```
use CarsDB
```

Query No.-1

```
select * from cclass
select * from audi
select * from hyndai
select * from merc
select * from models
select * from transmission
select * from fueltype
select * from bmw
```

Query No.-2

```
---AUDI Table---
select * from audi
inner join models on audi.model_ID = models.model_ID
```

Query No.-3

```
---BMW Table---
select * from bmw
inner join models on bmw.model_ID = models.model_ID
```

Query No.-4

```
---Cclass Table---
select * from cclass
inner join models on cclass.model_ID = models.model_ID
```

Query No.-5

```
---Hyundai Table---
select * from hyndai
inner join models on hyndai.model_ID = models.model_ID
```

Query No.-6

```
---Mercedes Table---
select * from merc
inner join models on merc.model_ID = models.model_ID
```

Query No.-7

```
---Adding BrandId---
ALTER TABLE audi
ADD Brand_ID int NOT NULL
DEFAULT 1
WITH VALUES;
```

Query No.-8

```
ALTER TABLE bmw
ADD Brand_ID int NOT NULL
```

```
DEFAULT 2
WITH VALUES;
```

Query No.-9

```
ALTER TABLE cclass
ADD Brand_ID int NOT NULL
DEFAULT 3
WITH VALUES;
```

Query No.-10

```
ALTER TABLE hyndai
ADD Brand_ID int NOT NULL
DEFAULT 4
WITH VALUES;
```

Query No.-11

```
ALTER TABLE merc
ADD Brand_ID int NOT NULL
DEFAULT 5
WITH VALUES;
```

Query No.-12

```
---Creating a View---
create view all_brands_information as
(select 'Audi' as brand_name, * from audi
union
select 'BMW' as brand_name, * from bmw
union
select 'CClass' as brand_name, * from cclass
union
select 'Hyundai'as brand_name, * from hyndai
union
select 'Mercedes'as brand_name, * from merc)
```

Query No.-13

```
---Brands Table---
create table Brands
(ID int , Name varchar(20))

insert into Brands values(1, 'Audi')
insert into Brands values(2, 'BMW')
insert into Brands values(3, 'Mercedes')
insert into Brands values(4, 'Hyundai')
insert into Brands values(5, 'Cclass')

select * from Brands
```

Query No.-14

```
ALTER TABLE Brands
drop column tax
```

Query No.-15

```
ALTER TABLE Brands
drop column mpg
```

Query No.-16

```
---Counting No of cars of each brand---

select count(*) as no_of_audi_cars_sold from audi
select count(*) as no_of_bmw_cars_sold from bmw
select count(*) as no_of_cclass_cars_sold from cclass
select count(*) as no_of_hyundai_cars_sold from hyndai
select count(*) as no_of_mercedes_cars_sold from merc

select * from all_brands_information
```

Query No.-17

```
---Sales data Model wise---
select model_ID, count(*) as No_of_car_sold from all_brands_information
group by model_ID
```

Query No.-18

```
---Sales data of Transmission used in cars---
select transmission_ID, count(*) as No_of_car_sold from all_brands_information
group by transmission_ID
```

Query No.-19

```
---Sales data of Fuel used in cars---
select fuel_ID, count(*) as No_of_car_sold from all_brands_information
group by fuel_ID
```

Query No.-20

```
---Volume of cars sold over the years---
select YEAR, count(*) as No_of_cars_sold from all_brands_information
group by year order by year
```

Query No.-21

```
select brand_name, count(*) as No_of_car_sold from all_brands_information
group by brand_name
```

Query No.-22

```
select count(*) as total_cars_sold from all_brands_information
```

Query No.-

```
select max(price) as maximum_price from all_brands_information
select min(price) as minimum_price from all_brands_information
select avg(price) as average_price from all_brands_information
```

Query No.-23

```
select max(price) as maximum_price,
min(price) as minimum_price,
avg(price) as average_price from all_brands_information
```

Query No.-24

```
select brand_name, min(year) as from_, max(year) as to_, sum(price) as total_amt
from all_brands_information
group by brand_name, floor( year / 6 )
order by brand_name
```

Query No.-25

```
select * from all_brands_information


select case when price <= 10000 then 'LIG'
                     when price > 10000 and price <=20000 then 'MIG'
                     when price > 20000 and price <=30000 then 'HIG'
                     when price > 30000 then 'Rich'
              end income_group, count(*) as Total_Count
from all_brands_information
group by case when price <= 10000 then 'LIG'
                     when price > 10000 and price <=20000 then 'MIG'
                     when price > 20000 and price <=30000 then 'HIG'
                     when price > 30000 then 'Rich'
              end
```

Query No.-26

```
select case when price <= 10000 then 'Mini Compact'
                     when price > 10000 and price <=20000 then 'Sub-Compact'
                     when price > 20000 and price <=40000 then 'Compact'
                     when price > 40000 and price <=650000 then 'Mid-Size'
                     when price > 65000 and price <=100000 then 'Full-Size'
                     when price > 100000 then 'Full-Luxury'
              end categories, year, count(*) as Total_Count
from all_brands_information
group by case when price <= 10000 then 'Mini Compact'
                     when price > 10000 and price <=20000 then 'Sub-Compact'
                     when price > 20000 and price <=40000 then 'Compact'
                     when price > 40000 and price <=650000 then 'Mid-Size'
                     when price > 65000 and price <=100000 then 'Full-Size'
                     when price > 100000 then 'Full-Luxury'
              end, year;
```

Query No.-27

```
select case when price <= 10000 then 'Mini Compact'
                     when price > 10000 and price <=20000 then 'Sub-Compact'
                     when price > 20000 and price <=40000 then 'Compact'
                     when price > 40000 and price <=650000 then 'Mid-Size'
                     when price > 65000 and price <=100000 then 'Full-Size'
                     when price > 100000 then 'Full-Luxury'
              end categories, AVG(mpg) as Fuel_efficiency
from all_brands_information
group by case when price <= 10000 then 'Mini Compact'
                     when price > 10000 and price <=20000 then 'Sub-Compact'
                     when price > 20000 and price <=40000 then 'Compact'
                     when price > 40000 and price <=650000 then 'Mid-Size'
                     when price > 65000 and price <=100000 then 'Full-Size'
                     when price > 100000 then 'Full-Luxury'
              end;
```

Query No.-28

```
select case when price <= 10000 then 'Mini Compact'
                     when price > 10000 and price <=20000 then 'Sub-Compact'
                     when price > 20000 and price <=40000 then 'Compact'
                     when price > 40000 and price <=650000 then 'Mid-Size'
                     when price > 65000 and price <=100000 then 'Full-Size'
                     when price > 100000 then 'Full-Luxury'
              end categories,  (LAST_VALUE(price) OVER (PARTITION BY year ORDER BY year)
-
               FIRST_VALUE(price) OVER (PARTITION BY year ORDER BY year))
from all_brands_information
group by case when price <= 10000 then 'Mini Compact'
                     when price > 10000 and price <=20000 then 'Sub-Compact'
                     when price > 20000 and price <=40000 then 'Compact'
                     when price > 40000 and price <=650000 then 'Mid-Size'
                     when price > 65000 and price <=100000 then 'Full-Size'
                     when price > 100000 then 'Full-Luxury'
              end, year, price;
```

Query No.-29

```
SELECT ABI.brand_name, Mo.model_name, ABI.year, ABI.price, ABI.mileage, ABI.tax, ABI.mpg,
ABI.engineSize, T.transmission, F.fueltype, Fp.column2 as fuelprice,
case when ABI.price <= 10000 then 'Mini Compact'
                     when ABI.price > 10000 and ABI.price <=20000 then 'Sub-Compact'
                     when ABI.price > 20000 and ABI.price <=40000 then 'Compact'
                     when ABI.price > 40000 and ABI.price <=650000 then 'Mid-Size'
                     when ABI.price > 65000 and ABI.price <=100000 then 'Full-Size'
                     when ABI.price > 100000 then 'Full-Luxury'
              end AS category,
case when price <= 10000 then 'LIG'
                     when price > 10000 and price <=20000 then 'MIG'
                     when price > 20000 and price <=30000 then 'HIG'
                     when price > 30000 then 'Rich'
              end income_group
```

```
INTO Brandstable
FROM all_brands_information ABI
LEFT JOIN models Mo ON ABI.model_ID = Mo.model_ID
LEFT JOIN transmission T ON ABI.transmission_ID = T.ID
LEFT JOIN fueltype F ON ABI.fuel_ID = F.fuel_ID
LEFT JOIN fuelprice Fp ON ABI.year = Fp.column1;

SELECT * FROM Brandstable;
```

Query No.-30

```
---Category wise price change---
SELECT category, MAX(price) - MIN(price) AS price_change FROM Brandstable
GROUP BY category;

SELECT category, AVG(mpg) AS avgfuelefficiency, MAX(transmission) AS transmission,
MAX(fueltype) AS fueltype
FROM Brandstable
GROUP BY category;
```

Query No.-31

```
select
    CONCAT(bucket_floor, '-', bucket_ceiling) as fuelspending,
    count(*) as sale
from (
       select
              floor(FUELSPEND/10000.00)*10000 as bucket_floor,
              floor(FUELSPEND/10000.00)*10000 + 10000 as bucket_ceiling
       from fueltbl
) a
group by bucket_floor, bucket_ceiling
order by bucket_floor;

SELECT * INTO fueltbl FROM
(SELECT ((mileage/mpg)*fuelprice) AS FUELSPEND FROM Brandstable
WHERE mpg != 0) AS FLTBL;
```

Query No.-32

```
SELECT * FROM fueltbl;

SELECT category, (MAX(sale) - MIN(sale)) AS sale_incr
FROM (SELECT  category, year, COUNT(*) AS sale FROM Brandstable GROUP BY  category, year
) B
GROUP BY category ;
```

Query No.-33

```
SELECT transmission , COUNT(*) AS Sale
FROM Brandstable
GROUP BY transmission;
```