Indian Institute of Technology, Bhubaneswar

PROJECT ABSTRACT

LEISURE VOYAGE RESERVATION SYSTEM

CONTRIBUTORS

Abhinav Sharma – 19ME02030

abhinavons@gmail.com

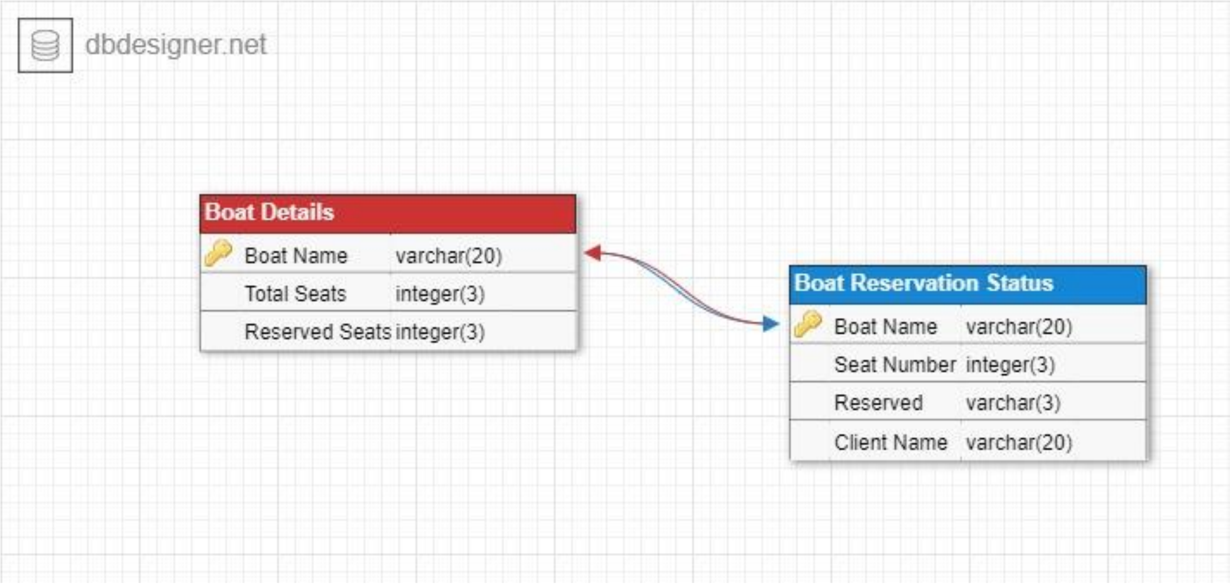
Abhinav Singh – 19ME02045

Abhinav.singh1911@gmail.com

Introduction

The project's main goal is to computerize a company's data management. It takes the place of all paper work. It maintains and ensures that the computerized boat reservation system is implemented successfully 100 percent of the time. There are three modules in this reservation system. The first module assists the customer in determining the availability of seats on a specific boat. The second module assists him in confirming a ticket reservation. He can cancel a reserved ticket using the third module.

The first module gathers information from tables that are needed for the inquiry. On reservation, the second module enters values into the tables. On ticket cancellation, the third module deletes values from the table. Because the database is hosted on the internet using Oracle Server, the application can access data from anywhere in the world and by a large number of people at the same time.



Boat Details	
Boat Name	varchar(20)
Total Seats	integer(3)
Reserved Seats	integer(3)

Boat Reservation Status	
Boat Name	varchar(20)
Seat Number	integer(3)
Reserved	varchar(3)
Client Name	varchar(20)

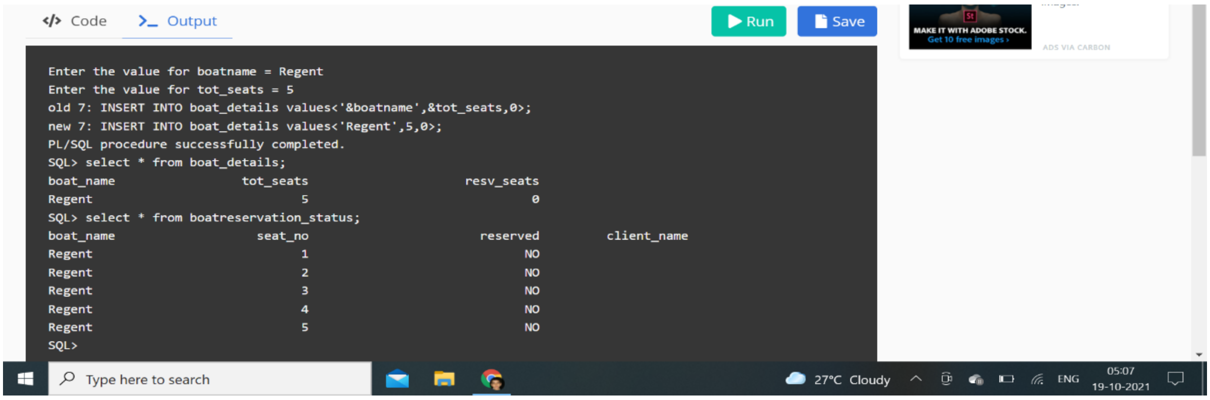
CREATING TABLES

```
CREATE TABLE boat_details(boat_name CHAR(20)PRIMARY KEY, tot_seats NUMBER(3), reserved_seats NUMBER(3)) ;
```

```
CREATE TABLE boatreservation_status(boat_name CHAR(20) REFERENCES boat_details(boat_name),seat_no NUMBER(3),reserved CHAR(3) CHECK(reserved in('YES','NO')),client_name CHAR(20));
```

BOAT DETAILS

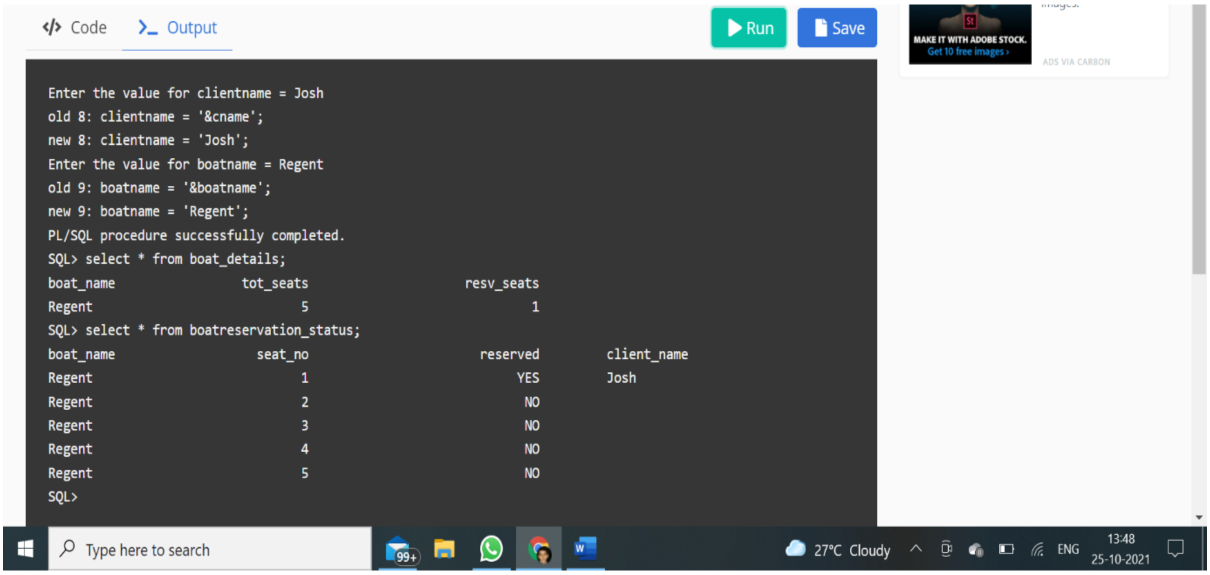
```
DECLARE
boatname CHAR(20);
tot_seats NUMBER(3);
resv_seats NUMBER(3);
CURSOR cur IS SELECT * FROM boat_details;
BEGIN
INSERT INTO boat_details VALUES('&boat_name',&tot_seatsal_seats,0);
OPEN cur;
loop
FETCH cur INTO boatname,tot_seats,resv_seats;
if cur % found then
for i in 1..tot_seats
loop
INSERT INTO boatreservation_status VALUES(boatname,i,'NO',NULL);
end loop;
else
exit;
end if;
end loop;
lose cur;
end;
```



```
Enter the value for boatname = Regent
Enter the value for tot_seats = 5
old 7: INSERT INTO boat_details values<'&boatname',&tot_seats,0>;
new 7: INSERT INTO boat_details values<'Regent',5,0>;
PL/SQL procedure successfully completed.
SQL> select * from boat_details;
boat_name                  tot_seats                    resv_seats
Regent                             5                             0
SQL> select * from boatreservation_status;
boat_name                    seat_no                      reserved         client_name
Regent                             1                            NO
Regent                             2                            NO
Regent                             3                            NO
Regent                             4                            NO
Regent                             5                            NO
SQL>
```

BOAT RESERVATION

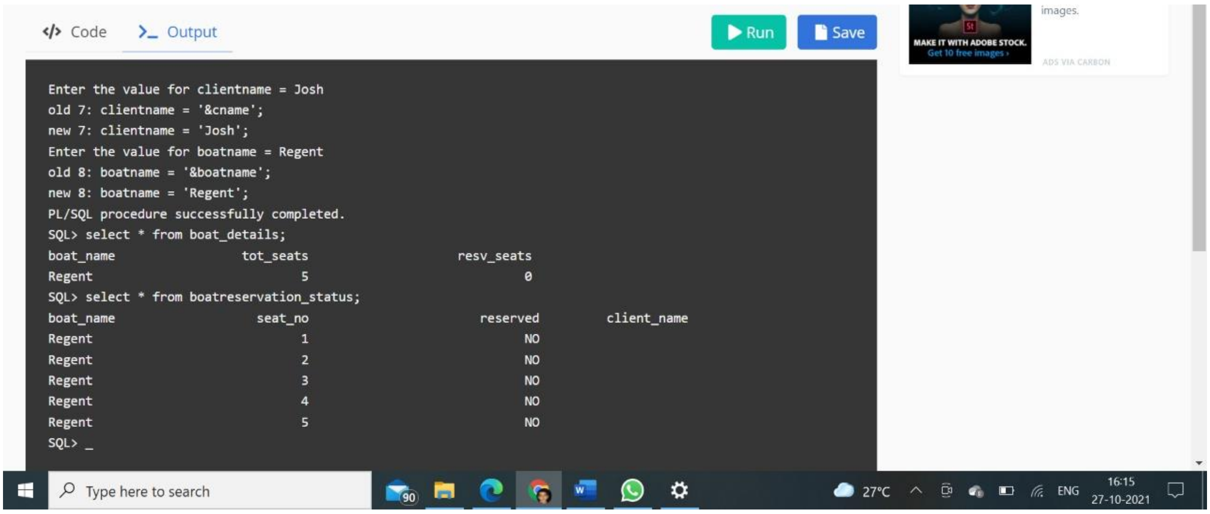
```
DECLARE
clientname CHAR(15);
boatname CHAR(15);
seatno NUMBER(3);
tot_seats NUMBER(3);
resv_seats NUMBER(3);
BEGIN
clientname:='&clientname';
boatname:='&boatname';
SELECT tot_seats INTO tot_seats FROM bus_details WHERE boat_name=boatname;
SELECT reserved_seats INTO resv_seats FROM boat_details WHERE
boat_name=boatname;
if tot_seats>resv_seats then
SELECT MIN(seat_id) INTO seatno FROM boatreservation_status WHERE
boat_name=boatname and reserved='NO';
UPDATE boatreservation_status SET reserved='YES' WHERE boat_name=boatname and
seat_no=seatno;
UPDATE busreservation_status SET client_name=clientname WHERE
boat_name=boatname and seat_no=seatno;
UPDATE boat_details SET reserved_seats=reserved_seats+1 WHERE
boat_name=boatname;
end if;
dbms_output.put_line('No seat avalable');
end;
```



```
Enter the value for clientname = Josh
old 8: clientname = '&cname';
new 8: clientname = 'Josh';
Enter the value for boatname = Regent
old 9: boatname = '&boatname';
new 9: boatname = 'Regent';
PL/SQL procedure successfully completed.
SQL> select * from boat_details;
boat_name                tot_seats                    resv_seats
Regent                           5                             1
SQL> select * from boatreservation_status;
boat_name                  seat_no                      reserved        client_name
Regent                           1                           YES        Josh
Regent                           2                            NO
Regent                           3                            NO
Regent                           4                            NO
Regent                           5                            NO
SQL>
```

TICKET CANCELLATION

```
Declare
clientname char(15);
boatname char(15);
seatno number(3);
resv_seats number(3);
BEGIN
clientname:='&clientname';
boatname:='&boatname';
SELECT seat_id INTO seatno FROM busreservation_status WHERE boat_name=boatname
and client_name=clientname;
SELECT reserved_seats INTO resv_seats FROM bus_details WHERE
boat_name=boatname;
if resv_seats<0 then
dbms_output.put_line('Cancelation not allow');
else
UPDATE boatreservation_status set reserved='n' where bus_name=boatname and
seat_id=seatno;
UPDATE boatreservation_status set customer_name=null WHERE bus_name=boatname and
seat_id=seatno;
UPDATE bus_details set reserved_seats=reserved_seats-1 WHERE bus_name=boatname;
end if;
end;
```



```
Enter the value for clientname = Josh
old 7: clientname = '&cname';
new 7: clientname = 'Josh';
Enter the value for boatname = Regent
old 8: boatname = '&boatname';
new 8: boatname = 'Regent';
PL/SQL procedure successfully completed.
SQL> select * from boat_details;
boat_name                 tot_seats                  resv_seats
Regent                            5                           0
SQL> select * from boatreservation_status;
boat_name                   seat_no                    reserved        client_name
Regent                            1                          NO
Regent                            2                          NO
Regent                            3                          NO
Regent                            4                          NO
Regent                            5                          NO
SQL> _
```

References:

Books:

Fundamentals of Database Systems - Ramez Elmasri, Shamkant B. Navathe

Oracle PL/SQL Programming - Feuerstein, SPD/O’REILLY

Website:

https://www.wikipedia.org

http://plsql-tutorial.com/

https://www.tutorialspoint.com