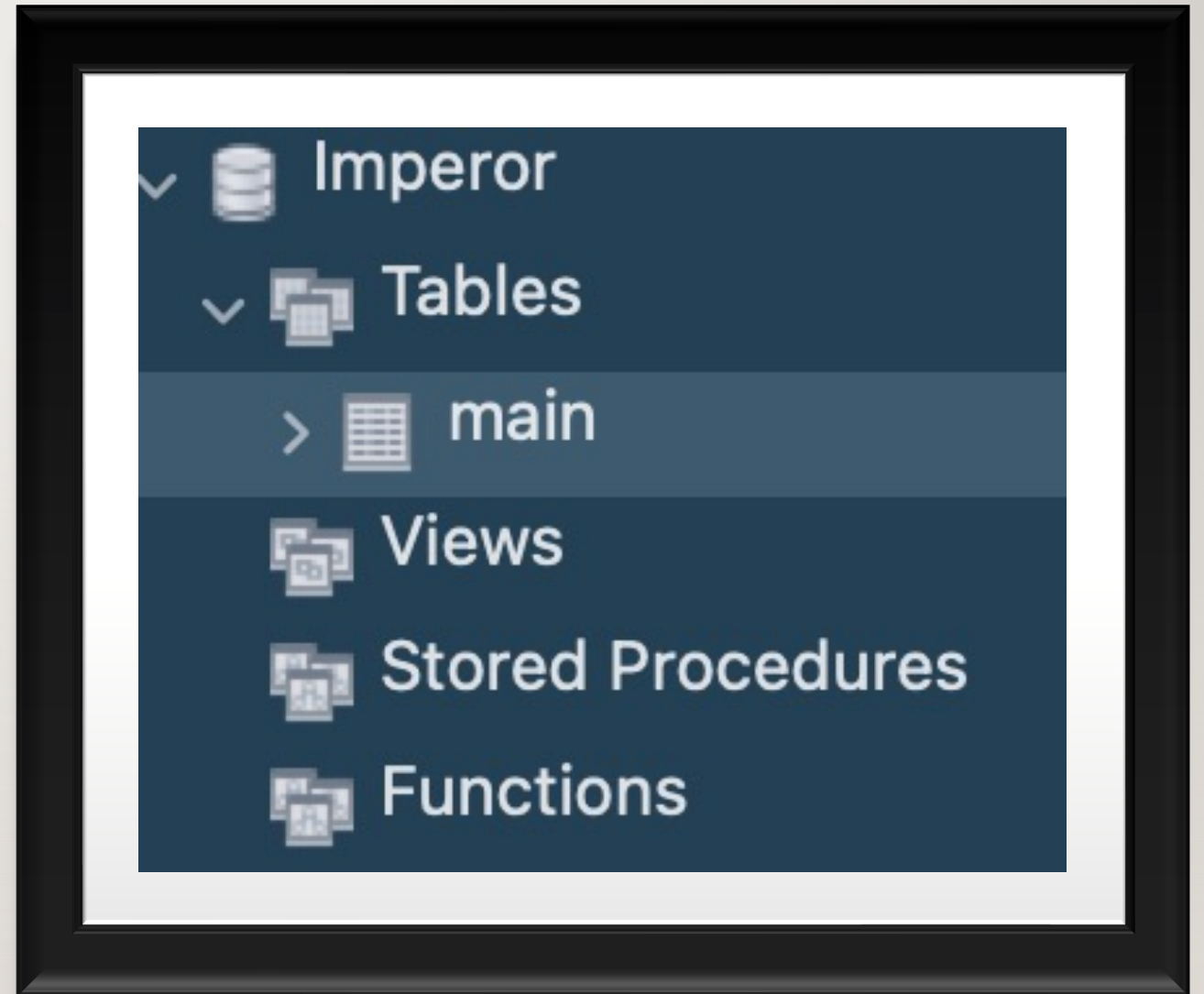
---

- 1) clean and import data to the workshop
- 2) Create 3 tables with information
- 3) provide EER Diagram
- 4) 7 unique tasks associated with tables

# INTRODUCTION

---

- First of all I created a database called “Imperator” then I imported the cvs File with all the information and created a table “main”.



# IMPORTING DATA

---

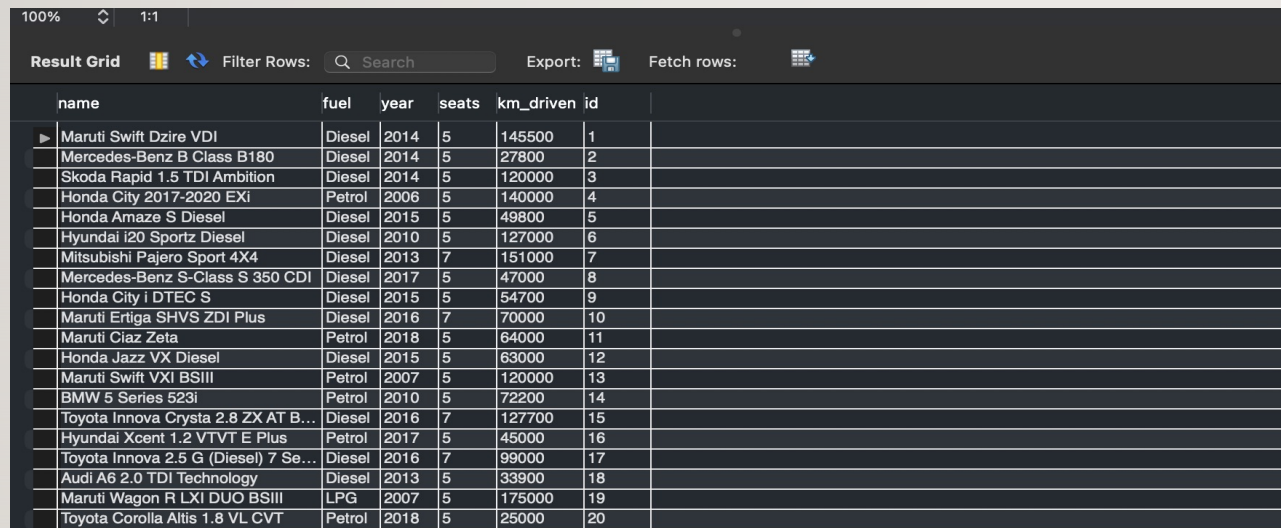
- To make the process of importing data easier I created the column Id in the main Table.
- Then I started creating tables.

```
ALTER TABLE main  
ADD COLUMN id INT PRIMARY KEY AUTO_INCREMENT;
```



# FIRST TABLE

The result:



The screenshot shows a database application interface with a table of vehicle information. The table has columns for name, fuel, year, seats, km\_driven, and id. The data is as follows:

	name	fuel	year	seats	km_driven	id
▶	Maruti Swift Dzire VDI	Diesel	2014	5	145500	1
▶	Mercedes-Benz B Class B180	Diesel	2014	5	27800	2
▶	Skoda Rapid 1.5 TDI Ambition	Diesel	2014	5	120000	3
▶	Honda City 2017-2020 EXi	Petrol	2006	5	140000	4
▶	Honda Amaze S Diesel	Diesel	2015	5	49800	5
▶	Hyundai i20 Sportz Diesel	Diesel	2010	5	127000	6
▶	Mitsubishi Pajero Sport 4X4	Diesel	2013	7	151000	7
▶	Mercedes-Benz S-Class S 350 CDI	Diesel	2017	5	47000	8
▶	Honda City i DTEC S	Diesel	2015	5	54700	9
▶	Maruti Ertiga SHVS ZDI Plus	Diesel	2016	7	70000	10
▶	Maruti Ciaz Zeta	Petrol	2018	5	64000	11
▶	Honda Jazz VX Diesel	Diesel	2015	5	63000	12
▶	Maruti Swift VXI BSIII	Petrol	2007	5	120000	13
▶	BMW 5 Series 523i	Petrol	2010	5	72200	14
▶	Toyota Innova Crysta 2.8 ZX AT B...	Diesel	2016	7	127700	15
▶	Hyundai Xcent 1.2 VTVT E Plus	Petrol	2017	5	45000	16
▶	Toyota Innova 2.5 G (Diesel) 7 Se...	Diesel	2016	7	99000	17
▶	Audi A6 2.0 TDI Technology	Diesel	2013	5	33900	18
▶	Maruti Wagon R LXI DUO BSIII	LPG	2007	5	175000	19
▶	Toyota Corolla Altis 1.8 VL CVT	Petrol	2018	5	25000	20

The code I used to create the table:

```
CREATE TABLE vehicle_information
AS SELECT
    name,
    fuel,
    year,
    seats,
    km_driven,
    id
FROM main;
```

This table Use for Basic Vehicle Information

# SECOND TABLE

The Result:

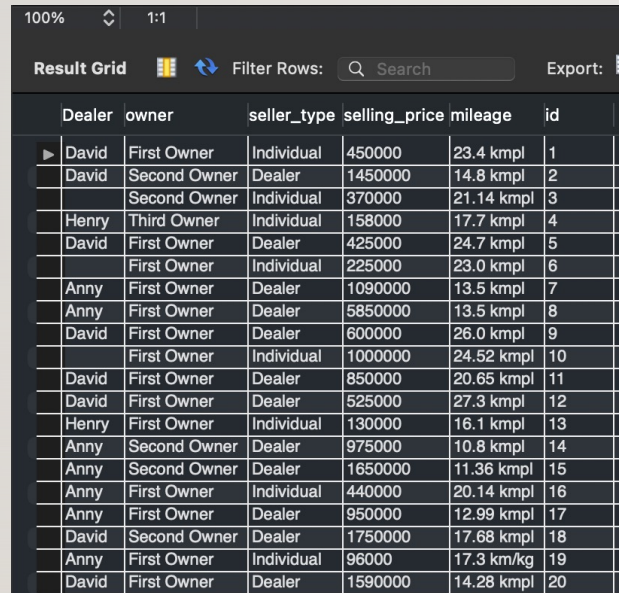
Result Grid						Filter Rows:	Search	Export:
	transmission	engine	max_power	torque	id			
▶	Manual	1248 CC	74 bhp	190Nm@ 2000rpm	1			
	Automatic	2143 CC	120.7 bhp	200Nm@ 1250-4000rpm	2			
	Manual	1498 CC	103.52 bhp	250Nm@ 1500-2500rpm	3			
	Manual	1497 CC	78 bhp	12.7@ 2,700(kgm@ rpm)	4			
	Manual	1498 CC	98.63 bhp	200Nm@ 1750rpm	5			
	Manual	1396 CC	90 bhp	22.4 kgm at 1750-2750rpm	6			
	Manual	2477 CC	175.56 bhp	400Nm@ 2000-2500rpm	7			
	Automatic	2987 CC	282 bhp	490Nm@ 1600rpm	8			
	Manual	1498 CC	98.6 bhp	200Nm@ 1750rpm	9			
	Manual	1248 CC	88.5 bhp	200Nm@ 1750rpm	10			
	Manual	1462 CC	103.25 bhp	138Nm@ 4400rpm	11			
	Manual	1498 CC	98.6 bhp	200Nm@ 1750rpm	12			
	Manual	1298 CC	88.2 bhp	11.5@ 4,500(kgm@ rpm)	13			
	Automatic	2497 CC	150 bhp	250 Nm at 2750 rpm	14			
	Automatic	2755 CC	171.5 bhp	360Nm@ 1200-3400rpm	15			
	Manual	1197 CC	81.86 bhp	113.75nm@ 4000rpm	16			
	Manual	2494 CC	100.6 bhp	200Nm@ 1200-3600rpm	17			
	Automatic	1968 CC	174.33 bhp	380Nm@ 1750-2500rpm	18			
	Manual	1061 CC	57.5 bhp	7.8@ 4,500(kgm@ rpm)	19			
	Automatic	1798 CC	138.03 bhp	173Nm@ 4000rpm	20			

The code I used to create the table

```
CREATE TABLE vehicle_consumption
AS SELECT
    transmission,
    engine,
    max_power,
    torque,
    id
FROM main;
```

# THIRD TABLE

The Result:



	Dealer	owner	seller_type	selling_price	mileage	id
▶	David	First Owner	Individual	450000	23.4 kmpl	1
	David	Second Owner	Dealer	1450000	14.8 kmpl	2
		Second Owner	Individual	370000	21.14 kmpl	3
	Henry	Third Owner	Individual	158000	17.7 kmpl	4
	David	First Owner	Dealer	425000	24.7 kmpl	5
		First Owner	Individual	225000	23.0 kmpl	6
	Anny	First Owner	Dealer	1090000	13.5 kmpl	7
	Anny	First Owner	Dealer	5850000	13.5 kmpl	8
	David	First Owner	Dealer	600000	26.0 kmpl	9
		First Owner	Individual	1000000	24.52 kmpl	10
	David	First Owner	Dealer	850000	20.65 kmpl	11
	David	First Owner	Dealer	525000	27.3 kmpl	12
	Henry	First Owner	Individual	130000	16.1 kmpl	13
	Anny	Second Owner	Dealer	975000	10.8 kmpl	14
	Anny	Second Owner	Dealer	1650000	11.36 kmpl	15
	Anny	First Owner	Individual	440000	20.14 kmpl	16
	Anny	First Owner	Dealer	950000	12.99 kmpl	17
	David	Second Owner	Dealer	1750000	17.68 kmpl	18
	Anny	First Owner	Individual	96000	17.3 km/kg	19
	David	First Owner	Dealer	1590000	14.28 kmpl	20

The code used to create the table

```
CREATE TABLE vehicle_patern
AS SELECT
    Dealer,
    owner,
    seller_type,
    selling_price,
    mileage,
    id
FROM main;
```



```
ALTER TABLE
    vehicle_patern
RENAME COLUMN id TO patern_id;
ALTER TABLE
    vehicle_patern
MODIFY COLUMN patern_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
ADD COLUMN information_id INT,
ADD FOREIGN KEY (information_id) REFERENCES vehicle_information(information_id) ON DELETE CASCADE;
```

```
ALTER TABLE
    vehicle_consumption
    rename column id to consumption_id;
ALTER TABLE
    vehicle_consumption
MODIFY COLUMN consumption_id INT PRIMARY KEY AUTO_INCREMENT,
ADD COLUMN
    information_id INT,
ADD FOREIGN KEY (information_id) REFERENCES vehicle_information(information_id) ON DELETE CASCADE;
```

```
ALTER TABLE
    vehicle_information
RENAME COLUMN id TO information_id;
ALTER TABLE
    vehicle_information
MODIFY COLUMN information_id INT PRIMARY KEY AUTO_INCREMENT;
```

# THE CONNECTION AMONG TABLES

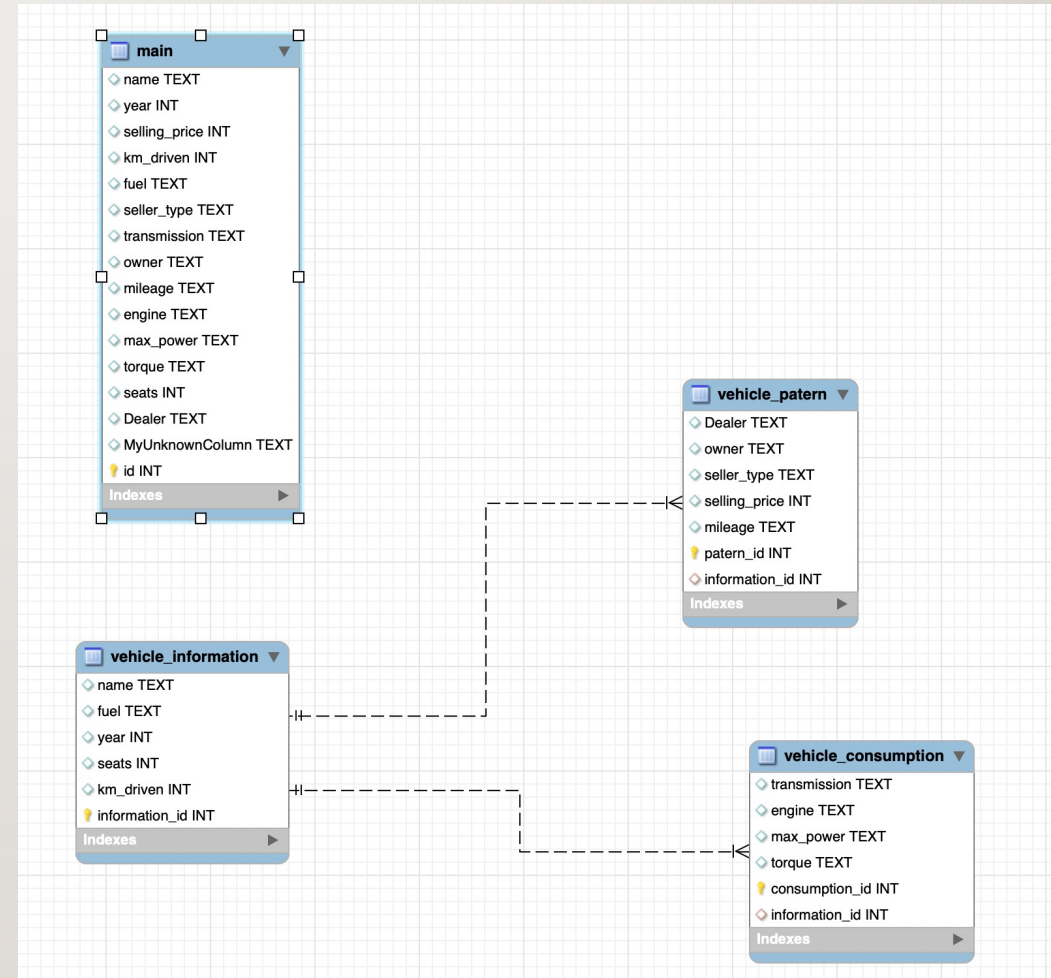
---

I CHANGED THE COLUMNS  
NAMES AND CONNECT  
KEYS OF 3 TABLES



# AFTER CREATING MY TABLES I GOT THIS DIAGRAM

---



# FIRST EXERSISE

---

code

```
SELECT
    name as "Most Sold Vehicle",
    COUNT(name) AS Number_of_Cars
FROM
    vehicle_information
GROUP BY
    name
ORDER BY
    Number_of_Cars DESC
LIMIT 1;
```

result

	Most Sold Vehicle	Number_of_Cars
▶	Maruti Swift Dzire VDI	162

# SECOND TASK

---

Code

```
SELECT
  dealer AS 'Dealer Name',
  COUNT(*) AS Total_Sold_Numer_of_Vehicles,
  SUM(selling_price) AS 'Revenue'
FROM
  vehicle_patern
WHERE
  dealer <> ''
GROUP BY
  dealer
ORDER BY
  Total_Sold_Numer_of_Vehicles DESC
LIMIT 1;
```

result

Dealer Name	Total_Sold_Numer_of_Vehicl...	Revenue
Anny	871	1224686970



# THIRD TASK

---

## Code

```
SELECT
    name AS 'Vehicle Model',
    AVG(selling_price) AS 'Average Price'
FROM
    vehicle_information cd
JOIN
    vehicle_patern ce
ON
    cd.information_id = ce.patern_id
GROUP BY
    name;
```

## result

	Vehicle Model	Average Price
▶	Maruti Swift Dzire VDI	524648.0926
	Mercedes-Benz B Class B180	1450000.0000
	Skoda Rapid 1.5 TDI Ambition	540000.0000
	Honda City 2017-2020 EXi	165142.8571
	Honda Amaze S Diesel	366666.6667
	Hyundai i20 Sportz Diesel	225000.0000
	Mitsubishi Pajero Sport 4X4	1360000.0000
	Mercedes-Benz S-Class S 350 CDI	5962500.0000
	Honda City i DTEC S	607000.0000
	Maruti Ertiga SHVS ZDI Plus	824066.6667
	Maruti Ciaz Zeta	850000.0000
	Honda Jazz VX Diesel	525000.0000
	Maruti Swift VXI BSIII	150000.0000
	BMW 5 Series 523i	975000.0000
	Toyota Innova Crysta 2.8 ZX AT B...	1751600.0000
	Hyundai Xcent 1.2 VTVT E Plus	470000.0000
	Toyota Innova 2.5 G (Diesel) 7 Se...	807666.6667
	Audi A6 2.0 TDI Technology	1750000.0000
	Maruti Wagon R LXI DUO BSIII	138166.6667
	Toyota Corolla Altis 1.8 VL CVT	1590000.0000



# FOURTH TASK

---

## Code

```
SELECT
  name AS 'Vehicle Model',
  year AS 'Year of Manufacture'
FROM vehicle_information
WHERE
  year = (SELECT min(year) FROM vehicle_information)
UNION
SELECT
  name AS 'Vehicle Model',
  year AS 'Year of Manufacture'
FROM
  vehicle_information
WHERE year = (SELECT MAX(year) FROM vehicle_information)
LIMIT 2;
```

## result

	Vehicle Model	Year of Manufactu...
▶	Mahindra Willys CJ 3B 4X4	1983
	Ford Freestyle Titanium Petrol BSIV	2020

# FIFTH TASK

---

## Code

```
SELECT
  name AS 'Vehicle Model',
  year AS 'Year of Manufacture',
  km_driven AS 'Kilometer Driven',
  selling_price AS 'Selling Price'
FROM
  vehicle_information cd
LEFT JOIN vehicle_patern ce
  ON cd.information_id=ce.patern_id
ORDER BY
  selling_price,
  km_driven
LIMIT 1;
```

## result

	Vehicle Model	Year of Manufactu...	Kilometer Driven	Selling Price
▶	Maruti 800 AC	1997	80000	29999

# SIXTH TASK

---

Code

```
SELECT
    sum(selling_price) AS 'Total Revenue of Individual'
FROM
    vehicle_patern
WHERE
    seller_type = "Individual";
```

result

Total Revenue of Individ...
3373265296

# SEVENTH TASK


## Code

```
(SELECT
  dealer as 'Dealer',
  name as 'Vehicle Model',
  COUNT(name) Soled_Vehicles
FROM vehicle_information cd
JOIN vehicle_patern ce
  On cd.information_id = ce.patern_id
WHERE dealer = 'Anny'
GROUP BY name
ORDER BY Soled_Vehicles desc
LIMIT 1)
UNION
(SELECT
  dealer as 'Dealer',
  name as 'Vehicle Model',
  COUNT(name) as Soled_Vehicles
FROM vehicle_information cd
JOIN vehicle_patern ce
  On cd.information_id = ce.patern_id
WHERE dealer = 'Henry'
GROUP BY name
ORDER BY Soled_Vehicles desc
LIMIT 1)
UNION
(SELECT
  dealer as 'Dealer',
  name as 'Vehicle Model',
  COUNT(name) as Soled_Vehicles
FROM vehicle_information cd
JOIN vehicle_patern ce
  On cd.information_id = ce.patern_id
WHERE dealer = 'David'
GROUP BY name
ORDER BY Soled_Vehicles desc
LIMIT 1)
```

## result

	Dealer	Vehicle Model	Soled_Vehicles
▶	Anny	BMW X4 M Sport X xDrive20d	47
▶	David	BMW X4 M Sport X xDrive20d	13
▶	Henry	Maruti Swift VDI BSIV W ABS	3



A top-down view of a Thanksgiving dinner spread on a dark wooden table. The spread includes a whole roasted turkey on a large platter, a pumpkin pie with pecan decorations, a bowl of apple and pecan salad, a bowl of mashed potatoes with carrots, a bowl of Brussels sprouts, a small pumpkin, and various condiments like cranberry sauce and apple sauce. The text "THANKS FOR ATTENTION" is overlaid in white on a dark rectangular background, with a thin orange line underneath.

THANKS FOR ATTENTION