Raktim Saikia |Nikhil | DDA | April 1, 2021

Database Design and Application

AssignMENET-1

Submitted to: Prof-Amit Dua



Contents

[Requirement: 3](#_Toc69236901)

[Who can Use this Application 3](#_Toc69236902)

[Functions and Benefits 3](#_Toc69236903)

[ER Diagram 4](#_Toc69236904)

[Relational Schema: 4](#_Toc69236905)

[Physical schema 6](#_Toc69236906)

[Functional Dependencies: 7](#_Toc69236907)

[Normalization: 7](#_Toc69236908)

[1st Normal Form 7](#_Toc69236909)

[2nd Normal form 7](#_Toc69236910)

[3rd Normal form 8](#_Toc69236911)

[DECOMPISITION TO HIGHER NF 8](#_Toc69236912)

[To check Dependency Preservation 9](#_Toc69236913)

[To check the above decomposition is Lossless Join 10](#_Toc69236914)

[IMPLEMENTION: 11](#_Toc69236915)

[Create Tables: 11](#_Toc69236916)

[VMS\_CAR 11](#_Toc69236917)

[Car\_TYPE 11](#_Toc69236918)

[VMS\_Customer 11](#_Toc69236919)

[VMS\_DRIVER 12](#_Toc69236920)

[VMS\_OWNER 12](#_Toc69236921)

[VMS\_PAYS 12](#_Toc69236922)

[VMS\_RENTS 13](#_Toc69236923)

[VMS\_RESERVATION 13](#_Toc69236924)

[VMS\_STORE 14](#_Toc69236925)

[INSERT STATEMENT: 14](#_Toc69236926)

[INSERT INTO VMS\_CUSTOMER 14](#_Toc69236927)

[INSERT INTO VMS\_CARTYPE 14](#_Toc69236928)

[INSERT INTO VMS\_STORE 14](#_Toc69236929)

[INSERT INTO VMS\_OWNER 15](#_Toc69236930)

[INSERT INTO VMS\_CAR 15](#_Toc69236931)

[INSER INTO VMS\_DRIVER 15](#_Toc69236932)

[INSERT INTO VMS\_RESERVATION 15](#_Toc69236933)

[INSERT INTO VMS\_RENT 15](#_Toc69236934)

[INSERT INTO VMS\_PAY 16](#_Toc69236935)

[SQL SELECT QUERY: 16](#_Toc69236936)

[QUERY TO FETCH REGERVATION DETAILS 16](#_Toc69236937)

[Query To fetch RENT Details: 16](#_Toc69236938)

[Query to fetch Payment Details 17](#_Toc69236939)

[Query To fetch Driver details and rent for a trip 17](#_Toc69236940)

[Index Creation: 17](#_Toc69236941)

[Index structure 18](#_Toc69236942)

[Space utilized and time saved on the retrieval 18](#_Toc69236943)

[Construct the B+ tree for VMS\_Registration attribute. 19](#_Toc69236944)

[Conclusion 21](#_Toc69236945)

[References 21](#_Toc69236946)

# Requirement:

Vehicle Rental and Management system involved around three main entities Car, User and Reservation.

## Who can Use this Application

This being a Web application, this application would be used by End users i.e. customer to Book any car of his choose and can pay the total rent once the trip is completed

## Functions and Benefits

This Application has the function to capture all details related to any trip from start time to end time and will charge based on the total distance, car type being selected by any user and Number of days

It is also having the feature to pre book and pay at the end of journey

User can also able to canceled the registration any time before 1 hour of registration

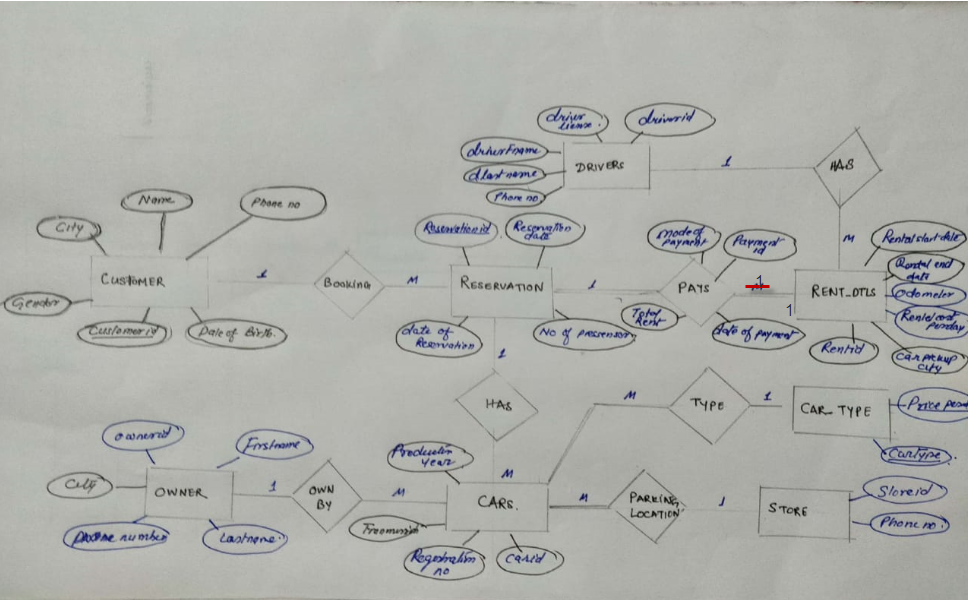
Car can be reserved from a rental location with a specific address. Car has a parameter Car Type. It can be ‘MUV’, ‘Standard’, ’SUV’, ‘Sedan’. Car Type defines the rental price per day

A user can reserve a car for a number of days. When a user books a car he mentions the start date and end date for which he needs the car. The end date will be hypothetical at the time of reservation and updated with actual end date when the car is returned. The total amount and net amount are calculated based on start date, end date, rental price per day

A user can cancel a reserved car before the car has been rented. A reservation can have status as ‘Reserved’, ‘Completed’ and ‘Cancelled’. When the car is reserved, status will be in ‘Reserved’ Status. Once the car is returned and the total amount is paid, the status will be ‘Completed’.

Multiple user and login and perform registration at the same time. Also no login details would be stored and would be done via sso such as google or Fb

# ER Diagram



# Relational Schema:

Relational Schema is the set of formulas (sentence) called integrity constraints imposed on database.

Step 1: Mapping of regular entity types

Step 2: Mapping of weak entity type’s .The ERD of Vehicle Rental and Management system does not contain any weak entity types

Step 3: Mapping of binary 1:1 relationship types The ERD of Vehicle Rental and Management system Contain 1:1 binary relationship

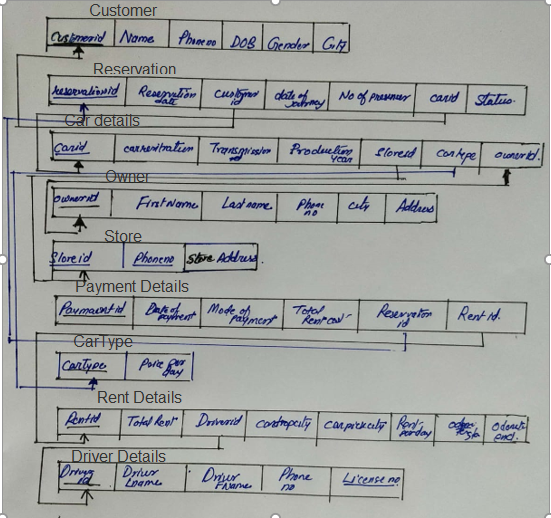
Step 4: Mapping of binary 1: N relationship types The ERD of Vehicle Rental and Management does contain binary 1: N relationship

Step 5: Mapping of binary M:N relationship types

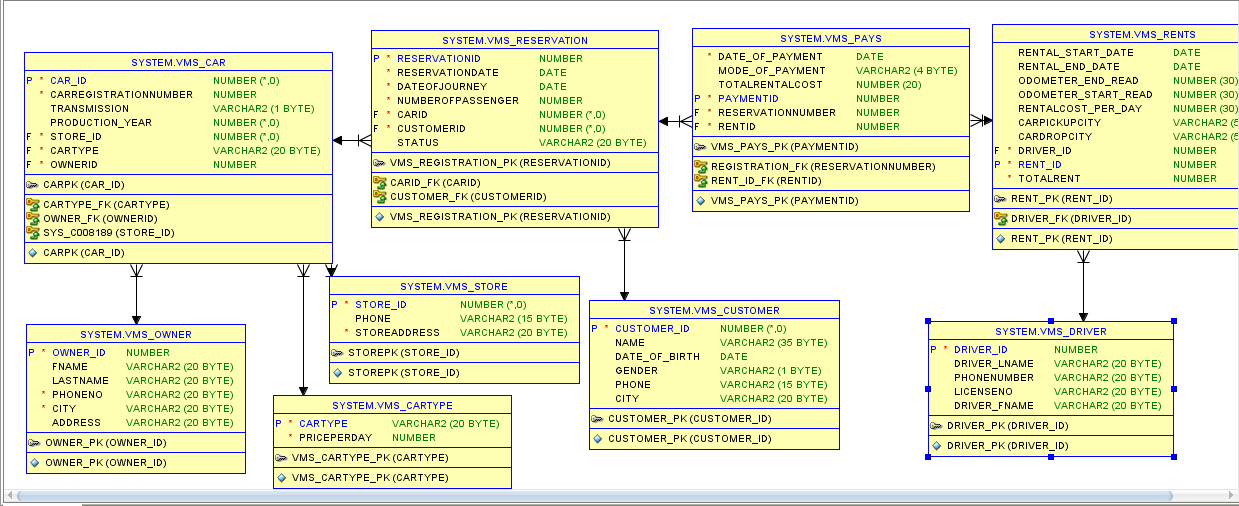
Step 6: Mapping of multivalued attributes The ERD of Travel Agency Management System does not contain any multivalued attributes

Step 7: Mapping of N-array relationship types

Schema diagram The Schema diagram of Travel Agency Management System is shown in the figure:



# Physical schema



# Functional Dependencies:

Here are the 5 relation showcasing the Functional dependencies between various attribute

* Type of a car define the rental price of the car per day
  + Car\_Type🡪Price\_per day
* Customer ID of any user determine information related to customer
  + CustomerID🡪Name, PhoneNo, DOB,Gender,City
* Reservation ID define the reservation related details for any customer
  + ReservationID🡪Reservationdate,DateofJourney,NumberofPressenger,Status
* Store ID define the location where car is available
  + StoreID🡪Address,PhoneNumber
* OwnerID Define owner for any car
  + OwnerID🡪firstname,Lastname, PhonoNumber, Address
  + PhoneNo->,Address

# Normalization:

## 1st Normal Form

Normalization is the process of decomposing unsatisfactory bad relations by breaking up their attributes into smaller relations.

First Normal Form (1NF) The relations are said to be in 1NF if there are no multivalued attributes or nested relations.

In our database design none of the attribute is multivalued attribute hence it is in 1st NF

## 2nd Normal form

Second Normal Form (2NF) The relations are said to be in 2NF if it is in 1NF and no non-key attributes are functionally dependent on part of the primary key.2NF tries to reduce redundant data getting stored in memory

IF there is partial dependencies then relation is not in 2nd normal form. In case of the above functional dependencies primary key is the one that determine all the respective attribute in that table, ie there is no partial dependencies where a non-prim attribute is functionally dependent on a part of candidate key.

***In other way we can also say it is not in 2NF as we only have one primary key in each of the table which means there is no subset of that key. An hence satisfy 2NF condition***

Henceforth all the relation in the application design satisfy 2NF

## 3rd Normal form

Third Normal Form (3NF) The relations are said to be in 3NF if it is 2NF and no non-key attributes are functionally determined by another non- key attribute.

A relation is in third normal form, If there is no transitive dependency for non-Prime attribute as well as it is second normal form

In all of the above relation we have an X that is super key and have a prim attributes which is only determined by super key except in the case of relation of Car owner where non key attribute is determining a non-key attribute so this is in transitive dependencies.

Ie

* OwnerID Define owner for any car
  + OwnerID🡪firstname,Lastname, PhonoNumber, Address
  + PhoneNo->,Address

Because of this relation is in 2nd NF and need to decomposed the above relation to 3nf

## DECOMPISITION TO HIGHER NF

Now consider

R(ownerid, fistname, lastname, phoneNumber, Address)

F{ownerid🡪firstname,LastName,Phonenumber,Address, PhoneNumber->Address}

Phonenumber clousure 🡪phonenumber, Address

Therefore after decomposition we get below two relation R11 and R12 and to get lossless join decomposition we have to have common attribute and in our case Phone Number is common between both the relations

**R11{ phoneNumber, Address } ,**

**R12{Ownerid, firstname, lastname,PhoneNumber} with functional dependencies as**

**F11{PhoneNumber🡪Address} and F12{ ownerid-> firstname, lastname,PhoneNumber}**

**Now to confirm the highest normal from we can confirm that it satisfy 2NF and there is no Transitive dependencies as well in both the decomposed relation, So this is definitely in 3NF.**

**Lastly each non trivial FD, determinant is a Superkey and as per the definition if any relation is in 3NF and if every functional dependency X → Y, X is the super key of the table then Relation is in BCNF.**

**Apart from this other 4 relation are already in 3NF as does not have transitive dependencies and also determinant is a Superkey in call cases.**

**So we can conclude that Relations are in BCNF normal form**

## To check Dependency Preservation

**Original Relation and Functional dependencies**

R(ownerid, fistname, lastname, phoneNumber, Address)

F{ownerid🡪firstname,LastName,Phonenumber,Address, PhoneNumber->Address}

**Decomposed relation**

**R11{ phoneNumber, Address } ,**

**R12{Ownerid, firstname, lastname,PhoneNumber} with functional dependencies as**

**F11{PhoneNumber🡪Address} and F12{ ownerid-> firstname, lastname,PhoneNumber}**

**Now to check dependency preservation we need to do union of new Functional dependency**

**F11{PhoneNumber🡪Address} Union F12{ ownerid-> firstname, lastname,PhoneNumber}**

**As we can see every FD of original Relation is member of decomposed relation and hence we can say above decomposition is a dependency preservation**

## To check the above decomposition is Lossless Join

**Step1- To union both the relation to check if all the attribute are there or not as that of original relation.**

**In our case if we do union we can get all the attribute of the original relation**

**Spep2- To check by doing a natural join among two decomposed relation.**

**R1112{** ownerid, fistname, lastname, phoneNumber, Address **}**

**In our case PhoneNumber is a common attribute and hence we can do Natural join among both the relation and Phone Number is Candidate key of relation R11.**

**As both the Condition satisfies we can conclude that above decomposition is Lossless join decomposition**

# IMPLEMENTION:

## Create Tables:

### VMS\_CAR

CREATE TABLE "SYSTEM"."VMS\_CAR" ( "CAR\_ID" NUMBER(\*,0) NOT NULL ENABLE, "CARREGISTRATIONNUMBER" NUMBER NOT NULL ENABLE, "TRANSMISSION" VARCHAR2(1 BYTE), "PRODUCTION\_YEAR" NUMBER(\*,0), "STORE\_ID" NUMBER(\*,0) NOT NULL ENABLE, "CARTYPE" VARCHAR2(20 BYTE) NOT NULL ENABLE, "OWNERID" NUMBER NOT NULL ENABLE, CHECK (Transmission in ('a', 'm')) ENABLE, CONSTRAINT "CARPK" PRIMARY KEY ("CAR\_ID") USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ENABLE, FOREIGN KEY ("STORE\_ID") REFERENCES "SYSTEM"."VMS\_STORE" ("STORE\_ID") ON DELETE SET NULL ENABLE, CONSTRAINT "CARTYPE\_FK" FOREIGN KEY ("CARTYPE") REFERENCES "SYSTEM"."VMS\_CARTYPE" ("CARTYPE") ENABLE, CONSTRAINT "OWNER\_FK" FOREIGN KEY ("OWNERID") REFERENCES "SYSTEM"."VMS\_OWNER" ("OWNER\_ID") ENABLE ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ;

### Car\_TYPE

CREATE TABLE "SYSTEM"."VMS\_CARTYPE" ( "CARTYPE" VARCHAR2(20 BYTE) NOT NULL ENABLE, "PRICEPERDAY" NUMBER NOT NULL ENABLE, CONSTRAINT "VMS\_CARTYPE\_PK" PRIMARY KEY ("CARTYPE") USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ENABLE ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ;

### VMS\_Customer

CREATE TABLE "SYSTEM"."VMS\_CUSTOMER" ( "CUSTOMER\_ID" NUMBER(\*,0) NOT NULL ENABLE, "NAME" VARCHAR2(35 BYTE), "DATE\_OF\_BIRTH" DATE, "GENDER" VARCHAR2(1 BYTE), "PHONE" VARCHAR2(15 BYTE), "CITY" VARCHAR2(20 BYTE), CHECK (gender in ('m', 'f')) ENABLE, CONSTRAINT "CUSTOMER\_PK" PRIMARY KEY ("CUSTOMER\_ID") USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ENABLE ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ;

### VMS\_DRIVER

CREATE TABLE "SYSTEM"."VMS\_DRIVER" ( "DRIVER\_ID" NUMBER NOT NULL ENABLE, "DRIVER\_LNAME" VARCHAR2(20 BYTE), "PHONENUMBER" VARCHAR2(20 BYTE), "LICENSENO" VARCHAR2(20 BYTE), "DRIVER\_FNAME" VARCHAR2(20 BYTE), CONSTRAINT "DRIVER\_PK" PRIMARY KEY ("DRIVER\_ID") USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ENABLE ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ;

### VMS\_OWNER

CREATE TABLE "SYSTEM"."VMS\_OWNER" ( "OWNER\_ID" NUMBER NOT NULL ENABLE, "FNAME" VARCHAR2(20 BYTE), "LASTNAME" VARCHAR2(20 BYTE), "PHONENO" VARCHAR2(20 BYTE) NOT NULL ENABLE, "CITY" VARCHAR2(20 BYTE) NOT NULL ENABLE, "ADDRESS" VARCHAR2(20 BYTE), CONSTRAINT "OWNER\_PK" PRIMARY KEY ("OWNER\_ID") USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ENABLE ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ;

### VMS\_PAYS

CREATE TABLE "SYSTEM"."VMS\_PAYS" ( "DATE\_OF\_PAYMENT" DATE NOT NULL ENABLE, "MODE\_OF\_PAYMENT" VARCHAR2(4 BYTE), "TOTALRENTALCOST" NUMBER(20,0), "PAYMENTID" NUMBER NOT NULL ENABLE, "RESERVATIONNUMBER" NUMBER NOT NULL ENABLE, "RENTID" NUMBER NOT NULL ENABLE, CHECK (mode\_of\_payment in ('cash', 'card')) ENABLE, CONSTRAINT "VMS\_PAYS\_PK" PRIMARY KEY ("PAYMENTID") USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ENABLE, CONSTRAINT "REGISTRATION\_FK" FOREIGN KEY ("RESERVATIONNUMBER") REFERENCES "SYSTEM"."VMS\_RESERVATION" ("RESERVATIONID") ENABLE, CONSTRAINT "RENT\_ID\_FK" FOREIGN KEY ("RENTID") REFERENCES "SYSTEM"."VMS\_RENTS" ("RENT\_ID") ENABLE ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ;

### VMS\_RENTS

CREATE TABLE "SYSTEM"."VMS\_RENTS" ( "RENTAL\_START\_DATE" DATE, "RENTAL\_END\_DATE" DATE, "ODOMETER\_END\_READ" NUMBER(30,0), "ODOMETER\_START\_READ" NUMBER(30,0), "RENTALCOST\_PER\_DAY" NUMBER(30,0), "CARPICKUPCITY" VARCHAR2(50 BYTE), "CARDROPCITY" VARCHAR2(50 BYTE), "DRIVER\_ID" NUMBER NOT NULL ENABLE, "RENT\_ID" NUMBER NOT NULL ENABLE, "TOTALRENT" NUMBER NOT NULL ENABLE, CONSTRAINT "RENT\_PK" PRIMARY KEY ("RENT\_ID") USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ENABLE, CONSTRAINT "DRIVER\_FK" FOREIGN KEY ("DRIVER\_ID") REFERENCES "SYSTEM"."VMS\_DRIVER" ("DRIVER\_ID") ENABLE ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ;

### VMS\_RESERVATION

CREATE TABLE "SYSTEM"."VMS\_RESERVATION" ( "RESERVATIONID" NUMBER NOT NULL ENABLE, "RESERVATIONDATE" DATE NOT NULL ENABLE, "DATEOFJOURNEY" DATE NOT NULL ENABLE, "NUMBEROFPASSENGER" NUMBER NOT NULL ENABLE, "CARID" NUMBER NOT NULL ENABLE, "CUSTOMERID" NUMBER NOT NULL ENABLE, "STATUS" VARCHAR2(20 BYTE), CONSTRAINT "VMS\_REGISTRATION\_PK" PRIMARY KEY ("RESERVATIONID") USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ENABLE, CONSTRAINT "CUSTOMER\_FK" FOREIGN KEY ("CUSTOMERID") REFERENCES "SYSTEM"."VMS\_CUSTOMER" ("CUSTOMER\_ID") ENABLE, CONSTRAINT "CARID\_FK" FOREIGN KEY ("CARID") REFERENCES "SYSTEM"."VMS\_CAR" ("CAR\_ID") ENABLE ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ;

### VMS\_STORE

CREATE TABLE "SYSTEM"."VMS\_STORE" ( "STORE\_ID" NUMBER(\*,0) NOT NULL ENABLE, "PHONE" VARCHAR2(15 BYTE), "STOREADDRESS" VARCHAR2(20 BYTE) NOT NULL ENABLE, CONSTRAINT "STOREPK" PRIMARY KEY ("STORE\_ID") USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ENABLE ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT) TABLESPACE "SYSTEM" ;

## INSERT STATEMENT:

### INSERT INTO VMS\_CUSTOMER

INSERT INTO "SYSTEM"."VMS\_CUSTOMER" (CUSTOMER\_ID, NAME, DATE\_OF\_BIRTH, GENDER, PHONE, CITY) VALUES ('2', 'RekhaSaikia', TO\_DATE('1996-06-18 13:54:27', 'YYYY-MM-DD HH24:MI:SS'), 'f', '7696111150', 'Guwahati')

### INSERT INTO VMS\_CARTYPE

INSERT INTO "SYSTEM"."VMS\_CARTYPE" (CARTYPE, PRICEPERDAY) VALUES ('SUV', '2000')

INSERT INTO "SYSTEM"."VMS\_CARTYPE" (CARTYPE, PRICEPERDAY) VALUES ('MUV', '1500')

INSERT INTO "SYSTEM"."VMS\_CARTYPE" (CARTYPE, PRICEPERDAY) VALUES ('Sudan', '1500')

INSERT INTO "SYSTEM"."VMS\_CARTYPE" (CARTYPE, PRICEPERDAY) VALUES ('Mini', '1000')

### INSERT INTO VMS\_STORE

INSERT INTO "SYSTEM"."VMS\_STORE" (STORE\_ID, PHONE) VALUES ('32', '9999887646')

INSERT INTO "SYSTEM"."VMS\_STORE" (STORE\_ID, PHONE) VALUES ('12', '9999834646');

INSERT INTO "SYSTEM"."VMS\_STORE" (STORE\_ID, PHONE) VALUES ('22', '9999823646');

INSERT INTO "SYSTEM"."VMS\_STORE" (STORE\_ID, PHONE) VALUES ('02', '9999822646');

### INSERT INTO VMS\_OWNER

INSERT INTO "SYSTEM"."VMS\_OWNER" (OWNER\_ID, FNAME, LASTNAME, PHONENO, CITY, ADDRESS) VALUES ('6721', 'Suresh', 'A', '990322221', 'Benglore', 'C23 block C');

INSERT INTO "SYSTEM"."VMS\_OWNER" (OWNER\_ID, FNAME, LASTNAME, PHONENO, CITY, ADDRESS) VALUES ('6722', 'Ramesh', 'App', '990322222', 'Benglore', 'C23 block D')

### INSERT INTO VMS\_CAR

INSERT INTO "SYSTEM"."VMS\_CAR" (CAR\_ID, REGISTRATIONNUMBER, TRANSMISSION, PRODUCTION\_YEAR, STORE\_ID, CARTYPE, OWNERID) VALUES ('1', '2133', 'a', '2020', '32', 'SUV', '6721');

INSERT INTO "SYSTEM"."VMS\_CAR" (CAR\_ID, REGISTRATIONNUMBER, TRANSMISSION, PRODUCTION\_YEAR, STORE\_ID, CARTYPE, OWNERID) VALUES ('2', '2134', 'm', '2021', '32', 'SUV', '6721');

INSERT INTO "SYSTEM"."VMS\_CAR" (CAR\_ID, REGISTRATIONNUMBER, TRANSMISSION, PRODUCTION\_YEAR, STORE\_ID, CARTYPE, OWNERID) VALUES ('3', '2135', 'a', '2021', '12', 'SUV', '6721');

INSERT INTO "SYSTEM"."VMS\_CAR" (CAR\_ID, REGISTRATIONNUMBER, TRANSMISSION, PRODUCTION\_YEAR, STORE\_ID, CARTYPE, OWNERID) VALUES ('4', '2136', 'm', '2021', '02', 'MUV', '6721');

INSERT INTO "SYSTEM"."VMS\_CAR" (CAR\_ID, REGISTRATIONNUMBER, TRANSMISSION, PRODUCTION\_YEAR, STORE\_ID, CARTYPE, OWNERID) VALUES ('5', '2137', 'm', '2021', '22', 'Sudan', '6721');

### INSER INTO VMS\_DRIVER

INSERT INTO "SYSTEM"."VMS\_DRIVER" (DRIVER\_ID, DRIVER\_LNAME, PHONENUMBER, LICENSENO, DRIVER\_FNAME) VALUES ('4311', '322133', '445656122', '7663771', 'Ravi')

INSERT INTO "SYSTEM"."VMS\_DRIVER" (DRIVER\_ID, DRIVER\_LNAME, PHONENUMBER, LICENSENO, DRIVER\_FNAME) VALUES ('4312', 'Kumar', '765656122', '7663743', 'James')

### INSERT INTO VMS\_RESERVATION

INSERT INTO "SYSTEM"."VMS\_RESERVATION" (RESERVATIONID, RESERVATIONNDATE, DATEOFJOURNEY, NUMBEROFPASSENGER, CARID, CUSTOMERID) VALUES ('4553', TO\_DATE('2021-04-01 14:05:33', 'YYYY-MM-DD HH24:MI:SS'), TO\_DATE('2021-04-03 14:05:42', 'YYYY-MM-DD HH24:MI:SS'), '2', '1', '1')

### INSERT INTO VMS\_RENT

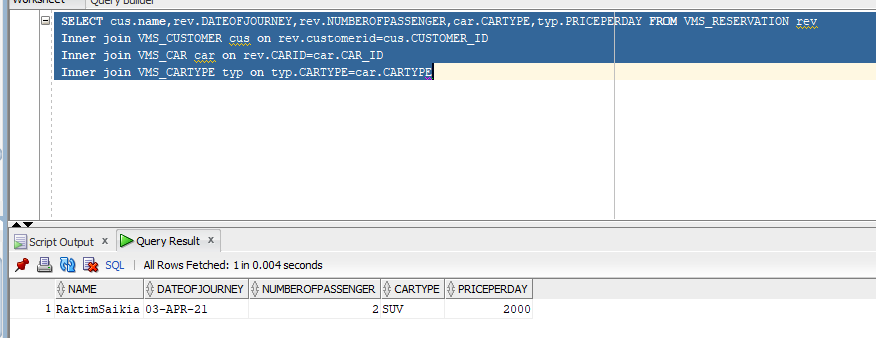
INSERT INTO "SYSTEM"."VMS\_RENTS" (RENTAL\_START\_DATE, RENTAL\_END\_DATE, ODOMETER\_END\_READ, ODOMETER\_START\_READ, RENTALCOST\_PER\_DAY, CARPICKUPCITY, CARDROPCITY, CUSTOMER\_ID, CAR\_ID, STORE\_ID, DRIVER\_ID, RENT\_ID, TOTALRENT) VALUES (TO\_DATE('2021-04-02 14:19:47', 'YYYY-MM-DD HH24:MI:SS'), TO\_DATE('2021-04-03 14:19:53', 'YYYY-MM-DD HH24:MI:SS'), '12', '122', '132', 'ghy', 'ghy', '1', '1', '32', '4311', '1001', '2001')

### INSERT INTO VMS\_PAY

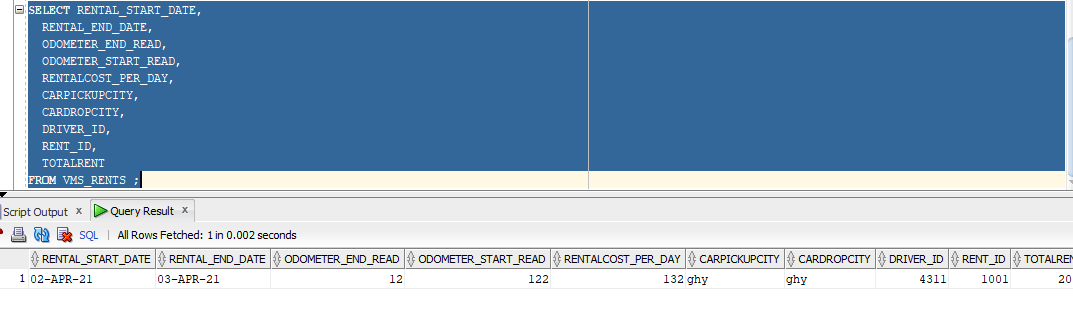
INSERT INTO "SYSTEM"."VMS\_PAYS" (DATE\_OF\_PAYMENT, MODE\_OF\_PAYMENT, TOTALRENTALCOST, CUSTOMER\_ID, STORE\_ID, CAR\_ID, PAYMENTID, REGISTRATIONID, RENTID) VALUES (TO\_DATE('2021-04-03 14:26:28', 'YYYY-MM-DD HH24:MI:SS'), 'card', '213', '1', '32', '1', '332', '4553', '1001')

## SQL SELECT QUERY:

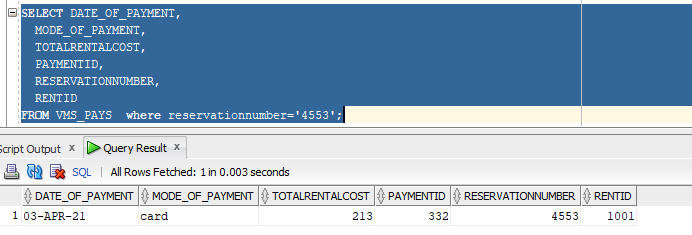
### QUERY TO FETCH REGERVATION DETAILS



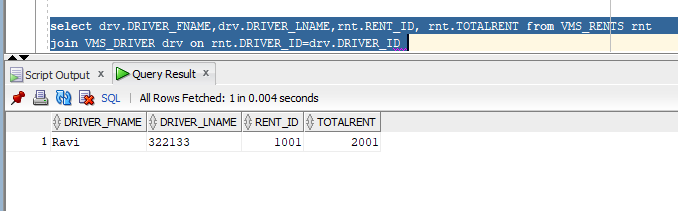
### Query To fetch RENT Details:



### Query to fetch Payment Details

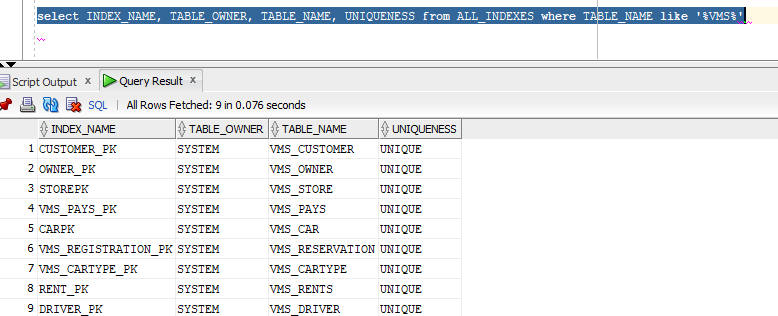


### Query To fetch Driver details and rent for a trip



## Index Creation:

Default Indexes based on Primary Key:



New Index CITY\_Name has been created on Customer table. This index will help to get information of user that are using the application in various City. This is enhance the performance to get users based on various city.

### Index structure

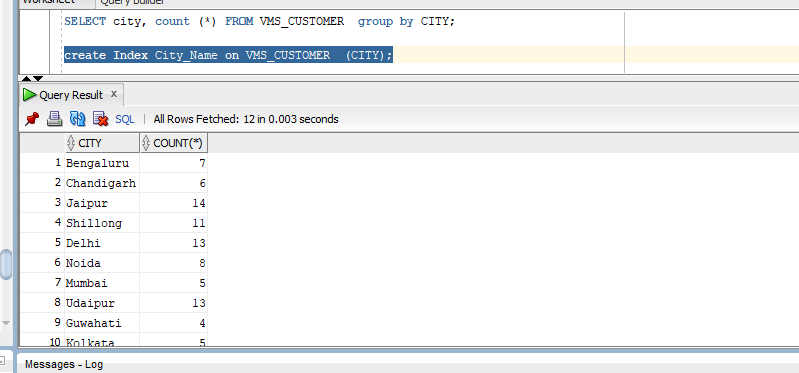
This is basically is secondary index as it is on a unsorted attribute.(Secondary index is an index that is not a primary index and may have duplicates). Also it is type of dense indexing and sparse indexing is not possible.

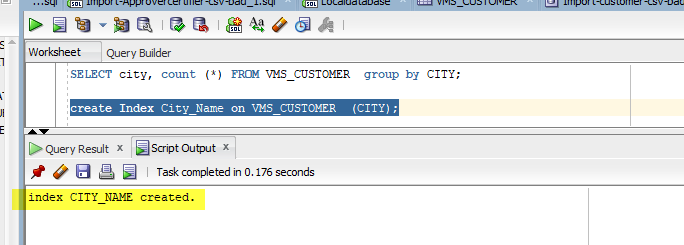
### Space utilized and time saved on the retrieval

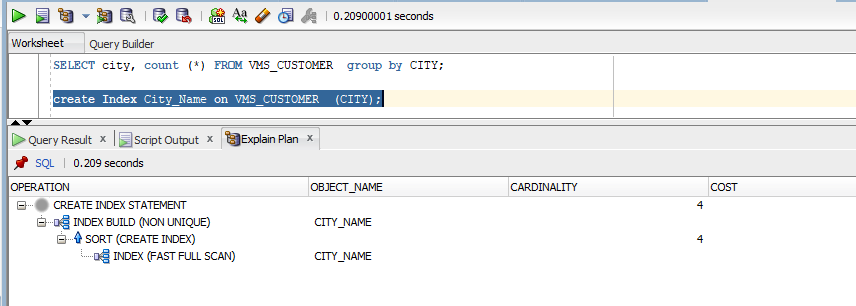
Number of bock access will be upper bond of log of n base 2 +1 and will improve the time complexity by log of n. Additional space would be needed to save the index file and will need to maintain N number of entries as that of the original table.

If we do not use this index then time complexity and in worst case it would be order of N

In our case if we want to search any user based on city\_name , witout index it would going to take 86 by number of entry in any block ie total entry, Once we have the index it would be log 86 +1 ie 7 by number of entries in any block







### Construct the B+ tree for VMS\_Registration attribute.

We have following registration number that has been done against various user

4530,4531,4532,4533,4534,4535,4536,

Now consider order (m) is 4

Therefore maximum children is 4 and minimum children is ceil of 4 by 2 ie 2

Maximum key is 4-1 ie 3

And minimum key is 2-1 ie 1

Now we can contract B+ tress as follows for the above key attribute.

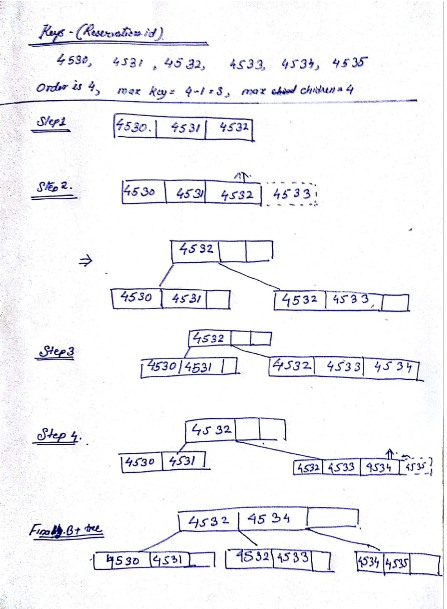


Fig: B + Tree

Conclusion

As part of the assignment we have achieved the following

1. Designed an Entity-Relationship diagram to represent the business rules.

2. Mapped the ER diagram to a relational table

3. Created a database in oracle, as per the schema, complete with all the tables and respective attributes

4. Inserted some sample representative data into the database

5. Executed some queries to retrieve data from the database, so as to achieve the objectives defined in the problem statement.

References

1. Ramez Elmasri & Shamkant B. Navathe, Database Systems; Models, Languages, Design and Application Programming, Pearson Education, 7th Edition, 2017.

2. Abraham Silberschatz, Henry F Korth and S Sudarshan, Database System Concepts, McGraw Hill, 6th Ed., 2013.