

DATABASE SYSTEMS LAB

INTERNAL QUIZ

Nirajit Pramanik 22BAI1347

Question 1:

Consider the Sailors-Boats-Reserves DB described in the text.

sailor (sid, sname, rating, age)

boat (bid, bname, color)

reserves (sid, bid, rdate)

Write each of the following queries in SQL

(a) Create the above tables with constraints and insert minimum 5 records into it. **Check Qn. 2 for inserting sid values** [6]

(b) Create a sequence and insert sid values (contains only odd numbers) starting from 101 [2]

```
CREATE TABLE SAILOR_1347(  
    SID INT PRIMARY KEY,  
    SNAME VARCHAR(20),  
    RATING INT,  
    AGE INT  
);
```

```
CREATE TABLE BOAT_1347(  
    BID INT PRIMARY KEY,  
    BNAME VARCHAR(20),  
    COLOR VARCHAR(15)  
);
```

```
CREATE TABLE RESERVES_1347(  
    SID INT PRIMARY KEY,  
    BID INT PRIMARY KEY,  
    RDATE DATE  
);
```

```
SID INT,  
BID INT,  
RDATE DATE,  
FOREIGN KEY(SID) REFERENCES SAILOR_1347(SID),  
FOREIGN KEY(BID) REFERENCES BOAT_1347(BID)  
);
```

--part (b)

```
CREATE SEQUENCE SEQ1  
START WITH 99  
MINVALUE 99  
MAXVALUE 201  
INCREMENT BY 2;
```

```
INSERT INTO SAILOR_1347 VALUES(SEQ1.NEXTVAL, 'JOHN', 4, 25);  
INSERT INTO SAILOR_1347 VALUES(SEQ1.NEXTVAL, 'JACOB', 4, 25);  
INSERT INTO SAILOR_1347 VALUES(SEQ1.NEXTVAL, 'STEVE', 2, 29);  
INSERT INTO SAILOR_1347 VALUES(SEQ1.NEXTVAL, 'BOBBY', 3, 32);  
INSERT INTO SAILOR_1347 VALUES(SEQ1.NEXTVAL, 'THOR', 5, 39);  
INSERT INTO SAILOR_1347 VALUES(SEQ1.NEXTVAL, 'CHESTER', 4, 41);
```

```
INSERT INTO BOAT_1347 VALUES(101, 'MARS', 'BLUE');  
INSERT INTO BOAT_1347 VALUES(102, 'STORM TROOPER', 'RED');  
INSERT INTO BOAT_1347 VALUES(103, 'MIDSTORM', 'GREEN');  
INSERT INTO BOAT_1347 VALUES(104, 'TITANIC', 'BLUE');  
INSERT INTO BOAT_1347 VALUES(105, 'WAVECRASHER', 'BLACK');
```

```
INSERT INTO RESERVES_1347 VALUES(101, 101, '21/MAR/2024');  
INSERT INTO RESERVES_1347 VALUES(103, 103, '22/FEB/2024');
```

```
INSERT INTO RESERVES_1347 VALUES(105, 104, '15/JAN/2024');
INSERT INTO RESERVES_1347 VALUES(107, 102, '02/APR/2024');
INSERT INTO RESERVES_1347 VALUES(111, 102, '02/JAN/2024');
INSERT INTO RESERVES_1347 VALUES(109, 105, '22/SEP/2024');
INSERT INTO RESERVES_1347 VALUES(107, 105, '22/JAN/2024');
INSERT INTO RESERVES_1347 VALUES(105, 105, '22/MAR/2024');
INSERT INTO RESERVES_1347 VALUES(103, 105, '22/MAY/2024');
INSERT INTO RESERVES_1347 VALUES(101, 105, '22/JUL/2024');
```

--displaying tables

```
SELECT * FROM SAILOR_1347;
SELECT * FROM BOAT_1347;
SELECT * FROM RESERVES_1347;
```

Table created.

Table created.

Table created.

Sequence created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

1 row created.

SID	SNAME	RATING	AGE
101	JOHN	4	25
103	JACOB	4	25
105	STEVE	2	29
107	BOBBY	3	32
109	THOR	5	39
111	CHESTER	4	41

6 rows selected.

BID	BNAME	COLOR
101	MARS	BLUE
102	STORM TROOPER	RED
103	MIDSTORM	GREEN
104	TITANIC	BLUE
105	WAVECRASHER	BLACK

SID	BID	RDATE
101	101	21-MAR-24
103	103	22-FEB-24
105	104	15-JAN-24
107	102	02-APR-24
111	102	02-JAN-24
109	105	22-SEP-24
107	105	22-JAN-24
105	105	22-MAR-24
103	105	22-MAY-24
101	105	22-JUL-24

10 rows selected.

(c) Find all sailor id's of sailors who have a rating of at least 5 or reserved boat 103. [2]

```
SELECT S.SNAME FROM SAILOR_1347 S
JOIN RESERVES_1347 R ON S.SID = R.SID
WHERE (S.RATING >= 5) OR (R.BID = 103);
```

```
SNAME
-----
JACOB
THOR
```

(d) Find the names of sailors who have not reserved a boat whose name contains the string "storm". Order the names in ascending order. [2]

```
SELECT DISTINCT(S.SNAME) FROM SAILOR_1347 S
JOIN RESERVES_1347 R ON S.SID = R.SID
JOIN BOAT_1347 B ON R.BID = B.BID
WHERE R.BID NOT IN (SELECT BID FROM BOAT_1347 WHERE BNAME LIKE '%STORM%')
ORDER BY S.SNAME;
```

```
SNAME
-----
BOBBY
JACOB
JOHN
STEVE
THOR
```

(e) Update the sailor table by adding mobile_no and check to see that the values only start with "91" [2]

```
ALTER TABLE SAILOR_1347 ADD MOBILE_NO VARCHAR(11);
ALTER TABLE SAILOR_1347 ADD CONSTRAINT MOB_CONST CHECK (MOBILE_NO LIKE '91%');
```

```
Table altered.

Table altered.
```

(f) Create a sub table for sailor table to view sailor id's of sailors with the lowest rating [2]

```
CREATE TABLE LOWRATING_1347 AS
```

```
(SELECT * FROM SAILOR_1347 WHERE RATING=(SELECT MIN(RATING) FROM  
SAILOR_1347));
```

```
SELECT * FROM LOWRATING_1347;
```

Table created.

SID	SNAME	RATING	AGE	MOBILE_NO
105	STEVE	2	29	

(g) For each boat which was reserved by at least 5 distinct sailors, find the boat id and the average age of sailors who reserved it. [2]

```
SELECT R.BID, AVG(S.AGE) AS AVERAGE_AGE, COUNT(R.BID) AS SAILORS_COUNT  
FROM BOAT_1347 B
```

```
JOIN RESERVES_1347 R ON B.BID = R.BID
```

```
JOIN SAILOR_1347 S ON S.SID = R.SID
```

```
GROUP BY R.BID
```

```
HAVING COUNT(R.BID) >= 5;
```

BID	AVERAGE_AGE	SAILORS_COUNT
105	30	5

(h) Find the bid reserved by the oldest sailor. [2]

```
SELECT R.BID FROM BOAT_1347 B
```

```
JOIN RESERVES_1347 R ON R.BID = B.BID
```

```
JOIN SAILOR_1347 S ON S.SID = R.SID
```

```
WHERE S.AGE = (SELECT MAX(AGE) FROM SAILOR_1347);
```

BID
102