CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY

DEVANG PATEL INSTITUE OF ADVANCE TECHNOLOGY AND RESEARCH

NAME: Parth N Patel

ID: 19DCS098

SUBJECT: DBMS (Mini Project)

SUBJECT CODE: CE246

SEM: 4

PROJECT TITLE: BUS RESERVATION SYSTEM

DATA DICTIONARY:

TABLE NAME: BUS_INFO

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
BUS_ID	Number	3	For uniquely identify all buses
BUS_NAME	Varchar2	10	Name of bus
DRIVER_ID	Number	3	Id of driver driving the bus
TOTAL_SEATS	Number	2	Seats of bus
RESERVED_SEATS	Number	2	Reserved seats in bus

TABLE NAME: DRIVER

COLUMN NAME	DATA TYPE	LENGTH	DESCRPITION
DRIVER_ID	Number	2	ID of the driver
DRIVER_NAME	Varchar2	20	Name of driver

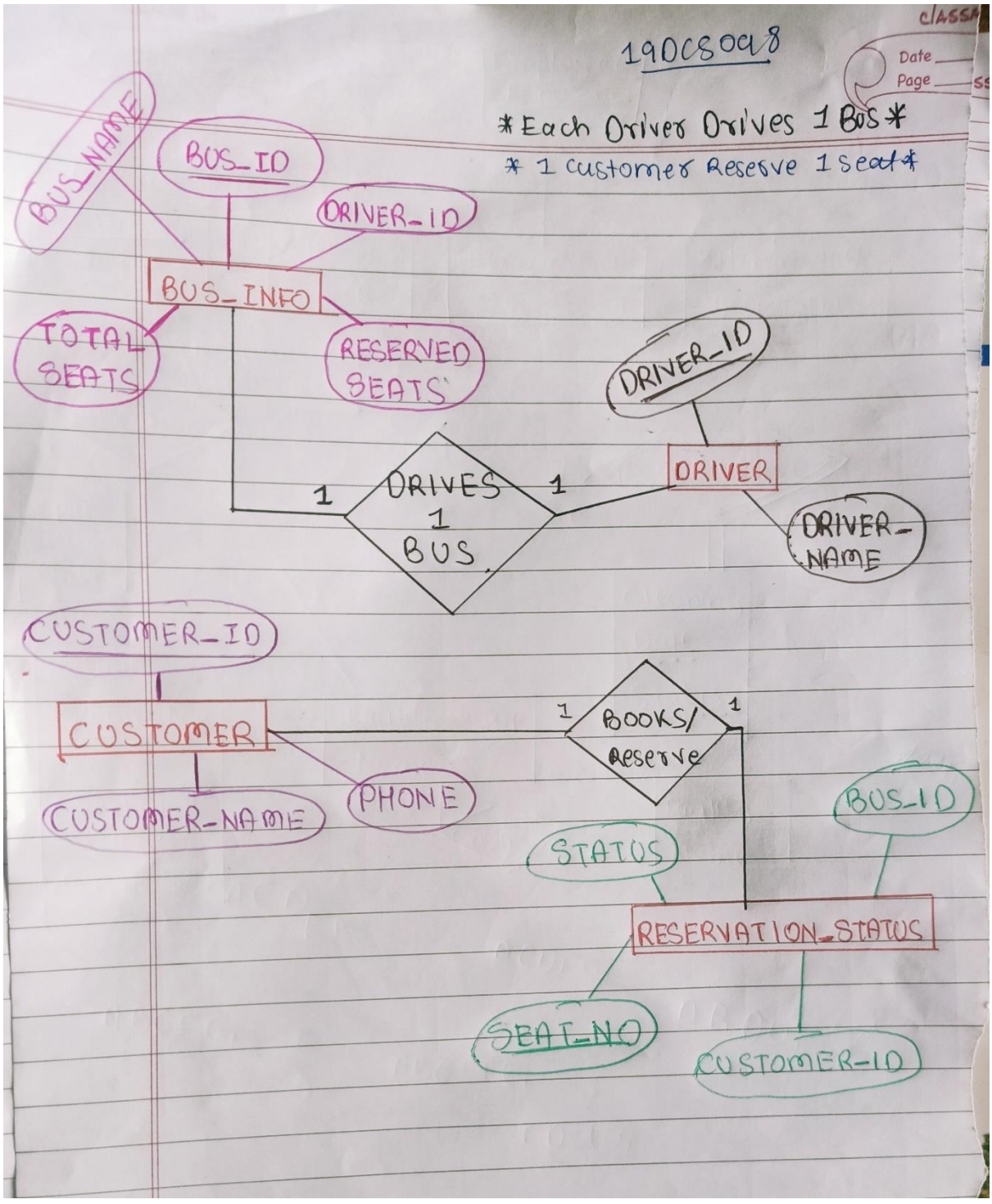
TABLE NAME: CUSTOMER

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
CUSTOMER_ID	Number	3	ID of customer
CUSTOMER_NAME	Varchar2	20	Name of customer
PHONE	Number	10	Phone number of customers

TABLE NAME: RESERVATION_STATUS

COLUMN NAME	DATA TYPE	LENGTH	DESCRIPTION
SEAT_NO	Number	4	Seat number in bus
CUSTOMER_ID	Number	3	ID of customer travelling
BUS_ID	Number	3	ID of BUS having seat number
STATUS	Char	3	We let know whether seat reserved or not

E-R DIAGRAM:


19DCS098
Date
Page
Each Driver Drives 1 Bus
* 1 customer Reserve 1 seat *
BUS_NAME
BUS_ID
DRIVER_ID
BUS_INFO
TOTAL SEATS
RESERVED SEATS
DRIVER_ID
DRIVER
DRIVER_NAME
1
DRIVES 1 BUS
1
CUSTOMER_ID
CUSTOMER
CUSTOMER-NAME
PHONE
1
BOOKS/ Reserve
1
BUS_ID
STATUS
RESERVATION_STATUS
SEAT_NO
CUSTOMER-ID

PROGRAM CODE:

/* CODE FOR CREATING TABLES */

```
CREATE TABLE bus_info ( bus_id NUMBER(3), bus_name VARCHAR2(10), driver_id NUMBER(3), total_seats NUMBER(2), reserved_seats NUMBER(2));

CREATE TABLE driver (driver_id NUMBER(2), driver_name VARCHAR2(20));

CREATE TABLE customer (customer_id NUMBER(3), customer_name VARCHAR2(20), phone NUMBER(10));

CREATE TABLE reservation_status (seat_no NUMBER(4), customer_id NUMBER(3), bus_id NUMBER(3), status CHAR(3));
```

/* CODE FOR ASSIGNNING THE PRIMARY KEY TO TABLES */

```
ALTER TABLE bus_info ADD CONSTRAINT pk_bus_id PRIMARY KEY(bus_id);

ALTER TABLE customer ADD CONSTRAINT pk_customer_id PRIMARY KEY(customer_id);

ALTER TABLE driver ADD CONSTRAINT pk_driver_id PRIMARY KEY(driver_id);

ALTER TABLE reservation_status ADD CONSTRAINT pk_seat_no PRIMARY KEY(seat_no);
```

```
/*ASSIGNNING THE FOREIGN KEY */

ALTER TABLE reservation_status
ADD CONSTRAINTS fk_customer_id FOREIGN KEY(customer_id) REFERENCES
customer;

ALTER TABLE reservation_status
ADD CONSTRAINTS fk_bus_id FOREIGN KEY(bus_id) REFERENCES bus_info;

ALTER TABLE bus_info
ADD CONSTRAINTS fk_driver_id FOREIGN KEY(driver_id) REFERENCES driver;

/* CODE FOR INSERTING THE DATA IN DRIVER TABLE */

CREATE OR REPLACE PROCEDURE UPDATE_DRIVER(id NUMBER, name
VARCHAR2)
IS
BEGIN
 IF(id>0) THEN
   INSERT INTO driver(driver_id,driver_name) VALUES (id,name);
   DBMS_OUTPUT.PUT_LINE('DRIVER INFO UPDATED');
 ELSE
   DBMS_OUTPUT.PUT_LINE('ERROR ENCOUNTERED');
 END IF;
END
```

```
/* CODE FOR INSERTING THE BUS DATA */


CREATE OR REPLACE PROCEDURE BUS(id NUMBER, busname VARCHAR2,seat
NUMBER)

IS BEGIN

 IF(id>0) THEN

   INSERT INTO bus_info(bus_id,bus_name,total_seats,reserved_seats) VALUES
(id,busname,seat,0);

   DBMS_OUTPUT.PUT_LINE('BUS INFO UPDATED');

 ELSE

  DBMS_OUTPUT.PUT_LINE('ERROR ENCOUNTERED');

 END IF;

END;


/* CODE FOR INSERTING THE CUSTOMER DATA */



CREATE OR REPLACE PROCEDURE CUSTOMER_INFO(id NUMBER,name
VARCHAR2,ph NUMBER)

IS BEGIN

 IF(id>0) THEN

   INSERT INTO customer(customer_id,customer_name,phone) VALUES (id,name,ph);

   DBMS_OUTPUT.PUT_LINE('CUSTOMER INFORMATION UPDATED');

 ELSE

  DBMS_OUTPUT.PUT_LINE('ERROR ENCOUNTERED');

 END IF;

END;
```

/*CODE FOR INSERTING CUSTOMER DETAILS IN RESERVATION TABLE*/

```
CREATE OR REPLACE PROCEDURE RESERVATION_INFO(id NUMBER,b_id
NUMBER)
IS BEGIN
 IF(id>0) THEN
    INSERT INTO reservation_status(seat_no,bus_id,status) VALUES (id,b_id,'NO');
    DBMS_OUTPUT.PUT_LINE('SEAT INFORMATION UPDATED');
 ELSE
   DBMS_OUTPUT.PUT_LINE('ERROR ENCOUNTERED');
 END IF;
END;
```

/* CODE FOR ASSIGNNING THE DRIVER TO HIS/HER BUS */

```
CREATE OR REPLACE PROCEDURE ASSIGN_DRIVER(id NUMBER,d_id NUMBER)
IS BEGIN
 IF(id>0) THEN
    UPDATE bus_info SET driver_id=d_id WHERE bus_id=id;
    DBMS_OUTPUT.PUT_LINE('DRIVER ASSIGNED TO BUS');
 ELSE
   DBMS_OUTPUT.PUT_LINE('ERROR ENCOUNTERED');
 END IF;
END;
```

/*CODE FOR CONFIRMING THE RESERVATION */

```
CREATE OR REPLACE PROCEDURE CONFIRM_RESERVATION(id NUMBER,cust_id
NUMBER,b_id NUMBER)
IS BEGIN
 IF(id>0) THEN
   UPDATE reservation_status SET customer_id=cust_id,bus_id=b_id,status='YES' WHERE
seat_no=id;
   UPDATE bus_info SET reserved_seats=reserved_seats+1 WHERE bus_id=b_id;
   DBMS_OUTPUT.PUT_LINE('RESERVATION CONFIRMED');
 ELSE
  DBMS_OUTPUT.PUT_LINE('ERROR ENCOUNTERED');
 END IF;
END;
```

```
/*CODE FOR CANCELLING THE RESERVATION */CREATE OR REPLACE
PROCEDURE DELETE_RESERVATION(id NUMBER,b_id NUMBER)
IS BEGIN
 IF(id>0) THEN
   UPDATE reservation_status SET customer_id=NULL,bus_id=NULL,status='NO'
WHERE seat_no=id;
   UPDATE bus_info SET reserved_seats=reserved_seats-1 WHERE bus_id=b_id;
   DBMS_OUTPUT.PUT_LINE('RESERVATION CANCELLED');
 ELSE
  DBMS_OUTPUT.PUT_LINE('ERROR ENCOUNTERED');
 END IF;
END;
```

/* CREATING VIEW FOR CUSTOMER TO SEE THE RESERVATION INFORMATION*/

```
CREATE VIEW customer_view AS
SELECT c.customer_id,c.customer_name,c.phone,r.seat_no,r.bus_id,r.status
FROM customer c INNER JOIN reservation_status r ON c.customer_id=r.customer_id;
```

/* CREATING VIEW FOR DRIVER AND BUS ASSIGNED TO THEM */

```
CREATE VIEW driver_view AS
SELECT d.driver_id,d.driver_name,b.bus_id,b.bus_name
FROM driver D INNER JOIN bus_info b ON d.driver_id=b.driver_id;
```

/* CREATING VIEW FOR OWNER TO SEE WHICH DRIVER IS ASSIGNNED TO WHICH BUS */

```
CREATE VIEW TRANSPORTER_VIEW_BUS AS
SELECT d.driver_id,d.driver_name,b.bus_id,b.bus_name,b.total_seats,b.reserved_seats
FROM driver d INNER JOIN bus_info b ON d.driver_id=b.driver_id;
```

IMPLEMENTATION SCREEN SHOTS:

- **DETAILS OF BUS_INFO**:

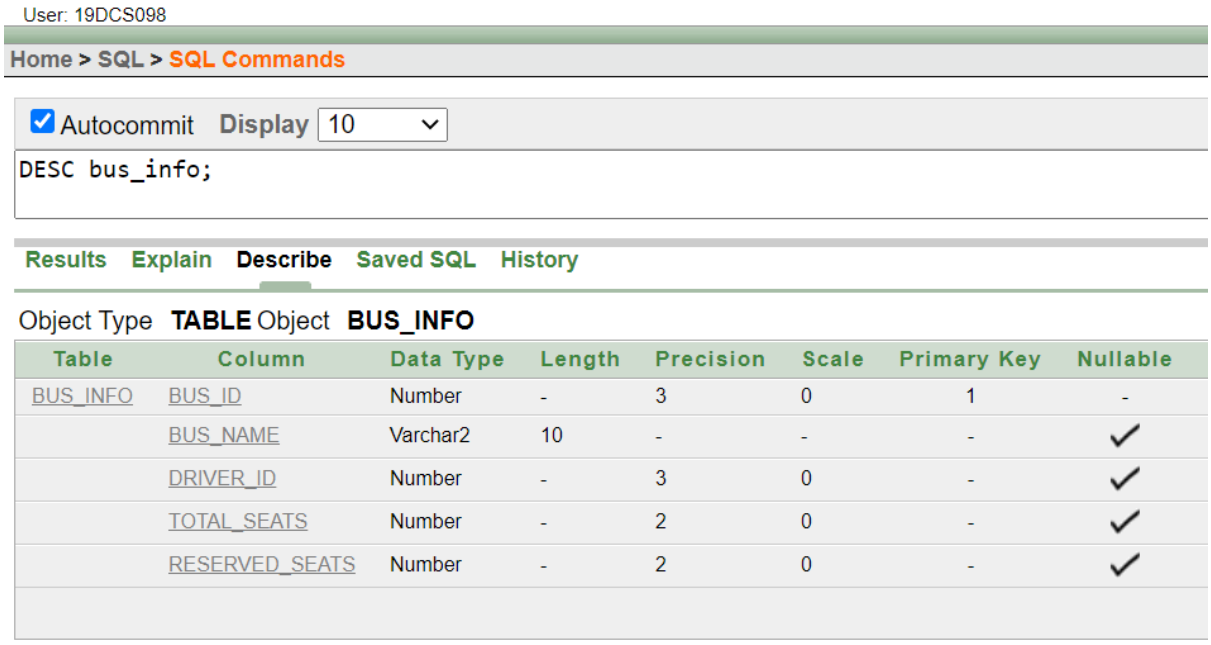


Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable
BUS_INFO	BUS_ID	Number	-	3	0	1	-
	BUS_NAME	Varchar2	10	-	-	-	✓
	DRIVER_ID	Number	-	3	0	-	✓
	TOTAL_SEATS	Number	-	2	0	-	✓
	RESERVED_SEATS	Number	-	2	0	-	✓

- **DETAILS OF DRIVER TABLE**:

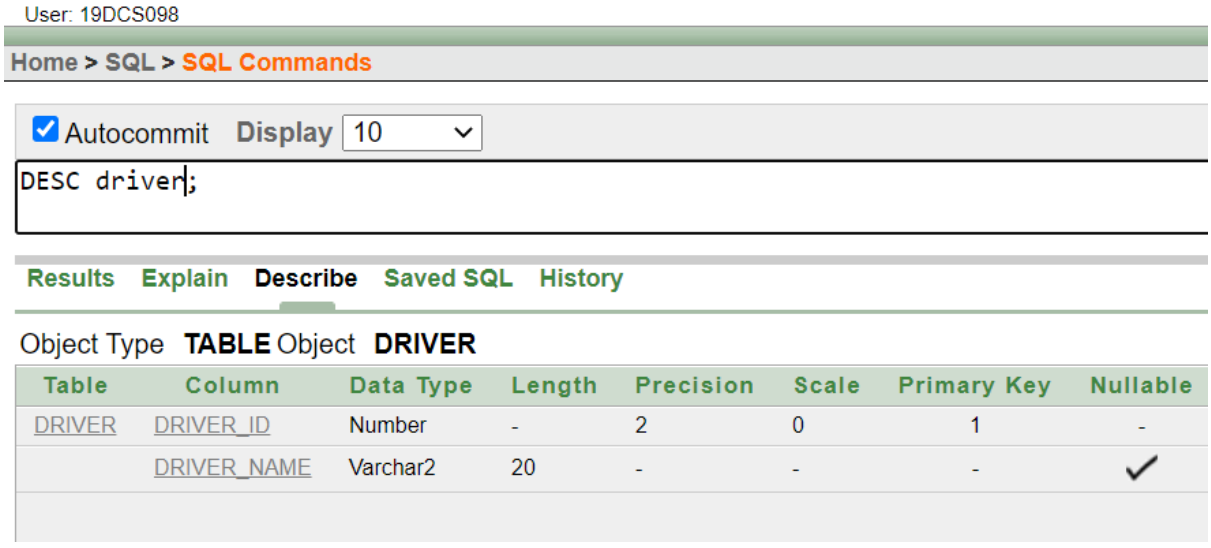
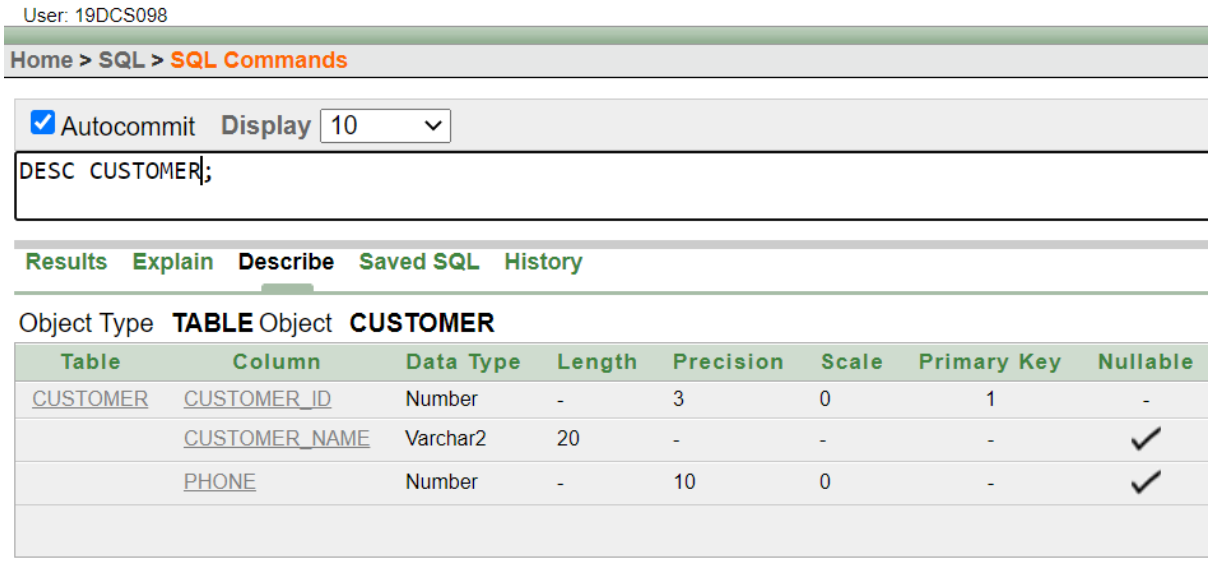


Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable
DRIVER	DRIVER_ID	Number	-	2	0	1	-
	DRIVER_NAME	Varchar2	20	-	-	-	✓

- **DETAILS OF CUSTOMER TABLE :**



User: 19DCS098

Home > SQL > SQL Commands

Autocommit Display 10

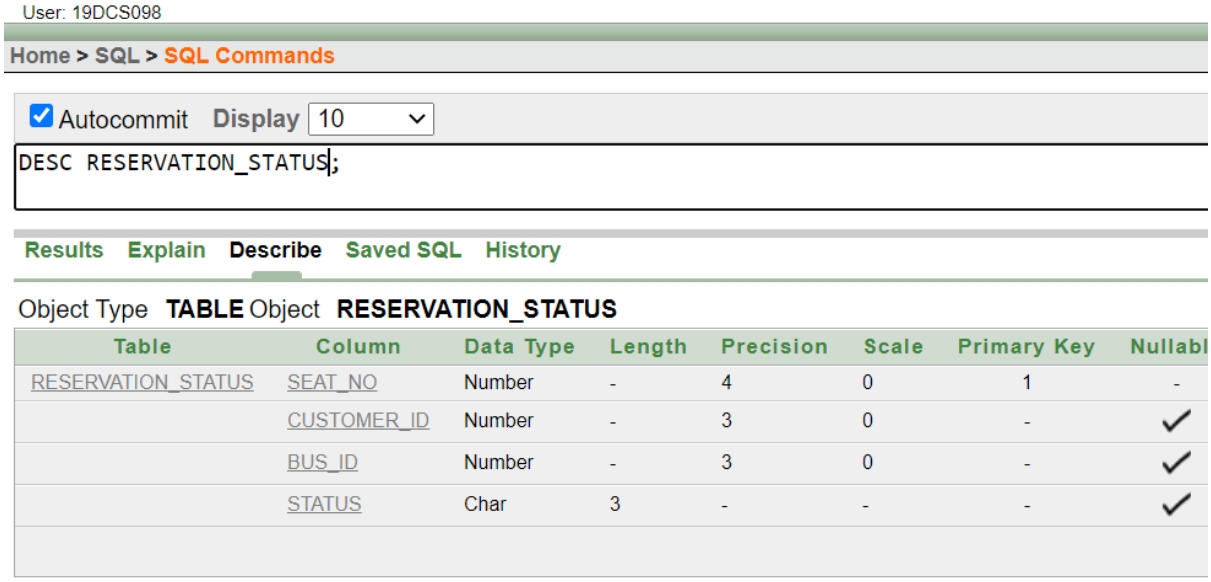
```
DESC CUSTOMER;
```

Results Explain Describe Saved SQL History

Object Type **TABLE** Object **CUSTOMER**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable
CUSTOMER	CUSTOMER_ID	Number	-	3	0	1	-
	CUSTOMER_NAME	Varchar2	20	-	-	-	✓
	PHONE	Number	-	10	0	-	✓

- **DETAILS OF RESERVATION_STATUS TABLE**:



User: 19DCS098

Home > SQL > SQL Commands

Autocommit Display 10

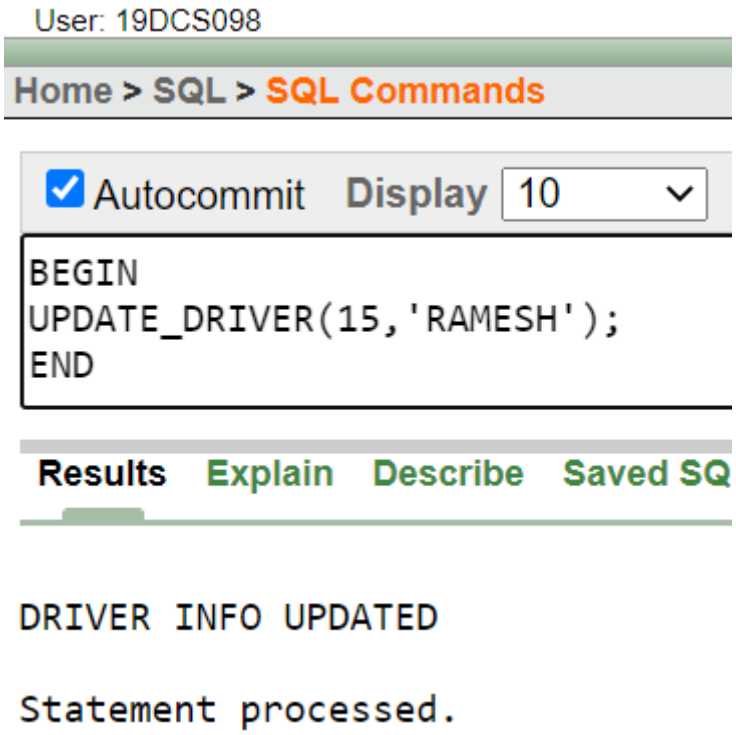
```
DESC RESERVATION_STATUS;
```

Results Explain Describe Saved SQL History

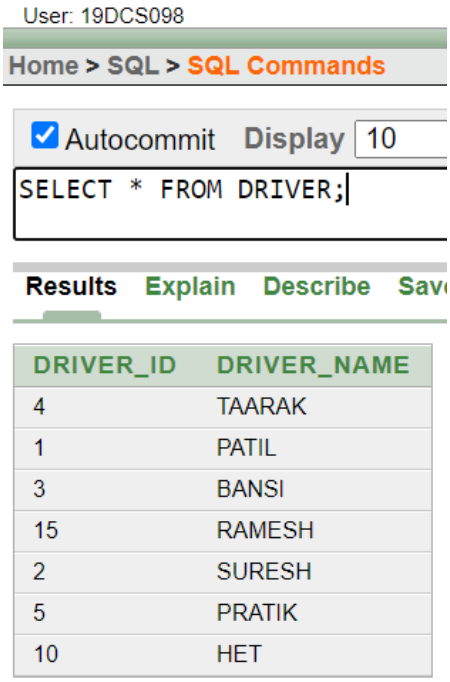
Object Type **TABLE** Object **RESERVATION_STATUS**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullabl
RESERVATION_STATUS	SEAT_NO	Number	-	4	0	1	-
	CUSTOMER_ID	Number	-	3	0	-	✓
	BUS_ID	Number	-	3	0	-	✓
	STATUS	Char	3	-	-	-	✓

- **INSERTING DRIVER'S INFO USING PL/SQL PROCEDURE**:

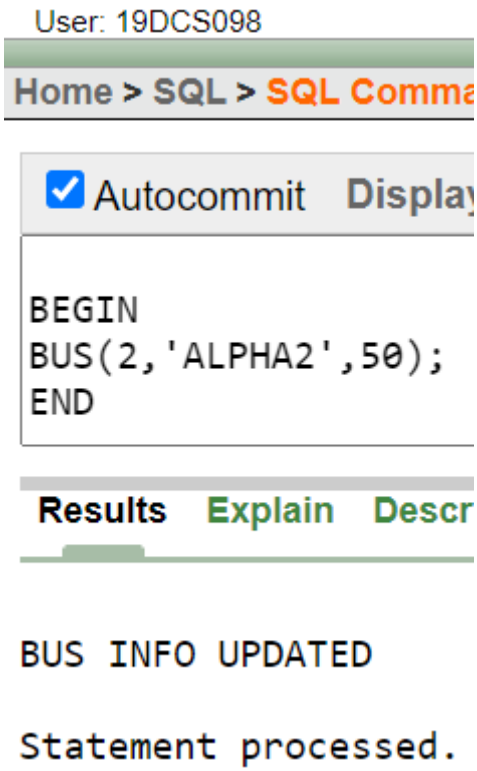


- **TABLE OF DRIVER AFTER UPPDATION:**

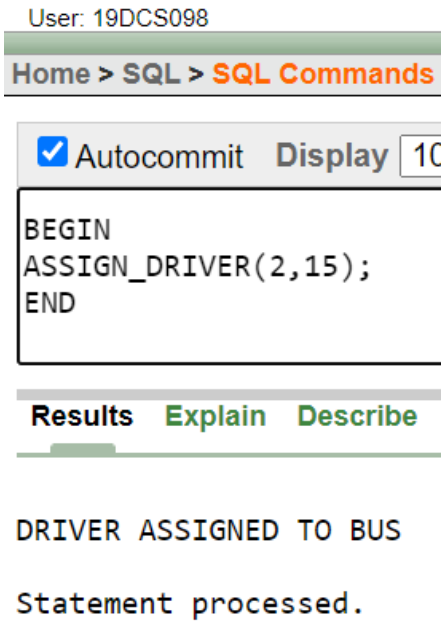


DRIVER_ID	DRIVER_NAME
4	TAARAK
1	PATIL
3	BANSI
15	RAMESH
2	SURESH
5	PRATIK
10	HET

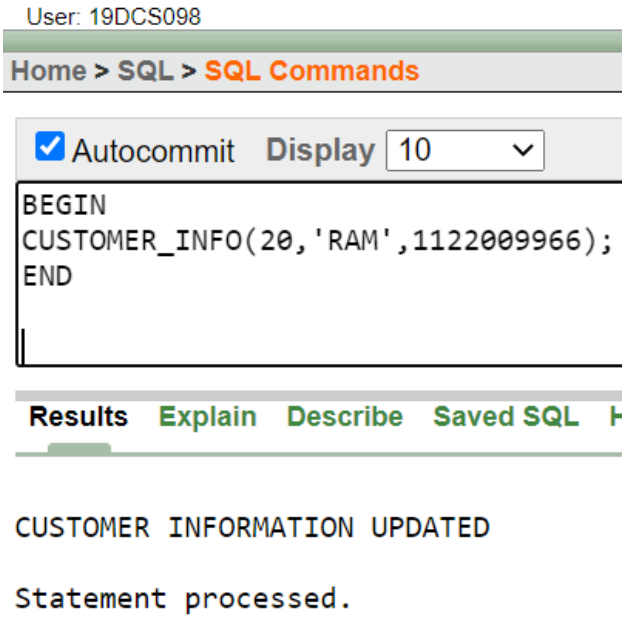
- **UPDATING BUS INFO USING PL/SQL PROCEDURE:**



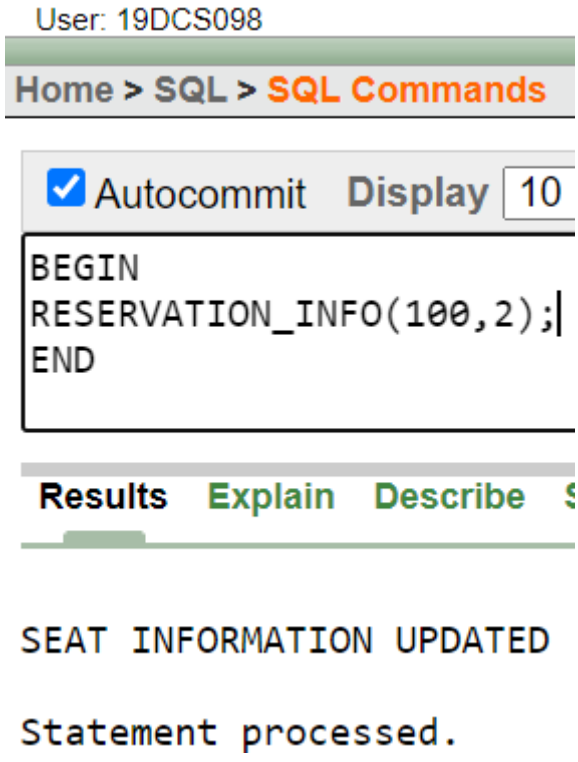
- **ASSIGN DRIVER TO HIS/HER BUS:**



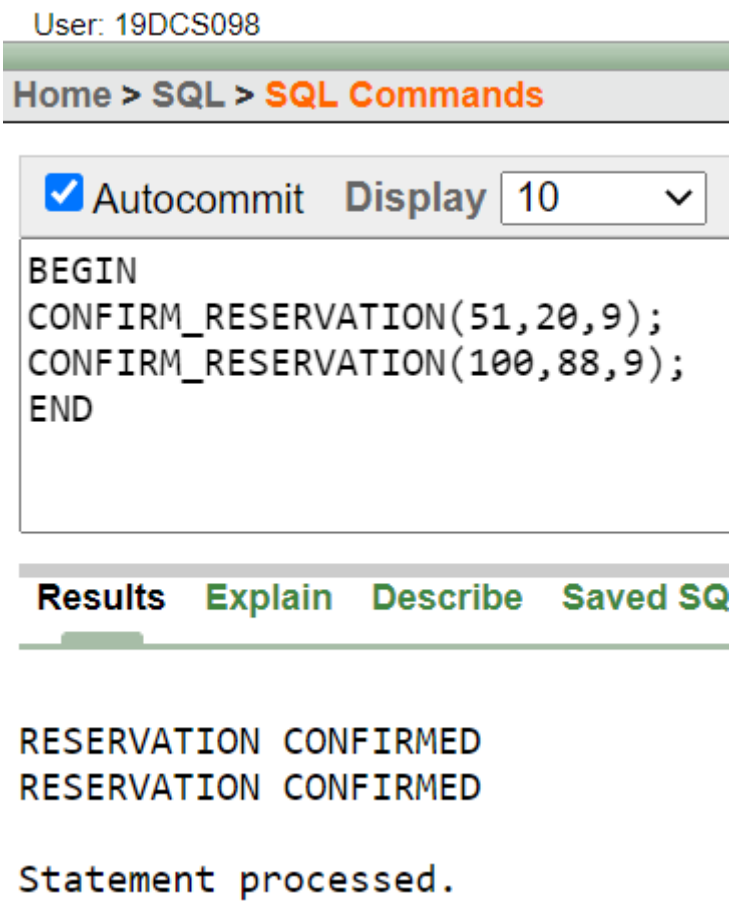
- **INSERTING DATA OF NEW CUSTOMER**



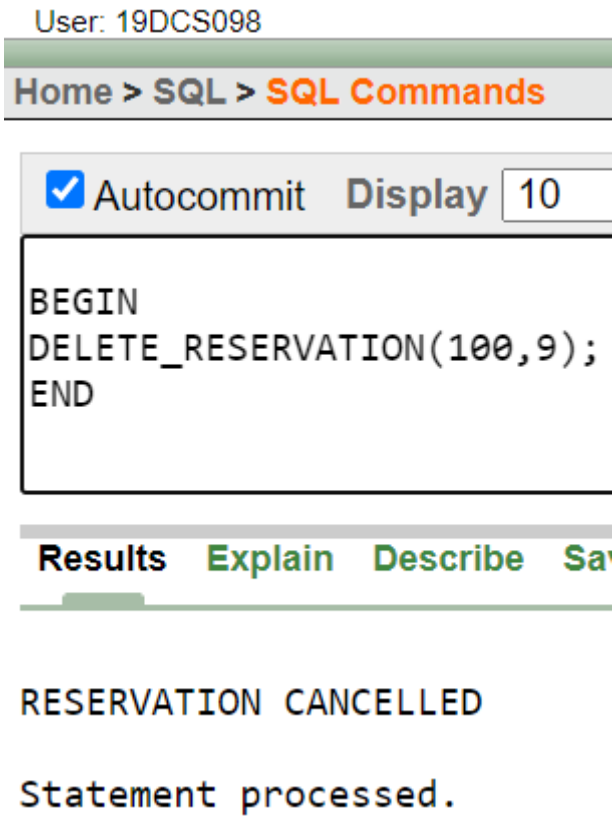
- **ASSIGNNING SEATS IN RESERVATION QUOTA**



- **CONFIRMING THE RESERVATION**



- **CANCEL THE RESERVATION**



- **GLIMPSES OF ALL TABLES AFTER PERFORMING THE ABOVE OPERATIONS:**

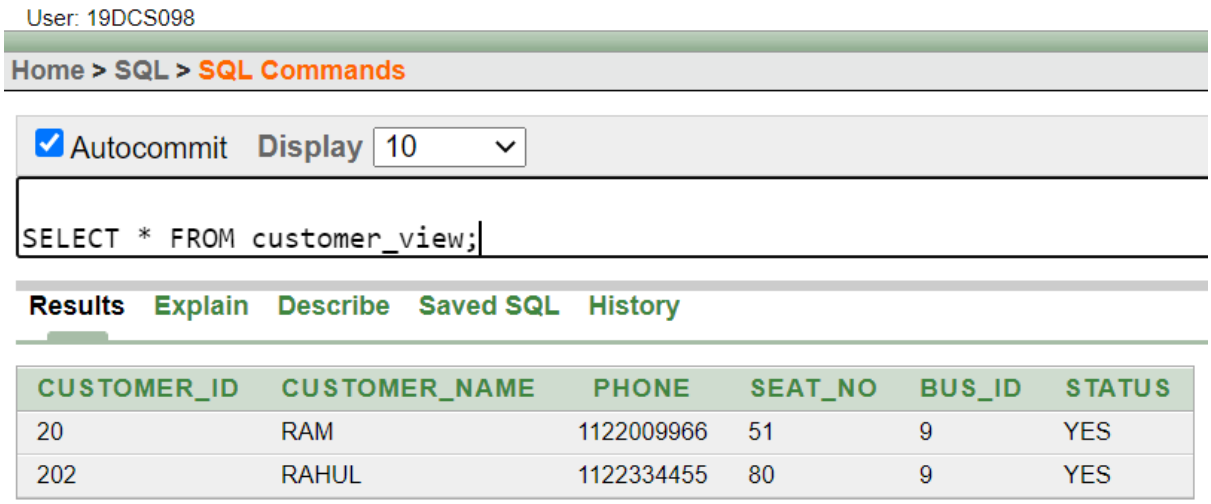
BUS_ID	BUS_NAME	DRIVER_ID	TOTAL_SEATS	RESERVED_SEATS
9	PASCAL1	2	45	2
2	ALPHA2	15	50	0
5	BETA1	4	60	0
11	FLAMINGO1	5	65	0
1	ALPHA1	1	50	0

DRIVER_ID	DRIVER_NAME
4	TAARAK
1	PATIL
3	BANSI
15	RAMESH
2	SURESH
5	PRATIK
10	HET

CUSTOMER_ID	CUSTOMER_NAME	PHONE
80	BHAVESHA	2433300098
111	NARENDRA	8140334422
20	RAM	1122009966
202	RAHUL	1122334455
88	PRASHANT	8877662233

SEAT_NO	CUSTOMER_ID	BUS_ID	STATUS
51	20	9	YES
80	202	9	YES
100	-	-	NO

- **GLIMPSE OF VIEWS OF CUSTOMERS HAVING CONFIRMED SEATS**

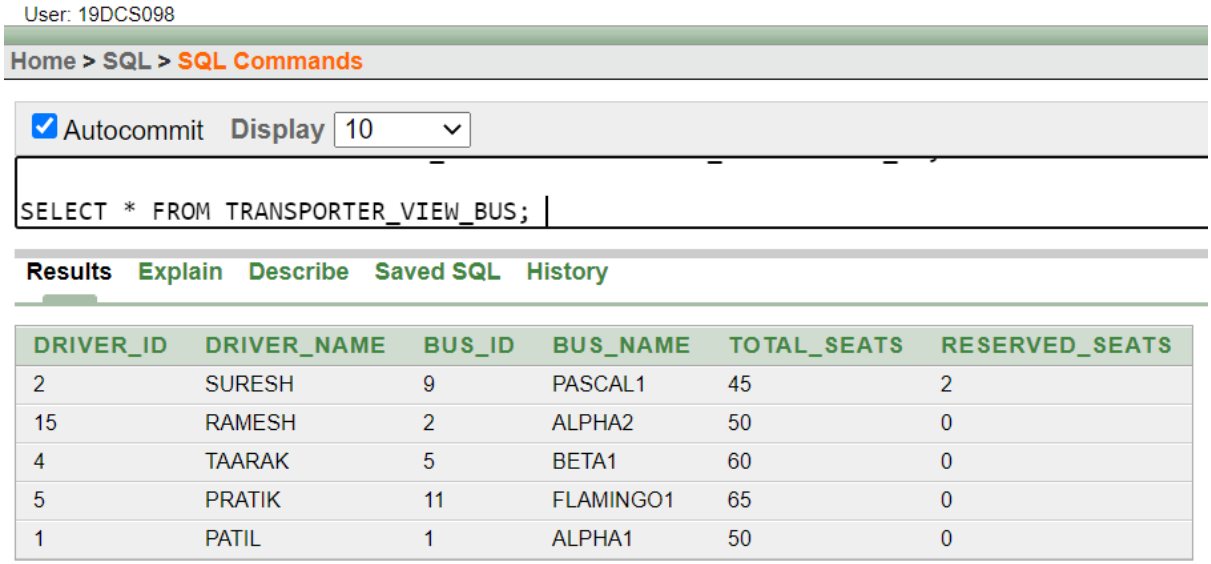


CUSTOMER_ID	CUSTOMER_NAME	PHONE	SEAT_NO	BUS_ID	STATUS
20	RAM	1122009966	51	9	YES
202	RAHUL	1122334455	80	9	YES

- **GLIMPSE OF VIEW OF ASSIGNED DRIVER**

DRIVER_ID	DRIVER_NAME	BUS_ID	BUS_NAME
2	SURESH	9	PASCAL1
15	RAMESH	2	ALPHA2
4	TAARAK	5	BETA1
5	PRATIK	11	FLAMINGO1
1	PATIL	1	ALPHA1

- **GLIMPSE OF OWNER**



DRIVER_ID	DRIVER_NAME	BUS_ID	BUS_NAME	TOTAL_SEATS	RESERVED_SEATS
2	SURESH	9	PASCAL1	45	2
15	RAMESH	2	ALPHA2	50	0
4	TAARAK	5	BETA1	60	0
5	PRATIK	11	FLAMINGO1	65	0
1	PATIL	1	ALPHA1	50	0

CONCLUSION:

By doing this mini project we polished our skills in DBMS,PL/SQL,SQL and other concepts