**Lab Assignment 8- Join and Sub Queries**

**Create the following tables and insert some tuples in these tables shown below. Where sid is the**

**primary key for the Sailors table, bid is the primary key for the Boats table and sid and bid are**

**the foreign keys for the Reserves table referencing to the Sailors and Boats table, respectively.**

**Sailors(sid: integer, sname: string, rating: integer, age: real)**

**Boats(bid: integer, bname: string, color: string)**

**Reserves(sid: integer, bid: integer, day: date)**

**After inserting the records in these tables, the instances should look like as follows:**

A table with numbers and a few words

Description automatically generated with medium confidence

CREATE TABLE SAILORS(

SID NUMBER,

SNAME VARCHAR(20),

RATING NUMBER,

AGE DECIMAL(3,1),

CONSTRAINT S\_PK PRIMARY KEY (SID)

);

INSERT INTO SAILORS VALUES(22,'Dustin',7,45.0);

INSERT INTO SAILORS VALUES(29,'Brutus',1,33.0);

INSERT INTO SAILORS VALUES(31,'Lubber',8,55.5);

INSERT INTO SAILORS VALUES(32,'Andy',8,25.5);

INSERT INTO SAILORS VALUES(58,'Rusty',10,35.0);

INSERT INTO SAILORS VALUES(64,'Horatio',7,35.0);

INSERT INTO SAILORS VALUES(71,'Zorba',10,16.0);

INSERT INTO SAILORS VALUES(74,'Horatio',9,35.0);

INSERT INTO SAILORS VALUES(85,'Art',3,25.5);

INSERT INTO SAILORS VALUES(95,'Bob',3,63.5);

SELECT \* FROM SAILORS;

CREATE TABLE BOATS(

BID NUMBER,

BNAME VARCHAR(20),

COLOR VARCHAR(20),

CONSTRAINT B\_PK PRIMARY KEY(BID)

);

INSERT INTO BOATS VALUES(101,'Interlake','blue');

INSERT INTO BOATS VALUES(102,'Interlake','red');

INSERT INTO BOATS VALUES(103,'Clipper','green');

INSERT INTO BOATS VALUES(104,'Marine','red');

SELECT \* FROM BOATS;

CREATE TABLE RESERVES(

SID NUMBER,

BID NUMBER,

DAY DATE,

CONSTRAINT R\_SFK FOREIGN KEY (SID) REFERENCES SAILORS(SID),

CONSTRAINT R\_BPK FOREIGN KEY (BID) REFERENCES BOATS(BID),

CONSTRAINT R\_PK PRIMARY KEY (SID,BID)

);

INSERT INTO RESERVES VALUES(22,101,'10 OCTOBER 1998');

INSERT INTO RESERVES VALUES(22,102,'10 OCTOBER 1998');

INSERT INTO RESERVES VALUES(22,103,'10 AUGUST 1998');

INSERT INTO RESERVES VALUES(22,104,'10 JULY 1998');

INSERT INTO RESERVES VALUES(31,102,'11 OCTOBER 1998');

INSERT INTO RESERVES VALUES(31,103,'11 JUNE 1998');

INSERT INTO RESERVES VALUES(31,104,'11 DECEMBER 1998');

INSERT INTO RESERVES VALUES(64,101,'9 MAY 1998');

INSERT INTO RESERVES VALUES(64,102,'9 AUGUST 1998');

INSERT INTO RESERVES VALUES(74,103,'9 AUGUST 1998');

SELECT \* FROM RESERVES;

**Write SQL command using JOINs and/or Sub-queries for the following:**

**Q1. Find the names of sailors who have reserved a red or a green boat.**

SELECT S.SNAME

FROM SAILORS S

WHERE S.SID IN(

SELECT DISTINCT R.SID

FROM RESERVES R

WHERE R.BID IN (

SELECT B.BID

FROM BOATS B

WHERE B.COLOR IN ('red','green')

)

);

A screenshot of a phone

Description automatically generated

**Q2. Find the names of sailors who have reserved both a red and a green boat.**

SELECT SNAME

FROM SAILORS

WHERE SID IN (

SELECT DISTINCT R.SID

FROM RESERVES R

WHERE EXISTS (

SELECT SID

FROM RESERVES

WHERE BID IN (

SELECT BID FROM BOATS

WHERE COLOR='red'

)

AND SID=R.SID

) AND EXISTS(

SELECT SID

FROM RESERVES

WHERE BID=(

SELECT BID FROM BOATS

WHERE COLOR='green'

)

AND SID = R.SID

)

);

A screenshot of a phone

Description automatically generated

**Q3. Find the names of sailors who have reserved boat 103**

SELECT S.SNAME

FROM SAILORS S

WHERE S.SID IN (

SELECT R.SID

FROM RESERVES R

WHERE R.BID=103

);

**A screenshot of a phone

Description automatically generated**

**Q4. Find the names of sailors who have reserved a red boat.**

SELECT S.SNAME

FROM SAILORS S

WHERE S.SID IN (

SELECT R.SID

FROM RESERVES R

WHERE R.BID IN (

SELECT B.BID

FROM BOATS B

WHERE COLOR = 'red'

)

);

A screenshot of a phone

Description automatically generated

**Q5. Find the names of sailors who have NOT reserved a red boat.**

SELECT S.SNAME

FROM SAILORS S

WHERE S.SID NOT IN (

SELECT R.SID

FROM RESERVES R

WHERE R.BID IN (

SELECT B.BID

FROM BOATS B

WHERE COLOR = 'red'

)

);

A screenshot of a phone

Description automatically generated

**Q6. Find the names of sailors who have reserved at least one boat.**

SELECT S.SNAME

FROM SAILORS S

WHERE S.SID IN (

SELECT DISTINCT R.SID

FROMRESERVES R

);

A screenshot of a phone

Description automatically generated

**Q7. Find sailors whose rating is better than some sailor called Horatio.**

SELECT S.SNAME

FROM SAILORS S

WHERE S.RATING > ANY (

SELECT RATING

FROM SAILORS

WHERE SNAME='Horatio'

);

**A screenshot of a phone

Description automatically generated**

**Q8. Find sailors whose rating is better than every sailor called Horatio.**

SELECT S.SNAME

FROM SAILORS S

WHERE S.RATING > ALL (

SELECT RATING

FROM SAILORS

WHERE SNAME='Horatio'

);

A screenshot of a computer

Description automatically generated

**Q9. Find the sailors with the highest rating.**

SELECT S.SNAME

FROM SAILORS S

WHERE S.RATING = (

SELECT MAX(RATING)

FROM SAILORS

);

A screenshot of a phone

Description automatically generated

**Q10. Find the names of sailors who have reserved all boats.**

SELECT S.SNAME

FROM SAILORS S

WHERE NOT EXISTS(

SELECT B.BID

FROM BOATS B

WHERE NOT EXISTS (

SELECT R.BID

FROM RESERVES R

WHERE R.BID=B.BID

AND R.SID = S.SID

)

);

A screenshot of a computer

Description automatically generated